Statistics and Its Interface Volume 6 (2013) 559–576

Bayesian analysis for exponential random graph models using the adaptive exchange sampler*

Ick Hoon Jin†, Ying Yuan† and Faming Liang§,‡

Exponential random graph models have been widely used in social network analysis. However, these models are extremely difficult to handle from a statistical viewpoint, because of the existence of intractable normalizing constants. In this paper, we consider a fully Bayesian analysis for exponential random graph models using the adaptive exchange sampler, which solves the issue of intractable normalizing constants encountered in Markov chain Monte Carlo (MCMC) simulations. The adaptive exchange sampler can be viewed as a MCMC extension of the exchange algorithm, and it generates auxiliary networks via an importance sampling procedure from an auxiliary Markov chain running in parallel. The convergence of this algorithm is established under mild conditions. The adaptive exchange sampler is illustrated using a few social networks, including the Florentine business network, molecule synthetic network, and dolphins network. The results indicate that the adaptive exchange algorithm can produce more accurate estimates than approximate exchange algorithms, while maintaining the same computational efficiency.

AMS 2000 subject classifications: Primary 65C05; secondary 05C80.
Keywords and phrases: Exchange algorithm, Exponential random graph model, Adaptive Markov chain Monte Carlo, Social network.

1. INTRODUCTION

The social network is a social structure made of actors (individuals, organizations, etc.), which are interconnected by a certain relationship, such as friendship, common interest, financial exchange, etc. The network can be represented in a graph with a node for each actor and an edge for each relation between a pair of actors. This graph representation can provide insight into organizational structures, social behavior patterns, and a variety of other social phenomena. Recently, social network analysis has been applied to many other disciplines, such as biology [42] and politics [7].

Many models have been proposed in the literature for social network analysis, including the Bernoulli random graph model [9], p_1 model [18], p_2 model [48], Markov random graph model [10], exponential random graph model [45], among others. The model of particular interest is the exponential random graph model (ERGM), which describes parsimoniously the local transitivities that shape the global structure of a network [22]. An ERGM allows one to include various network dependent structures in the analysis and thus can generally improve goodness-of-fit of social networks. The information from an ERGM can be used to understand a particular phenomenon or to simulate new random realizations to networks that retain the essential properties of the original. See [40] and [41] for an overview of ERGMs.

Consider a social network with n nodes, which can be either directed or undirected. The network can be specified in an $n \times n$-matrix $\boldsymbol{y} = (y_{ij})$, where $y_{ij} = 1$ if there is an edge between node i and node j and 0 otherwise. This matrix is also known as the adjacency matrix. The likelihood function of the ERGM is given by

$$f(\boldsymbol{y}|\theta) = \frac{1}{\kappa(\theta)} \exp\left\{\sum_{i\in\mathcal{A}} \theta_i S_i(\boldsymbol{y})\right\}, \tag{1}$$

where $S_i(\boldsymbol{y})$ denotes the network statistic (explained in Section 2), θ_i is the corresponding parameter, $\mathcal{A}$ specifies a set of network statistics included in the model, and $\kappa(\theta)$ is an intractable normalizing constant with $\theta = \{\theta_i : i \in \mathcal{A}\}$. The parameter θ_i measures how likely a structure summarized by $S_i(\boldsymbol{y})$ appears in the network. A larger value of θ_i indicates a higher probability that the structure described by $S_i(\boldsymbol{y})$ appears. The parameter θ_i can also be interpreted as the log-odds of different types of ties [22]. If we assign θ a prior density $\pi(\theta)$, the posterior density of θ is given by

$$\pi(\theta|\boldsymbol{y}) \propto \frac{\pi(\theta)}{\kappa(\theta)} \exp\left\{\sum_{i\in\mathcal{A}} \theta_i S_i(\boldsymbol{y})\right\}. \tag{2}$$

Sampling from this posterior density function is challenging due to the intractability of $\kappa(\theta)$. As a result, the existing

*The authors thank the editor, the associate editor, and referees for their detailed and constructive comments, which have led to significant improvement of this article.
†Yuan and Jin acknowledge support from the NIH grant R01 CA154591.
‡Liang's research was partially supported by grants from the National Science Foundation (DMS-1007457, DMS-1106494 and DMS-1317131) and the award (KUS-C1-016-04) made by King Abdullah University of Science and Technology (KAUST).
§Corresponding author.

Monte Carlo algorithms, such as the Metropolis-Hastings (MH) algorithm, cannot be directly applied to sample from the posterior (2) as the acceptance probability would involve an unknown normalizing constant ratio $\kappa(\theta)/\kappa(\theta')$, where θ' denotes the proposed parameter value.

The posterior sampling for the EGRM was further complicated by the problem of model degeneracy, which refers to the phenomenon that for some configuration of θ, the ERGM has all or most of its probability mass on just one or a few possible networks, typically the complete (fully connected) or empty (entirely unconnected) graphs [17]. The model degeneracy can cause serious issues on MCMC simulations and statistical inference for the ERGM [43]. For example, the Markov chain can mix very slowly, and in particular, if the mode of (1) is not unique, the Markov chain may be trapped at one of the modes.

Many methods have been proposed to address the problem of the intractability of $\kappa(\theta)$. Instead of directly sampling from $\pi(\theta|\boldsymbol{y})$, Møller [35] proposed to augment the posterior distribution $\pi(\theta|\boldsymbol{y})$ to $\pi(\theta, \boldsymbol{x}|\boldsymbol{y})$ by including the auxiliary variable $\boldsymbol{x}$. By choosing an appropriate auxiliary variable distribution $f(\boldsymbol{x}|\theta, \boldsymbol{y})$ and an appropriate proposal distribution $f(\theta'|\theta, \boldsymbol{y})$, the unknown normalizing constants $\kappa(\theta)/\kappa(\theta')$ canceled in simulation. The Møller's algorithm was further improved by the exchange algorithm [36] based on the idea of the parallel tempering algorithm [11]. Unfortunately, both the Møller's and exchange algorithms cannot be directly applied to the ERGM because these two methods require a perfect sampler, which is not available for the ERGM, for generating auxiliary social networks.

To address this issue, Liang [26] proposed the double Metropolis-Hastings (DMH) sampler, which avoids the use of perfect samples by using the auxiliary sample that is drawn via a short MH chain initialized at the observation $\boldsymbol{y}$. Although initializing the auxiliary MH chain with the observation $\boldsymbol{y}$ improves the convergence of the algorithm, the convergence of the short MH chain is not theoretically guaranteed and the resulting estimator can be biased.

Alternatively, some methods directly approximate the normalizing constant ratio $\kappa(\theta)/\kappa(\theta')$ or the function $\kappa(\theta)$ using Monte Carlo samples. Koskinen [25] and Liang and Jin [27], proposed to approximate the normalizing constant ratio $\kappa(\theta)/\kappa(\theta')$ at each iteration using an importance sampling approach. However, like the DMH sampler, these algorithms also suffer from the theoretical drawback on convergence, i.e., the importance sampling estimator might fail to converge to the true ratio $\kappa(\theta)/\kappa(\theta')$ with only a finite number of samples. By viewing $\kappa(\theta)$ as a marginal density of $g(\boldsymbol{y}, \theta) \propto \exp\{\sum_{i\in\mathcal{A}} \theta_i S_i(\mathbf{y})\}$, Atchade, Lartillot and Robert [3] proposed to approximate $\kappa(\theta)$ using Monte Carlo samples, but that method is usually very time consuming, especially when the dimension of θ is high. In addition, due to the model degeneracy of the ERGMs, approximation of $\kappa(\theta)$ over the entire parameter space is often impractical.

In this paper, we propose to conduct the Bayesian analysis for ERGMs using the adaptive exchange (AEX) algorithm [28]. The AEX algorithm is an adaptive Monte Carlo version of the exchange algorithm, where the auxiliary variables are generated via an importance sampling procedure from an auxiliary Markov chain running in parallel. To address the model degeneracy problem, we propose to use the approximate Bayesian computation (ABC) algorithm ([5], [33]) to select some auxiliary parameter points on which the auxiliary Markov chain is running. We study the convergence theory of AEX under relaxed conditions compared to those given in [28], which extends the application of AEX to problems with non-compact parameter spaces.

Compared to the exchange algorithm, the proposed AEX algorithm removes the requirement of perfect sampling, and thus is applicable to ERGMs. Compared to the DMH sampler, the AEX overcomes its theoretical flaw on convergence, while maintaining its computational efficiency. Compared to the normalizing constant or normalizing constant ratio-approximation methods, the AEX is more efficient and, more importantly, its convergence is ensured as the number of iterations becomes large.

This paper focus on the Bayesian analysis of ERGMs, but we note some frequentist approaches, including the maximum pseudo-likelihood method (see, e.g., [46]; [49]), Monte Carlo maximum likelihood estimation (MCMLE) method (see, e.g., [4]; [13]; [21]), and stochastic approximation-based methods ([23]; [44]), among others.

The paper is organized as follows. Section 2 starts with a description of the ERGM, and then we discuss the associated model degeneracy problem. In Section 3, we describe the AEX sampler and study its convergence. Section 4 is dedicated to an ABC-based method for auxiliary parameter points selection. Section 5 presents the numerical results of AEX for several social network examples. Section 6 concludes the paper with a brief discussion.

2. EXPONENTIAL RANDOM GRAPH MODELS, MODEL DEGENERACY AND PRIOR SELECTION

To define an ERGM, one needs to specify the network statistics $S_i(\boldsymbol{y})$ appeared in (1). Among a large number of network statistics which are available in the ERGMs, we will consider here some commonly used ones, including basic Markovian statistics [10], heterogeneity of degree statistics, and high-order transitivity statistics [45].

Basic Markovian statistics This class of statistics describe the basic structure of a network, including the edge count, two-star count, ..., k-star count (k_2-star, ..., k_k-star), and triangle count. The edge count, denoted by $S_1(\boldsymbol{y})$, is the count of the edges contained in the network $\boldsymbol{y}$. The two-star refers to a structure with one node connecting to two other nodes, and the k-star is a structure that one node

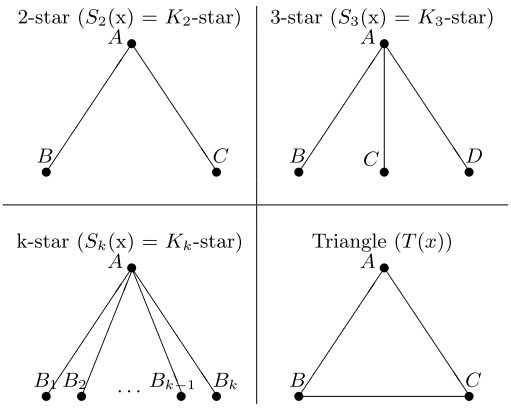

Figure 1. Visualization of basic Markovian statistics.

connects to k other nodes. The counts of two-stars, ..., k-stars are denoted by $S_2(\boldsymbol{y})$, ..., $S_k(\boldsymbol{y})$, respectively. If node 'a' connects to node 'b', node 'b' connects to 'c', and node 'c' connects to node 'a' simultaneously, then the nodes 'a', 'b', and 'c' form a triangle. The count of triangles is denoted by $T(\boldsymbol{y})$. Mathematically, the statistics $S_k(\boldsymbol{y})$ $(k = 1, \ldots, n-1)$ and $T(\boldsymbol{y})$ can be defined by

$$
\begin{aligned}
S_k(\boldsymbol{y}) &= \sum_{1 \le i \le n} \binom{y_{i+}}{k}, \quad k = 1, 2, n-1; \\
T(\boldsymbol{y}) &= \sum_{1 \le i < j < h \le n} y_{ij} y_{ih} y_{jh},
\end{aligned} \tag{3}
$$

where y_{i+} denotes the degree of nodes i, and the degree of a node refers to the number of edges incident to it.

Geometrically weighted degree Let the degree count, $D_k(\boldsymbol{y})$, denote the number of nodes with degree k. Since the number of stars is a function of the degrees, $D_k(\boldsymbol{y})$ is equivalent to modeling the k-star statistic. The geometrically weighted degree (GWD) statistic ([19]; [21]; [45]) enables to model all degree distributions as a function of single parameter by placing decreasing weights on the higher degrees. GWD is defined by

$$
u(\boldsymbol{y}|\tau) = e^{\tau} \sum_{k=1}^{n-2} \left\{ 1 - \left(1 - e^{-\tau} \right)^k \right\} D_k(\boldsymbol{y}), \tag{4}
$$

where the additional parameter τ specifies the decreasing rate of the weights put on the higher order terms. Following [20], τ is fixed as a constant in this paper.

Shared partnership Let $EP_k(\boldsymbol{y})$ denote the number of unordered pairs (i, j) for which i and j have exactly k common neighbors and $Y_{ij} = 1$. In the literature, $EP_k(\boldsymbol{y})$ is called the edge-wise shared partnership statistic. Since $EP_k(\boldsymbol{y})$ is a function of triangles, $EP_k(\boldsymbol{y})$ is equivalent to modeling the high-order transitivities. Like GWD, the distribution of edge-wise shared partnership can be modeled as a function of single parameter by placing decreasing weights on the higher transitivities, the geometrically weighted edge-wise shared partnership (GWESP) statistic ([19]; [21]; [45]). The GWESP statistic is defined, respectively, by

$$
v(\boldsymbol{y}|\tau) = e^{\tau} \sum_{k=1}^{n-2} \left\{ 1 - \left(1 - e^{-\tau} \right)^k \right\} EP_k(\boldsymbol{y}), \tag{5}
$$

where the parameter τ specifies the decreasing rate of weights put on the higher order terms. As for the GWD statistic, τ is fixed as a constant in this paper.

Model degeneracy and prior selection As aforementioned, ERGMs can suffer from the model degeneracy problem. Schweinberger [43] showed that the model degeneracy can be caused by Markov dependent statistics, heterogeneity of degree statistics or high-order transitivity terms, and that near-degeneracy will occur for any model, regardless of what statistics are included, if parameter values become too large. Since the observed network is not degenerated, it is natural to put our emphasis on the non-degeneracy region. This suggests that a data-dependent prior might be used in Bayesian analysis of ERGMs in order to restrict the parameter space to the non-degeneracy region. See Section 4 for an illustrative example. However, we note that using a data dependent procedure to select a prior would not be the first choice (so-called empirical Bayes aside) but seems required here to deal with the difficult problems of instability and near degeneracy.

Besides models for graphs, the near degeneracy problem has been encountered in some binary random field models. For example, the 2-D Ising model has the famous phase transition behavior; the system undergoes a disorder-to-order transition at the critical temperature. Similar behaviors are also found for the autologistic model [15] and some Markov random field models [24]. It is interesting to point out that the spatial modelers and the graph modelers have different views on this problem: The former see it as primarily one of parameter space difficulties, while the latter see it as primarily one of statistics included.

3. ADAPTIVE EXCHANGE ALGORITHM

This section is organized as follows. In Section 3.1, we briefly review the exchange algorithm. In Section 3.2, we describe the adaptive exchange algorithm and study its convergence theory in Section 3.3.

3.1 The exchange algorithm

Let $\psi(\boldsymbol{y}|\theta) = \kappa(\theta) f(\boldsymbol{y}|\theta)$ denote the kernel of the density/mass function $f(\boldsymbol{y}|\theta)$. The exchange algorithm requires a perfect sampler for generating auxiliary social networks and can be described as follows:

1. (Proposal) Propose a candidate point θ' from a proposal distribution $q(\theta'|\theta)$.

2. (Perfect Sampling) Generate an auxiliary variable $y \sim f(y|\theta')$ using a perfect sampler [38].
3. (Exchange) Set $\theta_{t+1} = \theta'$ with the probability (6)
$$\alpha(\theta_t, \boldsymbol{x}, \theta') = \min\left\{1, \frac{\pi(\theta')\psi(\boldsymbol{y}|\theta')}{\pi(\theta_t)\psi(\boldsymbol{y}|\theta_t)} \frac{q(\theta_t|\theta')}{q(\theta'|\theta_t)} \frac{\psi(\boldsymbol{x}|\theta_t)}{\psi(\boldsymbol{x}|\theta')}\right\},$$
and set $\theta_{t+1} = \theta_t$ with probability $1 - \alpha(\theta_t, \boldsymbol{x}, \theta')$.

Since a swapping operation between $(\theta, \boldsymbol{y})$ and $(\theta', \boldsymbol{x})$ is involved, the algorithm is called the exchange algorithm.

3.2 The adaptive exchange algorithm

The AEX algorithm consists of two chains running in parallel. The first chain is auxiliary, which is run in the space $\mathcal{X}$ (of social networks) with the aim to draw samples from a family of distributions $f(\boldsymbol{x}|\theta^{(1)}), \ldots, f(\boldsymbol{x}|\theta^{(m)})$ defined on a set of pre-specified parameter points $\theta^{(1)}, \ldots, \theta^{(m)}$. The second chain is the target chain, which is run in the space Θ (of parameters) with the aim to draw samples from the target distribution $\pi(\theta|\boldsymbol{y})$. For a candidate parameter point θ', the auxiliary network $\boldsymbol{x}$ is resampled from the past samples of the auxiliary chain via an importance sampling procedure. Here we assume that the neighboring distributions $f(\boldsymbol{x}|\theta^{(i)})$'s have a reasonable overlap and the set $\{\theta^{(1)}, \ldots, \theta^{(m)}\}$ has covered the major part of the support of $\pi(\theta|\boldsymbol{y})$, for example, $\int_{C_\theta} \pi(\theta|\boldsymbol{x})d\theta > 0.9999$ and C_θ denotes the convex set formed by $\theta^{(1)}, \ldots, \theta^{(m)}$. These assumptions ensure that $\boldsymbol{x}$ will be distributed as $f(\boldsymbol{x}|\theta')$ as the number of iterations of the auxiliary chain becomes large, and thus the target chain can converge to the right distribution $\pi(\theta|\boldsymbol{y})$.

To draw samples from the family of distributions $f(\boldsymbol{z}|\theta)$, $\theta \in \{\theta^{(1)}, \ldots, \theta^{(m)}\}$, we adopt the stochastic approximation Monte Carlo (SAMC) algorithm [29]. SAMC ensures that each of the distributions, $f(\boldsymbol{z}|\theta^{(1)}), \ldots, f(\boldsymbol{z}|\theta^{(m)})$, can be drawn with a pre-specified frequency. Some other MCMC algorithms, such as the reversible jump MCMC algorithm [14], parallel tempering [11] and evolutionary Monte Carlo [31], can also be used here to draw samples from the family of distributions. When parallel tempering or evolutionary Monte Carlo is used, the temperature can be set to 1 for each distribution $f(\boldsymbol{z}|\theta^{(i)})$. To implement the SAMC algorithm, we define $\boldsymbol{p} = (p_1, \ldots, p_m)$ to be the desired sampling frequencies of the distributions $f(\boldsymbol{z}|\theta^{(1)}), \ldots, f(\boldsymbol{z}|\theta^{(m)})$, where $0 < p_i < 1$ and $\sum_{i=1}^m p_i = 1$; and specify a positive, nonincreasing sequence $\{a_t\}$, which satisfies the condition (A_1):

(A_1) $\lim_{t\to\infty} a_t = 0$, $\sum_{t=1}^\infty a_t = \infty$, $\sum_{t=1}^\infty a_t^\eta < \infty$, for some $\eta \in (1, 2)$.

In this paper, we set $p_1 = \cdots = p_m = 1/m$ and

$$a_t = \frac{t_0}{\max(t_0, t)}, \quad t = 1, 2, \ldots, \tag{7}$$

for some known constant $t_0 > 1$. Let $w_t^{(i)}$ denote a weight attached to the distribution $f(\boldsymbol{z}|\theta^{(i)})$ at iteration t. In our simulations, we set the initial values $w_0^{(1)} = \cdots = w_0^{(m)} = 1$. Let $\boldsymbol{z}_t$ denote the samples generated by SAMC at iteration t, let ϑ_t denote the value of θ associated with $\boldsymbol{z}_t$, let $J(\vartheta_t)$ denote the point index of ϑ_t (i.e., $J(\vartheta_t) = i$ if $\vartheta_t = \theta^{(i)}$), let $w_t = (w_t^{(1)}, \ldots, w_t^{(m)})$ denote the weight vector learned at iteration t, and let θ_t denote the sample of θ drawn from the posterior $\pi(\theta|\boldsymbol{y})$ at iteration t. Given the above notations, one iteration of the AEX algorithm can be described as follows:

Part 1: (Auxiliary) Sample Collection via SAMC.

1. (Sampling) Choose to update ϑ_t or $\boldsymbol{z}_t$ with an equal probability.
   (a) Update ϑ_t: Select ϑ' from the set $\{\theta^{(1)}, \ldots, \theta^{(m)}\}$ according to a proposal distribution $T(\cdot|\vartheta_t)$; and set $(\vartheta_{t+1}, \boldsymbol{z}_{t+1}) = (\vartheta', \boldsymbol{z}_t)$ with probability
   $$\min\left\{1, \frac{w_t^{(J(\vartheta_t))}}{w_t^{(J(\vartheta_t'))}} \frac{\psi(\boldsymbol{z}_t|\vartheta')}{\psi(\boldsymbol{z}_t|\vartheta_t)} \frac{T(\vartheta_t|\vartheta')}{T(\vartheta'|\vartheta_t)}\right\}, \tag{8}$$
   and set $(\vartheta_{t+1}, \boldsymbol{z}_{t+1}) = (\vartheta_t, \boldsymbol{z}_t)$ with the remaining probability.
   (b) Update $\boldsymbol{z}_t$: Draw $\boldsymbol{z}_{t+1} \sim f(\cdot|\vartheta_t)$ via one or a few MH updates starting with the current sample $\boldsymbol{z}_t$, and set $\vartheta_{t+1} = \vartheta_t$.
2. (Weight updating) Set
$$\log(w_{t+1}^{(i)}) = \log(w_t^{(i)}) + a_{t+1}(e_{t+1,i} - p_i), \quad i = 1, 2, \ldots, m,$$
where $e_{t+1,i} = 1$ if $\vartheta_{t+1} = \theta^{(i)}$ and 0 otherwise.
3. (Sample Collection) Append the sample $(\boldsymbol{z}_{t+1}, \theta^{(j)}, w_{t+1}^{(j)})$, where $j = J(\vartheta_{t+1})$, to the collection $\boldsymbol{S}_t$. Denote the new collection by $\boldsymbol{S}_{t+1}$.

Part 2: (Target) Adaptive Exchange.

4. (Proposal) Propose a candidate point θ' from the proposal distribution $q(\theta'|\theta_t)$.
5. (Resampling) Resample an auxiliary variable $\boldsymbol{x}$ from the collection $\boldsymbol{S}_{t+1}$ via an importance sampling procedure; that is, setting $\boldsymbol{x} = \boldsymbol{z}_i$ with probability (9)
$$P(\boldsymbol{x} = \boldsymbol{z}_i) = \frac{\sum_{j=1}^{t+1} \omega_j \psi(\boldsymbol{z}_j|\theta')/\psi(\boldsymbol{z}_j|\vartheta_j) I(\boldsymbol{z}_j = \boldsymbol{z}_i)}{\sum_{j=1}^{t+1} \omega_j \psi(\boldsymbol{z}_j|\theta')/\psi(\boldsymbol{z}_j|\vartheta_j)},$$
where $(\boldsymbol{z}_j, \vartheta_j, \omega_j)$ denotes the j-th element of the set $\boldsymbol{S}_{t+1}$. Note that $\vartheta_j \in \{\theta^{(1)}, \ldots, \theta^{(m)}\}$.
6. (Exchange) Set $\theta_{t+1} = \theta'$ with the probability (10)
$$\alpha(\theta_t, \boldsymbol{x}, \theta') = \min\left\{1, \frac{\pi(\theta')\psi(\boldsymbol{y}|\theta')}{\pi(\theta_t)\psi(\boldsymbol{y}|\theta_t)} \frac{q(\theta_t|\theta')}{q(\theta'|\theta_t)} \frac{\psi(\boldsymbol{x}|\theta_t)}{\psi(\boldsymbol{x}|\theta')}\right\},$$
and set $\theta_{t+1} = \theta_t$ with probability $1 - \alpha(\theta_t, \boldsymbol{x}, \theta')$.

On this algorithm, we have the following remarks:

- The proposal $q(\cdot|\cdot)$ can depend on $\boldsymbol{y}$; that is, it can be written in the form $q(\cdot|\cdot, \boldsymbol{y})$. For notational simplicity, we depress the dependence of q on $\boldsymbol{y}$. For ERGMs, we have the auxiliary network updated (in step 1(b) of the auxiliary chain) by a sweep of the Gibbs sampler at each iteration.
- On the choice of auxiliary parameter points $\{\theta^{(1)}, \ldots, \theta^{(m)}\}$. In general, $\theta^{(i)}$'s can be chosen according to some approximate quantiles of $\pi(\theta|\boldsymbol{y})$, say, some samples by the DMH sampler. In this paper, we propose to choose the auxiliary parameter points for ERGMs based on ABC samples. This will be described in Section 4.
- On the choice of $\{a_t\}$ and convergence of AEX. In this paper, we set the gain factor in the form (7) with a free parameter t_0. As discussed in [29], a large value of t_0 will force the sampler to reach all distributions $f(\boldsymbol{z}|\theta^{(i)})$'s quickly. Therefore, t_0 should be set to a large value for a complex problem. In this paper, t_0 is set to 20,000 for all examples.
  In general, the choice of t_0 should be associated with the choice of N, total number of iterations of a run. The appropriateness of these choices can be diagnosed by checking the convergence of the auxiliary and target chains. The convergence of the auxiliary chain can be checked through an examination for the realized sampling frequencies $(\hat{p}_1, \ldots, \hat{p}_m)$, where $\hat{p}_i$ denotes the realized sampling frequency from the distribution $f(\boldsymbol{z}|\theta^{(i)})$. If $(\hat{p}_1, \ldots, \hat{p}_m)$ is not close to $(p_1, \ldots, p_m)$, the auxiliary chain should be diagnosed as non-converged. In this case, the algorithm should be re-run with a larger value of N or a larger value of t_0 or both. Note that for the convergence diagnosis of the auxiliary chain, multiple runs are not necessary, as it is known that each of the distributions $f(\boldsymbol{z}|\theta^{(i)})$'s is valid. However, to check the convergence of the target chain, multiple runs are still necessary.

3.3 Convergence of the adaptive exchange algorithm

Liang et al. [28] studied the convergence of the AEX algorithm by treating it as an adaptive MCMC algorithm, as for which the proposal distribution of generating auxiliary networks is changed from iteration to iteration. To establish its convergence under the framework of adaptive MCMC, restrictive conditions, such as both $\mathcal{X}$ and Θ are compact, are assumed. However, taking a closer look at the AEX algorithm, it is easy to see that AEX is different from conventional adaptive MCMC algorithms. In conventional adaptive MCMC algorithms, see, e.g., [16], the proposal changes from iteration to iteration but also dependent on the past samples of the chain. However, AEX is different, for which the proposal changes from iteration to iteration but is independent of past samples. This motivates us to develop some new theory for its convergence. Below we establish the convergence for the AEX algorithm under relaxed conditions, in particular, Θ is no longer required to be compact.

Lemma 3.1 concerns the weak convergence of auxiliary networks, and it is a partial restatement of Lemma 3.1 of [28]. To make the paper self-contained, a proof of this lemma is given in Appendix B.

Lemma 3.1. *Assume (i) both $\mathcal{W}$ (the space of w_t) and $\mathcal{X}$ (the space of $\boldsymbol{x}$) are compact; (ii) $f(\boldsymbol{x}|\theta)$ is bounded away from 0 and ∞ on $\Theta \times \mathcal{X}$; and (iii) the conditions (A_1) and (A_2) (given in Appendix A) are satisfied. Let $\{\boldsymbol{z}_1, \theta^{(J_1)}, w_1^{(J_1)}; \ldots; \boldsymbol{z}_N, \theta^{(J_N)}, w_N^{(J_N)}\}$ denote a set of samples generated by SAMC in an AEX run, where $J_t \in \{1, 2, \ldots, m\}$ for $t = 1, \ldots, N$. Let $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ be distinct samples in $\{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N\}$. Resample a random variable/vector $\boldsymbol{X}$ from $\{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N\}$ such that*

(11)
$$P(\boldsymbol{X} = \boldsymbol{x}_k|\theta') = \frac{\sum_{t=1}^{N}\sum_{i=1}^{m}\left\{w_t^{(i)}\frac{\psi(\boldsymbol{z}_t|\theta')}{\psi(\boldsymbol{z}_t|\theta^{(i)})}I(J_t = i \text{ and } \boldsymbol{z}_t = \boldsymbol{x}_k)\right\}}{\sum_{t=1}^{N}\sum_{i=1}^{m}\left\{w_t^{(i)}\frac{\psi(\boldsymbol{z}_t|\theta')}{\psi(\boldsymbol{z}_t|\theta^{(i)})}I(J_t = i)\right\}},$$
$$k = 1, \ldots, n,$$

then the distribution of $\boldsymbol{X}$ converges to $f(\cdot|\theta')$ as $N \to \infty$.

Andrieu et al. [1] considered the convergence of a varying truncation version of the stochastic approximation MCMC algorithm, which can be view as a more general version of SAMC. Under mild conditions, they showed that the varying truncation of w_t can only occur a finite number of times. Thus, it is reasonable to assume that w_t can be kept in a compact set during simulations. In this paper, following [29], we set $\mathcal{W}$ to $[1/B_w, B_w]^m$ with B_w being a huge number, say, 10^{100}, which, as a practical matter, is equivalent to setting $\mathcal{W} = \mathbb{R}^+$. Regarding other conditions of Lemma 3.1, we note that the condition A_2 can be verified as in [29]. Given the compactness of $\mathcal{X}$, which is true for social networks, a sufficient condition for satisfying A_2 is to choose the proposal distribution $q(\boldsymbol{x}, \boldsymbol{y})$ satisfying the local positive condition:

For every $\boldsymbol{x} \in \mathcal{X}$, there exist $\epsilon_1 > 0$ and $\epsilon_2 > 0$ such that $|\boldsymbol{x} - \boldsymbol{y}| \le \epsilon_1 \Longrightarrow q(\boldsymbol{x}, \boldsymbol{y}) \ge \epsilon_2$.

Obviously, the Gibbs sampler satisfies this condition. The tie-no-tie sampler [22], which will be described in Section 5.1, also satisfies this condition.

For the AEX algorithm, $\{\theta_t\}$ forms an adaptive Markov chain with the transition kernel given by

(12)
$$\tilde{P}_l(\theta, d\theta') = \int_{\mathcal{X}} \alpha(\theta, \boldsymbol{x}, \theta')q(\theta, d\vartheta)\nu_l(d\boldsymbol{x}|\theta') + \delta_\theta(d\theta')\rho(\theta),$$

where $\rho(\theta) = 1 - \int_{\Theta\times\mathcal{X}} \alpha(\theta, \boldsymbol{x}, \vartheta')q(\theta, d\vartheta')\nu_l(d\boldsymbol{x}|\theta')$ denotes the mean rejection probability at θ, $\alpha(\theta, \boldsymbol{x}, \theta')$ is defined in (10), l denotes the cardinality of the set of auxiliary networks collected from the auxiliary Markov chain, i.e., $l = |\boldsymbol{S}_l|$,

and $\nu_l(\boldsymbol{x}|\theta')$ denotes the true distribution of $\boldsymbol{x}$ resampled from $\boldsymbol{S}_l$. Since $\alpha(\theta, \boldsymbol{x}, \theta')$, $q(\theta, \vartheta)$ and $\nu_l(\boldsymbol{x}|\theta')$ are all upper bounded, it follows from Lemma 3.1 and Lebesgue's dominated convergence theorem that for any $(\theta, \theta') \in \Theta \times \Theta$,

$$\tilde{P}_l(\theta, d\theta') \to P(\theta, d\theta'), \quad \text{as } l \to \infty, \tag{13}$$

where $P(\theta, d\theta')$ denotes the transition kernel of the exchange algorithm with perfect auxiliary networks; that is,

$$P(\theta, d\theta') = \int_{\mathcal{X}} \alpha(\theta, \boldsymbol{x}, \theta') q(\theta, d\vartheta) f(d\boldsymbol{x}|\theta') + \delta_\theta(d\theta')\big[1 - \int_{\Theta \times \mathcal{X}} \alpha(\theta, \vartheta') q(\theta, d\vartheta') f(d\boldsymbol{x}|\theta')\big]. \tag{14}$$

It is known that $P(\theta, d\theta')$ can induce a Markov chain which is irreducible, aperiodic and admits $\pi(\theta|\boldsymbol{y})$ as the invariant distribution, provided an appropriate proposal $q(\cdot, \cdot)$ has been used therein.

Define

$$\beta_l(\theta, \boldsymbol{x}, \theta') = \frac{\nu_l(\boldsymbol{x}|\theta')}{\nu_l(\boldsymbol{x}|\theta)} \frac{f(\boldsymbol{x}|\theta)}{f(\boldsymbol{x}|\theta')},$$
$$r(\theta, \boldsymbol{x}, \theta') = \frac{\pi(\theta') f(\boldsymbol{y}|\theta')}{\pi(\theta) f(\boldsymbol{y}|\theta)} \frac{q(\theta|\theta')}{q(\theta'|\theta)} \frac{f(\boldsymbol{x}|\theta)}{f(\boldsymbol{x}|\theta')},$$
$$r_v(\theta, \boldsymbol{x}, \theta') = \frac{\pi(\theta') f(\boldsymbol{y}|\theta')}{\pi(\theta) f(\boldsymbol{y}|\theta)} \frac{q(\theta|\theta')}{q(\theta'|\theta)} \frac{\nu_l(\boldsymbol{x}|\theta)}{\nu_l(\boldsymbol{x}|\theta')},$$

which implies

$$r(\theta, \boldsymbol{x}, \theta') = \beta_l(\theta, \boldsymbol{x}, \theta') r_v(\theta, \boldsymbol{x}, \theta').$$

Also, the Markov chain defined by the acceptance rule $\min\{1, r_v(\theta, \boldsymbol{x}, \theta')\}$ is irreducible, aperiodic and admits the invariant distribution $\pi(\theta|\boldsymbol{y})$, provided the Markov chain induced by the transition kernel P is irreducible, aperiodic and admits the invariant distribution $\pi(\theta|\boldsymbol{y})$. To ensure the convergence of this Markov chain, $\nu_l(\boldsymbol{x}|\theta)$ is not necessarily to have a support as large as $\mathcal{X}$. In fact, its support can be only a subset of $\mathcal{X}$. Let $P_v(\theta, \theta')$ denote the transitional kernel of the Markov chain induced by the acceptance rule $\min\{1, r_v(\theta, \boldsymbol{x}, \theta')\}$. Lemma 3.2 shows that $\tilde{P}_l(\theta, \theta')$ is also irreducible and aperiodic and admits an invariant distribution, whose proof can be found in Appendix B.

Lemma 3.2. *Assume that (i) $\mathcal{X}$ is compact; (ii) $f(\boldsymbol{x}|\theta)$ is bounded away from 0 and ∞ on $\Theta \times \mathcal{X}$; and (iii) P is irreducible and aperiodic and admits an invariant distribution. Then for any $l \in \mathbb{N}$ such that for any $\theta \in \Theta$, $\rho(\theta) > 0$, $\tilde{P}_l$ is also irreducible and aperiodic, and hence there exists a stationary distribution $\tilde{\pi}_l(\theta|x)$ such that for any $\theta_0 \in \Theta$,*

$$\lim_{k \to \infty} \|\tilde{P}_l^k(\theta_0, \cdot) - \tilde{\pi}_l(\cdot|\boldsymbol{y})\| = 0.$$

Theorem 3.1 concerns the convergence of the AEX algorithm, whose proof can be found in Appendix B.

Theorem 3.1. *Assume the conditions of Lemma 3.1 and Lemma 3.2 hold. Then for any $\varepsilon > 0$, there exist $L(\varepsilon, \theta_0) \in \mathbb{N}$ and $K(\varepsilon, \theta_0, l) \in \mathbb{N}$ such that for any $l > L(\varepsilon, \theta_0)$ and $k > K(\varepsilon, \theta_0, l)$*

$$\|\tilde{P}_l^k(\theta_0, \cdot) - \pi(\cdot|\boldsymbol{y})\| \le \varepsilon. \tag{15}$$

This theorem implies that as the number of iterations of both the auxiliary and target chains goes to infinity, the samples drawn by the target chain will converge weakly toward the posterior $\pi(\theta|\boldsymbol{y})$. By standard MCMC theory (see, e.g., [47]), for any integrable function $h(\theta)$, the path averaging estimator $\sum_{k=1}^n h(\theta_k)/n$ will converge to its posterior mean almost surely; that is, as $l \to \infty$ and $k \to \infty$,

$$\frac{1}{n} \sum_{k=1}^n h(\theta_k) \to \int h(\theta) \pi(\theta|\boldsymbol{y}) d\theta, \quad a.s.,$$

provided that $\int |h(\theta)| \pi(\theta|\boldsymbol{y}) d\theta < \infty$.

4. AUXILIARY PARAMETER POINTS SELECTION

As discussed in Section 2, an ERGM may suffer from the model degeneracy problem if it includes a Markov dependence statistic and/or geometrically weighted statistics. When the model degeneracy happens, the network tends to be either empty or complete. Since the observed network is usually neither empty nor complete, it is natural to put our emphasis on the non-degeneracy region when conducting a Bayesian analysis for the ERGM. To illustrate the model degeneracy and non-degeneracy region to the graphics, we explore the parameter space of an ERGM with only two statistics, the edge count and k_2-star, for a social network with 16 nodes. It is known that this model can be degenerate in some regions, as k_2-star is included.

Figure 2 shows the degeneracy and non-degeneracy regions of this model. To produce this plot, the Gibbs sampler was run 5 times independently at each grid point of a 250×100-lattice defined on the region $[-3.5, -1.0] \times [-0.5, 0.5]$. Each run of the Gibbs sampler consisted of 100,000 iterations, and thus a total of 50,000 networks were simulated at each grid point. If any of the 50,000 networks had edge counts less than 5 or greater than 100, the point was classified as a degenerate point. As shown in Figure 2, the non-degeneracy region of an ERGM can be irregular. This suggests that a lattice-based auxiliary parameter points selection method, which is used in [28] for autologistic models, may not work well for ERGMs. For this reason, we propose to use an ABC ([5]; [33]) based method for selecting auxiliary parameter points. This method can be described as follows.

Let $\mathcal{S}(\boldsymbol{y})$ denote the set of statistics included in the ERGM. Let $\boldsymbol{x}_t$ and θ_t denote, respectively, the simulated network and parameter at iteration t. Let u be a counter of rejections. One iteration of the ABC-based method consists of the following steps:

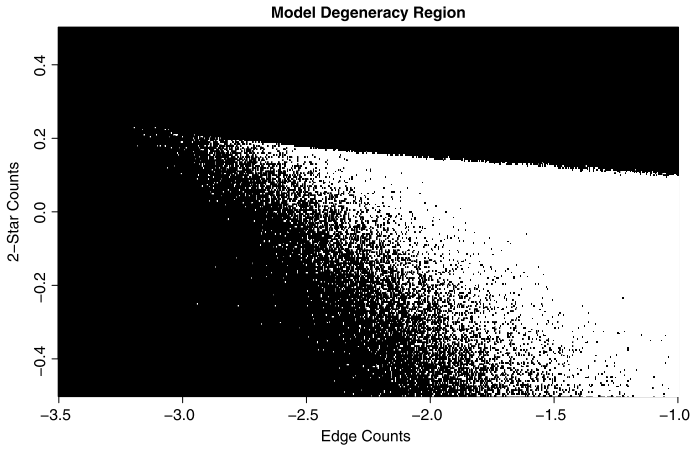

Figure 2. Degeneracy (black) and non-degeneracy (white) regions of an ERGM for a social network with 16 nodes.

ABC Algorithm.

(a) Propose a candidate parameter value θ'.
(b) Simulate a network $\boldsymbol{x}'$ from $f(\boldsymbol{x}|\theta')$ using the Gibbs sampler starting with $\boldsymbol{x}_t$.
(c) If $d(\mathcal{S}(\boldsymbol{y}), \mathcal{S}(\boldsymbol{x}')) \leq \boldsymbol{\epsilon}$, then accept $(\boldsymbol{x}', \theta')$ with probability $\min\{1, \frac{q(\theta',\theta)\pi(\theta')}{q(\theta,\theta')\pi(\theta)}\}$, where $d(\cdot,\cdot)$ denotes a distance measure between $\mathcal{S}(\boldsymbol{y})$ and $\mathcal{S}(\boldsymbol{x}')$. If it is accepted, set $\theta_{t+1} = \theta'$, $\boldsymbol{x}_{t+1} = \boldsymbol{x}'$ and $u = 0$. Otherwise, set $u = u + 1$.

We find that the simulation can get stuck sometimes. To prevent the ABC chain from clinging to certain locations, ABC will be reinitialized when u exceeds a threshold value. The threshold value was set to 100 throughout this paper. For each example, ABC was run for 10,000 iterations. After thinning by a factor of 200, 50 samples of θ were obtained, which were then used as the auxiliary parameter points in AEX. A proper choice for the number of auxiliary parameter points may be disputable, but our experience suggests that 50 auxiliary parameter points are sufficient for most ERGMs.

Of course, the ABC method is not the only choice for the pilot exploration of parameter space. Some other cheap MCMC algorithms, such as the DMH sampler, can also be applied. However, compared to DMH, the ABC-based method has one advantage that the auxiliary parameter points can be easily made more dispersed by choosing a larger value of $\boldsymbol{\epsilon}$.

5. NUMERICAL EXAMPLES

In this section, we illustrate the performance of AEX using three examples, the Florentine business network, molecule synthetic network, and dolphin network, which are shown in Figure 3.

5.1 Florentine business network

This network was collected by [37] from historic documents. It represents a set of business ties, such as loans, credits and joint partnerships, among families in Renaissance Florence, Italy. This network consists of 16 families who were locked in a struggle for political control of the city of Florence around 1430.

Since the size of this network is fairly small and this network shows that few edges are present between the families with quite a high level of two-star formulation, we considered the 2-dimensional model with edge count and k_2-star, leading to the following likelihood function

$$f(\boldsymbol{y}|\theta) = \frac{1}{\kappa(\theta)} \exp\{\theta_1 S_1(\boldsymbol{y}) + \theta_2 S_2(\boldsymbol{y})\}, \tag{16}$$

where $S_1(\boldsymbol{y})$ is the edge counts and $S_2(\boldsymbol{y})$ is the 2-star count. This network is a famous pedagogical example for showing difficulties in the parameter estimation of the ERGMs under the model degeneracy [6].

We adopted the following priors:

$$\theta_1 \sim \text{Uniform}(-4, 0), \quad \theta_2 \sim \text{Uniform}(0, 8), \tag{17}$$

based on two considerations. Firstly, the MPLE of this model, which is $(-3.39, 0.35)$ with the standard error $(0.70, 0.14)$, indicates that θ_1 may take a negative value and θ_2 may take a positive value. Secondly, our pilot parameter space exploration, as shown in Figure 2, indicates that $(-4, 0)$ should have been wide enough for θ_1, as the region with $\theta_1 < -4$ is degenerate for the model. As suggested by the MPLE, we restrict θ_2 to be positive. Setting a wider prior

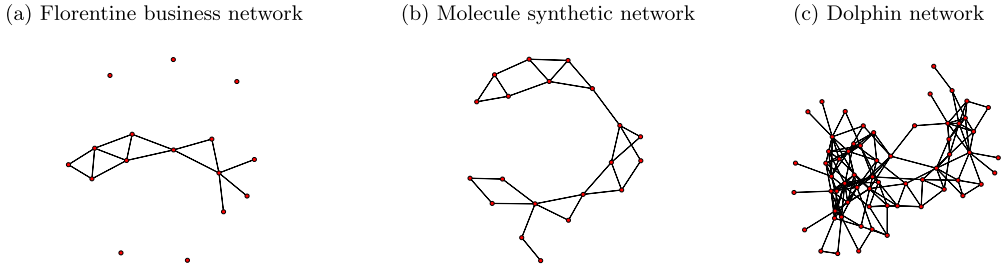

Figure 3. Visualization of three example networks.

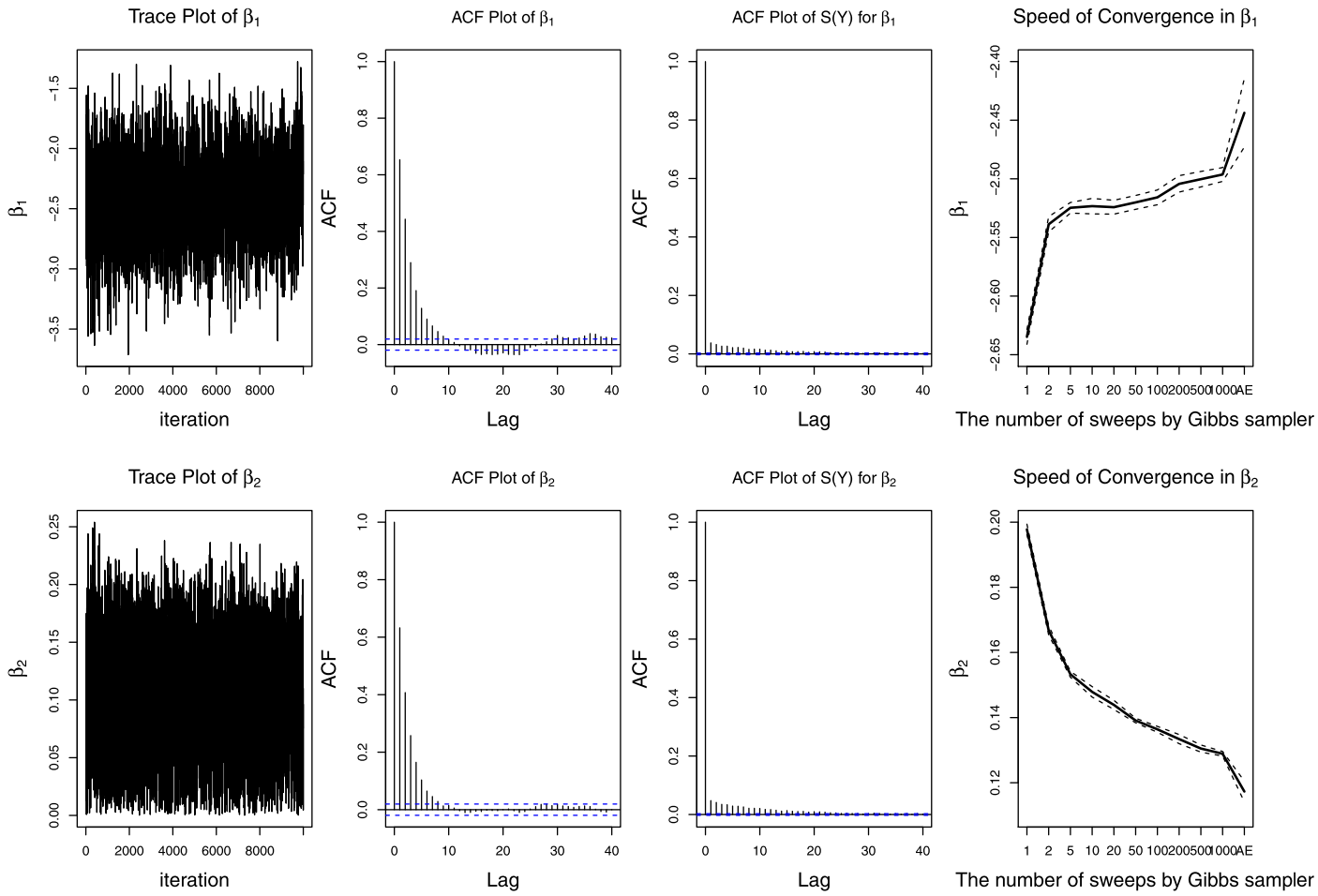

Figure 4. From the left to right: Column 1 shows the trace plots of the AEX samples of θ; column 2 shows the autocorrelation plots of the AEX samples of θ; column 3 shows the autocorrelation plots of sufficient statistics of the auxiliary networks resampled from the auxiliary chain; and column 4 shows the changing pattern of the plain-DMH estimates with the value of Π*, the number of sweeps of Gibbs sampler used for generating an auxiliary network.*

for θ_2, e.g., Uniform$(-100, 100)$, should not change the performance of AEX, but needing more auxiliary parameter points and thus longer CPU time. For the same reason, we have also restricted the parameter space of other examples to a relatively small region after a pilot exploration of the model through MPLE or DMH. If DMH is used, a wide Gaussian prior can be specified at this stage.

The AEX algorithm was first applied to this example. To select auxiliary parameter points, the ABC algorithm was run with $\boldsymbol{\epsilon} = (10, 20)$, where 10 denotes the tolerance limit for the edge count and 20 for k_2-star. AEX was run 10 times independently. Each run consisted of 160,000 iterations. For the first 100,000 iterations, only the auxiliary chain was run and a database of auxiliary networks were collected therefrom. Then the two chains were run simultaneously for 60,000 iterations and the samples generated from the auxiliary chain were continuously collected and added into the database. For the target chain, the first 10,000 iterations were discarded for the burn-in process, and 10,000 samples were collected from the remaining 50,000 iterations at a time space of 5 iterations. The same running schedule was also applied to other two examples of this paper. The resulting parameter estimates were reported in Table 1 and can be interpreted as follows: if neither of the nodes i and j is connected to some other nodes, then the log-odds for them becoming connected is -2.4322; and if either node i or node j is connected to some other nodes, then the log-odds for them becoming connected rises to -2.2911 $(= -2.4322 + 0.1141)$. The parameter estimates also show that the probability of adding edges is smaller than those of deleting edges, but there is a modest propensity to completion of a 2-star once an edge is formed.

To have an intuitive assessment for the convergence of the target chains, we drew the trace and autocorrelation plots of the samples of θ. Figure 4 shows that the target chain can mix reasonably fast, and an independent sample of θ can be obtained in about 50 iterations. This converts to about 1,000 independent samples generated in each run, and thus

*Table 1. Parameter estimation for the Florentine business network. The estimates were calculated by averaging over 10 independent runs with the standard Monte Carlo errors reported in the parentheses. CPU(s): The CPU time (in seconds) cost by a single run on a personal computer with a quad-core i7 2.2GHz processor. * The results of CF-DMH are from [6], where the standard errors were calculated based on 5 parallel chains, and the CPU time had been adjusted to our computer by re-running their codes under the same setting as given in the paper*

Method	Edge Counts	k2-Star	CPU(s)
plain-DMH ($\Pi = 1$)	-2.6348 (2.8e-3)	0.1978 (7e-4)	0.8
plain-DMH ($\Pi = 20$)	-2.5242 (2.6e-3)	0.1439 (6e-4)	9.2
plain-DMH ($\Pi = 200$)	-2.5043 (3.1e-3)	0.1334 (6e-4)	84.8
plain-DMH ($\Pi = 1000$)	-2.4963 (2.6e-3)	0.1289 (3e-4)	434.7
CF-DMH*	-2.44 (1.6e-2)	0.12 (4.5e-3)	47.7
AEX	-2.4322 (8.4e-3)	0.1141 (1.0e-3)	≈ 360

the AEX estimates reported in Table 1 are reliable. In addition, column 3 of Figure 4 shows the autocorrelation plots of the sufficient statistics of the auxiliary networks resampled from the auxiliary chain, and it indicates that those auxiliary networks are nearly independent. This further implies the validity of the AEX sampler.

Since the perfect sampler is not available for ERGMs, the DMH sampler was applied to this example for the purpose of comparison. A plain version of the DMH sampler is the same as part 2 of the AEX algorithm except that the auxiliary network $\boldsymbol{x}$ is simulated at each iteration via a short run (Π sweeps) of the Gibbs sampler initialized at the observed network $\boldsymbol{y}$. For this example, we tried four different values of Π, 1, 20, 200 and 1,000. The plain-DMH sampler should represent a fair comparison with the AEX algorithm, as they both employ the Gibbs sampler for simulating auxiliary networks. Meanwhile, we also compared AEX with the CF version of DMH [6]. The CF-DMH sampler is different from the plain-DMH sampler in two respects: Firstly, it employs the tie-no-tie sampler [22] for simulating auxiliary networks. At each iteration of the tie-no-tie sampler, it first selects with an equal probability the set of edges or the set of empty dyads, and then swaps a dyad at random within that chosen set. Since most of the realistic networks are sparse, the tie-no-tie sampler does not waste too much time proposing new edges which are likely to be rejected, and thus it can converge generally faster than the Gibbs sampler. Secondly, it employs the adaptive direction sampler ([12]; [39]), for updating θ. The adaptive direction sampler is a population MCMC algorithm, which makes use of distributed information of individual samples of the population and thus can generally converge faster than the single chain MCMC approach. Refer to [30] for more discussions on this issue.

As shown in Table 1, the performance of the plain-DMH sampler can heavily depend on the value of Π. For each value of Π given in Table 1, the plain-DMH sampler was run 10 times independently. Each run consisted of 60,000 iterations, where the first 10,000 iterations were discarded for the burn-in process and the samples of θ were collected from the remaining ones at every 5th iteration. As Π becomes larger and larger, the DMH estimates are closer and closer to the AEX estimate. Column 4 of Figure 4 shows this pattern. However, even when Π is equal to 1,000, the DMH estimate still cannot reach the AEX estimate. This experiment is consistent with the theory of [43] that a MCMC sampler for (1) can converge slowly if the model suffers from the degeneracy problem. The results from [6] indicate that an efficient MCMC sampler is essential for the success of the DMH sampler. However, even the CF-DMH sampler has equipped with two advanced MCMC techniques, as shown in Table 1, it seems that its estimate has still a little gap to the AEX estimate. The success of AEX is due to its efficiency in drawing auxiliary social networks via running a long Markov chain. As shown in column 3 of Figure 4, the auxiliary networks resampled from the auxiliary chain are nearly independent.

Regarding the comparison with CF-DMH, we note that the estimates were from [6], and the standard errors were calculated based on the results of parallel chains. The CPU time had been adjusted to our computer by re-running their codes under the same setting as given in [6]. Strictly speaking, the CF-DMH estimate is not directly comparable with ours, as different priors were used for them. In addition, due to the convergence issue that DMH suffered from, potentially, the CF-DMH estimate can be biased. However, the CF-DMH estimate does provide us a good reference: It indicates the validity of AEX. This note is also applicable to the other two examples of this paper.

To further assess the accuracy of the AEX, plain-DMH and CF-DMH estimates presented in Table 1, we evaluate the root mean squared error (RMSE) of the estimates of $S_a(\boldsymbol{y})$'s based on the idea of parametric bootstrap [8]. Since the statistics $\{S_a(\boldsymbol{y}) : a \in \mathcal{A}\}$ are sufficient for θ, $S_a(\boldsymbol{y})$'s can be reversely estimated by the simulated networks from the distribution $f(\boldsymbol{y}|\hat{\theta})$, where $\hat{\theta}$ denotes an estimate of θ. RMSE can be calculated as follows:

(a) For a given estimate $\hat{\theta}$, simulate K networks $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_K$, independently using the Gibbs sampler.
(b) Calculate the statistics

$$S_a(\boldsymbol{y}_i), a \in \mathcal{A} \quad \text{for} \quad i = 1, 2, \ldots, K.$$

Table 2. Root mean square errors (RMSE) of the AEX, plain-DMH and CF-DMH estimates for the Florentine business network

	plain-DMH					
Statistic	$\Pi = 1$	$\Pi = 20$	$\Pi = 200$	$\Pi = 1000$	CF-DMH	AEX
Edge Count	13.218	4.770	4.645	4.651	4.515	4.462
K_2-star	112.825	20.963	20.405	20.488	20.040	19.886

(c) Calculate

$$RMSE(S_a) = \sqrt{\sum_{i=1}^{K} \{S_a(\boldsymbol{y}_i) - S_a(\boldsymbol{y})\}^2 / K}$$

for all $a \in \mathcal{A}$.

Obviously, a smaller value of RMSE implies a more accurate estimate of θ. Table 2 reports respective RMSE values for the estimates presented in Table 1, and it shows a strictly decreasing pattern from left to right. This indicates that AEX provides a more accurate parameter estimate than the plain-DMH and CF-DMH samplers for this example.

To assess accuracy of the parameter estimates in a graphical way, we used the goodness-of-fit (GOF) plots [20]. The GOF plot shows the distribution (through box-plots and confidence intervals) of three sets of statistics, the degree distribution, the edgewise shared partnership distribution and the geodesic distance distribution, for the fitted model. If the statistics of the observed network, which are represented by a solid line in the GOF plots, falls into the confidence intervals of the fitted model, then the fit is considered good. The closer the solid line is to the center of the box-plots, the better the fit is. Figure 5 in Appendix C indicates that AEX provides a slightly better fit (in the plots of column 2) for the network than the plain-DMH sampler with $\Pi = 200$. The GOF plot for the CF-DMH estimate is similar.

Finally, to assess the effect of the choice of m, the number of auxiliary parameter points, on the stationary distribution $\pi(\theta|\boldsymbol{y})$, we compared the covariance matrix of $\pi(\theta|\boldsymbol{y})$ for different values of $m = 50$, 100, and 200. For each value of m, AEX was run 50 times independently, but the number of iterations of each run varied with the value of m. For $m = 50$, the total number of iterations was set to $160,000$, with the first $N_1 = 100,000$ iterations being solely used for auxiliary network collection, the next $N_2 = 10,000$ iterations being discarded for the burn-in process of the target chain, and the samples of θ being collected at every 250th iteration from the remaining $N_3 = 50,000$ iterations. Recall that during the last 60,000 iterations, the auxiliary and target chains were run simultaneously. For $m = 100$ and 200, only the setting of N_1 was changed, while the settings of N_2 and N_3 were kept the same. We set $N_1 = 200,000$ for $m = 100$ and $N_1 = 400,000$ for $m = 200$. The results were summarized in Table 3, which implies that different settings of m can lead to almost identical posterior estimates as long as they are large enough.

Table 3. Effects of the number of auxiliary parameter points on the stationary distribution $\pi(\theta|\boldsymbol{y})$. $\sigma^2_{\theta_1}$: the variance of θ_1; $\sigma^2_{\theta_2}$: the variance of θ_2; and $\sigma_{\alpha\beta}$: the covariance of θ_1 and θ_2

Settings	$\sigma^2_{\theta_1}$	$\sigma^2_{\theta_2}$	$\sigma_{\alpha\beta}$
$m = 50$	1.2×10^{-1}	2.8×10^{-3}	-1.4×10^{-2}
$m = 100$	1.2×10^{-1}	2.8×10^{-3}	-1.4×10^{-2}
$m = 200$	1.1×10^{-1}	2.8×10^{-3}	-1.4×10^{-2}

5.2 Molecule synthetic network

This network, shown in Figure 3(b), is obtained from the `ergm` package [22]. It consists of 20 nodes and resembles the chemical structure of a molecule. In this molecule synthetic network, every node has at least one other node connected, and there exist quite high levels of k-stars and triangles. To reflect the structure of this network, we consider a 4-dimensional ERGM model with edge counts, 2-star, 3-star, and triangle. This model also shows model degeneracy which causes difficulties for parameter estimation. The likelihood function of the model is given by

(18)
$$f(\boldsymbol{y}|\theta) = \frac{1}{\kappa(\theta)} \exp\{\theta_1 S_1(\boldsymbol{y}) + \theta_2 S_2(\boldsymbol{y}) + \theta_3 S_3(\boldsymbol{y}) + \theta_4 T(\boldsymbol{y})\},$$

where $S_1(\boldsymbol{y})$ is the edge count, $S_2(\boldsymbol{y})$ is k_2-star, $S_3(\boldsymbol{y})$ is k_3-star, and $T(\boldsymbol{y})$ is the triangle count.

The parameters are subject to the following priors:

(19)
$$\begin{aligned} &\theta_1 \sim \text{Unif}(0, 8), \quad \theta_2 \sim \text{Unif}(-8, 0), \\ &\theta_3 \sim \text{Unif}(-8, 0), \quad \theta_4 \sim \text{Unif}(0, 3), \end{aligned}$$

which constraints the parameters to be either positive or negative values based on our pilot exploration of the model through a short run of the plain-DMH sampler with $\Pi = 1$ and a wide Gaussian prior. As we mentioned in Section 5.1, we can improve the computational efficiency of the AEX algorithm by restricting the parameter spaces to a relatively small region after a pilot study.

AEX was applied to this example. To implement the ABC-based method for auxiliary parameter points selection, we set $\boldsymbol{\epsilon} = (10, 40)$, where the first component corresponds to the edge count and the second component corresponds to k_2-star. No constraints were set for k_3-star and triangle counts. AEX was run 10 times independently. Each run

*Table 4. Parameter estimation for the molecule synthetic network. The AEX estimates were calculated by averaging over 10 independent runs with standard Monte Carlo errors reported in the parentheses. CPU(s): The CPU time (in seconds) cost by a single run on a personal computer with a quad-core i7 2.2GHz processor. * The results of CF-DMH are from Caimo and Friel (2011), where the standard errors were calculated based on 8 parallel chains, and the CPU time had been adjusted to our computer*

Method	Edge Counts	k2-Star	k3-star	triangle	CPU(s)
AEX	1.93 (6.7e-2)	-0.71 (1.0e-2)	-0.25 (9.7e-3)	1.60 (2.4e-2)	95.4
CF-DMH	2.72 (3.9e-2)	-1.02 (1.3e-2)	-0.05 (1.2e-2)	1.60 (2.6e-2)	≈ 540

Table 5. Root mean square errors of the AEX and CF-DMH estimates for molecule synthetic network example

Methods	Edge Count	k_2-star	k_3-star	Triangle
AEX	2.181	10.555	10.574	2.458
CF-DMF	2.234	10.883	10.642	2.460

consisted of 160,000 iterations with the same iteration setting for the auxiliary and target chains as for the Florentine business network example. The results were summarized in Table 4. They can be interpreted as follows: if neither of the nodes i and j is connected to some other nodes, then the log-odds for them becoming connected is 1.93; if either node i or node j is connected to some other nodes, then the log-odds for them becoming connected is 1.22 ($= 1.93 - 0.71$); if one of the nodes i and j is connected to some other nodes which also forms a two-star, then the log-odds for them becoming connected is 0.97 ($= 1.93 - 0.71 - 0.25$); and if both node i and j are connected to some other nodes, then the log-odds for them becoming connected is 2.82 ($= 1.93 - 0.71 + 1.60$). The parameter estimates imply that the probability of adding edges is larger than that of deleting edges but there is no propensity to completion of 2-stars and 3-stars among edges. However, once a 2-star or 3-star is formed in the network, there is a tendency to completion of a triad. As a reference, we also gave in Table 4 the results of CF-DMH. The estimates produced by these two algorithms are not quite consistent, especially for θ_1, but the overall pattern is similar. Here we mention again that the AEX and CF-DMH estimates are not directly comparable due to the same reason as discussed in the previous example.

To assess accuracy of the AEX and CF-DMH estimates, we calculated their RMSE values with $K = 20,000$. The results are summarized in Table 5, which indicates that the AEX estimate is more accurate than the CF-DMH estimate in terms of RMSE values. Figure 6 in Appendix C shows the goodness-of-fit (GOF) plots for the two estimates, which also imply that the AEX estimate can be slightly better (in the degree column) than the CF-DMH estimate.

5.3 Dolphins network

This network, shown in Figure 3(c), represents social associations between 62 dolphins living in Doubtful Sound in New Zealand [32]. This network is inhomogeneous in that a few nodes possess a large number of edges and the others have only one or two edges. Since the size of the dolphins' network is modest, including the high-order transitivity, statistics is required to analyze ERGMs. Thus, we analyzed this network using an ERGM with the degree and shared partnership statistics. The likelihood function is given by

$$f(\boldsymbol{y}|\theta) = \frac{1}{\kappa(\theta)} \exp\left\{\theta_1 S_1(\boldsymbol{y}) + \theta_2 u(\boldsymbol{y}|\tau) + \theta_3 v(\boldsymbol{y}|\tau)\right\}, \tag{20}$$

where $S_1(\boldsymbol{y})$ is the edge count, $u(\boldsymbol{y}|\tau)$ is the GWD statistic, and $v(\boldsymbol{y}|\tau)$ is the GWESP statistic. For the GWD and GWESP statistics, τ is fixed to 0.8, same as [6].

The parameters are subject to the following priors:

$$\theta_1 \sim \text{Unif}(-6,-2), \quad \theta_2 \sim \text{Unif}(-8,8), \quad \theta_3 \sim \text{Unif}(-8,8). \tag{21}$$

As for the other two examples, this is set based on our pilot exploration of the model through a short run of the plain-DMH sampler with $\Pi = 1$ and a wide Gaussian prior.

AEX was applied to this network. To implement the ABC-based method for auxiliary parameter points selection, we set $\boldsymbol{\epsilon} = (60, 15, 120)$, where the three components correspond to the edge count, GWD, and GWESP, respectively. AEX was run 10 times independently. Each run consisted of 160,000 iterations with the same iteration distribution for the auxiliary and target chains for the Florentine business network. The results are summarized in Table 6 and can be interpreted as follows: if dolphins i and j have no common friends, then the log-odds for them becoming friends is -4.29; and if they have any positive number of common friends, and each is in at least one of the other triangle connections with their friends, then the log-odds for them becoming friends rises to -3.34. The parameter estimates indicate that in the dolphin network, the probability of adding edges is less than that of deleting edges but there is an overall tendency to form a k-star once an edge is formed in the network, and there is a tendency to completion of a triad once a k-star is formed in the network. For reference, Table 6 also gave the estimate from [6]. The two estimates are quite consistent for this example because the model does not suffer from the degeneracy problem.

This example also indicates that the AEX algorithm has a significant advantage over the CF-DMH sampler in CPU cost for large social networks. For the Florentine business

*Table 6. Parameter estimation for the dolphin network. The estimates were calculated by averaging over 10 independent runs with standard Monte Carlo errors reported in the parentheses. CPU(m): The CPU time (in minutes) cost by a single run on a personal computer with a quad-core i7 2.2GHz processor. * The results of CF-DMH are from Caimo and Friel (2011), where the standard errors were calculated based on 6 parallel chains, and the CPU time had been adjusted to our computer*

Method	Edge Counts	GWD($\tau = 0.8$)	GWESP($\tau = 0.8$)	CPU(m)
AEX	-4.29 (3.3e-2)	1.40 (7.0e-2)	0.95 (1.2e-3)	7.7
CF-DMH	-4.27 (1.0e-2)	1.30 (1.1e-2)	0.95 (2.1e-3)	80.8

Table 7. Rooted mean square errors the AEX and CF-DMH estimates for dolphin network example

Methods	Edge Count	GWD	GWESP
AEX	17.058	3.795	36.455
CF-DMH	17.232	3.828	36.215

network, which consists of 16 nodes, the CF-DMH sampler took 1,000 iterations to generate one auxiliary network and iterated for 30,000 iterations for simulating from the posterior. For the molecule synthetic network, which consists of 20 nodes, the CF-DMH sampler took 1,000 iterations to generate one auxiliary network and iterated for 32,000 iterations for simulating from the posterior. While, for the dolphin network, which consists of 62 nodes, the CF-DMH sampler took 15,000 iterations to generate one auxiliary network and iterated for 60,000 iterations for simulating from the posterior. In [6], the chain used for generating auxiliary networks is also called the auxiliary chain. As the network size increases, DMH needs to significantly increase the iteration number of the auxiliary chain, possibly, at an exponential rate. However, for the AEX algorithm, only one auxiliary chain is running, and its iteration number does not need to increase very fast with the network size. As seen from this example, the AEX algorithm still works well even with the same iteration setting as for the Florentine and molecule examples. This reflects that the AEX costs in each single run a little shorter CPU time for the Florentine and molecule examples, but much shorter CPU time for the dolphin example, than that of the CF-DMH sampler.

For both the AEX and CF-DMH estimates, we calculated RMSEs with $K = 20,000$. The results were summarized in Table 7, which indicates that the AEX estimate can be a little more accurate than the CF-DMH estimate. Figure 7 in Appendix C shows the goodness-of-fit (GOF) plots of the two estimates. Based on Tables 6 and 7, we can conclude that the AEX algorithm can perform equally well as or even better than the CF-DMH sampler for this example, while costing much shorter (less than 10%) CPU time.

6. DISCUSSION

In this paper, we have applied the AEX algorithm to Bayesian analysis for ERGMs, and established the convergence of the algorithm under mild conditions. Compared to the exchange algorithm, the AEX algorithm removes the requirement of perfect sampling, and thus is applicable to ERGMs. Compared to the DMH sampler, the AEX algorithm overcomes its theoretical flaw on convergence, while maintaining its computational efficiency. Due to the convergence issue, the DMH estimates can be biased, while this is not for the AEX estimates. Our numerical results indicate that the AEX algorithm can produce more accurate parameter estimates than the CF-DMH sampler for all three examples. In addition, the AEX algorithm has a significant advantage over the DMH sampler in CPU time for large social networks.

Our implementation for the AEX algorithm is plain in the sense that the auxiliary network is updated using the Gibbs sampler in the auxiliary chain and the parameter is updated using the MH algorithm in the target chain. Its efficiency can be improved by equipping with some advanced MCMC techniques in the auxiliary and/or target chains. For example, one can replace the Gibbs sampler used in the auxiliary chain by the tie-no-tie sampler, and apply adaptive direction sampling, parallel tempering, or evolutionary Monte Carlo [31] for updating the model parameters in the target chain.

In addition to parameter estimation, the AEX algorithm is ready to be applied to the problem of variable selection for ERGMs. This can be done by including multiple auxiliary chains in the simulation with each corresponding to a specific candidate model. In this way, a model can be selected from a pre-specified set of models under the Bayesian framework. Due to the parallel nature of the auxiliary and target chains, this algorithm can be conveniently implemented in a parallel machine.

APPENDIX A

(A_2) Let P_w denote the MH transition kernel for a given $w \in \mathcal{W}$ used in the auxiliary chain. For any $w \in \mathcal{W}$, P_w is ψ-irreducible and aperiodic [34]. In addition, there exist a function $V : \tilde{\mathcal{X}} \to [1, \infty)$ and a constant $\alpha \geq 2$ such that for any compact subset $\mathcal{K} \subset \mathcal{W}$,

(i) there exist a set $\mathcal{C} \subset \tilde{\mathcal{X}}$, an integer l, constants $0 < \lambda < 1$, b, ζ, $\delta > 0$ and a probability measure ν such that

- $\sup_{w\in\mathcal{K}} P_w^l V^\alpha(x) \le \lambda V^\alpha(x) + bI(x\in\mathcal{C}), \quad \forall x\in\tilde{\mathcal{X}}$,
- $\sup_{w\in\mathcal{K}} P_w V^\alpha(x) \le \zeta V^\alpha(x), \quad \forall x\in\tilde{\mathcal{X}}$,
- $\inf_{w\in\mathcal{K}} P_w^l(x,A) \ge \delta\nu(A), \quad \forall x\in\mathcal{C}, \quad \forall A\in\mathcal{B}_{\tilde{\mathcal{X}}}$.

where $P_wV(x) = \int_{\tilde{\mathcal{X}}} P_w(x,y)V(y)dy$ and $\mathcal{B}_{\tilde{\mathcal{X}}}$ is the Borel set defined on $\tilde{\mathcal{X}}$.

(ii) there exists a constant c such that for all $(w,w')\in\mathcal{K}\times\mathcal{K}$,

- $\|P_wg - P_{w'}g\|_V \le c\|g\|_V|w-w'|, \quad \forall g\in\mathcal{L}_V$,
- $\|P_wg - P_{w'}g\|_{V^\alpha} \le c\|g\|_{V^\alpha}|w-w'|, \quad \forall g\in\mathcal{L}_{V^\alpha}$,

where $|z|$ denotes the norm of the vector z, $\|g\|_V = \sup_{x\in\tilde{\mathcal{X}}}|g(x)|/V(x)$, and $\mathcal{L}_V = \{g:\tilde{\mathcal{X}}\to\mathbb{R}^m, \|g\|_V < \infty\}$.

APPENDIX B

Proof of Lemma 3.1. By the assumptions that $\mathcal{X}$ is compact and $f(\boldsymbol{x}|\theta)$ is bounded away from 0 and ∞, it follows from the convergence and strong law of large numbers (SLLN) of SAMC [2] that

(22)
$$\frac{1}{N}\sum_{t=1}^{N}\sum_{i=1}^{m}\left\{w_t^{(i)}\frac{\psi(\boldsymbol{z}_t|\theta')}{\psi(\boldsymbol{z}_t|\theta^{(i)})}I(J_t=i)\right\} \to \sum_{i=1}^{m}\int_{\mathcal{X}}\frac{\kappa(\theta^{(i)})}{p_i}\frac{\psi(\boldsymbol{z}|\theta')}{\psi(\boldsymbol{z}|\theta^{(i)})}p_if(\boldsymbol{z}|\theta^{(i)})d\boldsymbol{z} = m\kappa(\theta').$$

Note that as $t\to\infty$, the marginal distribution of $\boldsymbol{z}_t$ converges to the mixture distribution $g(\boldsymbol{z}) = \sum_{i=1}^m p_if(\boldsymbol{z}|\theta^{(i)})$. Similarly, for any Borel set $A\subset\mathcal{X}$,

(23)
$$\begin{aligned}&\frac{1}{N}\sum_{t=1}^{N}\sum_{i=1}^{m}\left\{w_t^{(i)}\frac{\psi(\boldsymbol{z}_t|\theta')}{\psi(\boldsymbol{z}_t|\theta^{(i)})}I(J_t=i\ \&\ \boldsymbol{z}_t\in A)\right\}\\ &\quad\to\sum_{i=1}^{m}\int_A\frac{\kappa(\theta^{(i)})}{p_i}\frac{\psi(\boldsymbol{z}|\theta')}{\psi(\boldsymbol{z}|\theta^{(i)})}p_if(\boldsymbol{z}|\theta^{(i)})d\boldsymbol{z}\\ &\quad= m\int_A\psi(\boldsymbol{z}|\theta')d\boldsymbol{z}.\end{aligned}$$

Putting (22) and (23) together, we have

$$\frac{\sum_{t=1}^N\sum_{i=1}^m\left\{w_t^{(i)}\frac{\psi(\boldsymbol{z}_t|\theta')}{\psi(\boldsymbol{z}_t|\theta^{(i)})}I(J_t=i)\right\}}{\sum_{t=1}^N\sum_{i=1}^m\left\{w_t^{(i)}\frac{\psi(\boldsymbol{z}_t|\theta')}{\psi(\boldsymbol{z}_t|\theta^{(i)})}I(J_t=i)\right\}} \to\int_A f(\boldsymbol{z}|\theta')d\boldsymbol{z}, \quad \text{as } N\to\infty,$$

which, by Lebesgue's dominated convergence theorem, implies that

(24)
$$\begin{aligned}&P(\boldsymbol{X}\in A)\\ &= E\left[P(\boldsymbol{X}\in A|\boldsymbol{z}_1,\theta^{(J_1)},w_1^{(J_1)};\ldots;\boldsymbol{z}_N,\theta^{(J_N)},w_N^{(J_N)})\right]\\ &\to\int_A f(\boldsymbol{z}|\theta')d\boldsymbol{z},\end{aligned}$$

where $\boldsymbol{X}$ denotes a sample resampled from $\{\boldsymbol{z}_1,\ldots,\boldsymbol{z}_N\}$ with a probability given in equation (11) of the original manuscript. This completes the proof of the lemma. □

Proof of Lemma 3.2. Since P defines an irreducible and aperiodic Markov chain, so does P_v. To show $\tilde{P}_l$ has the same property, it suffices to show that the accessible sets of P_v are included in those of $\tilde{P}_l$. More precisely, we show by induction that for any $k\in\mathbb{N}$, $\theta\in\Theta$ and $A\in\mathcal{B}(\Theta)$ such that $P_v^k(\theta,A)>0$, then $\tilde{P}_l^k(\theta,A)>0$. First, for any $\theta\in\Theta$ and $A\in\mathcal{B}(\Theta)$,

$$\begin{aligned}\tilde{P}_l(\theta,A) &\ge \int_A\int_{\mathcal{X}}(1\wedge\beta_l)(1\wedge r_v(\theta,\boldsymbol{x},\theta'))\nu_l(d\boldsymbol{x}|\theta')q(\theta,d\theta')\\ &\quad + I(\theta\in A)\rho(\theta)\\ &\ge (1\wedge\beta_*)\int_A\int_{\mathcal{X}}(1\wedge r_v(\theta,\boldsymbol{x},\theta'))\nu_l(d\boldsymbol{x}|\theta')q(\theta,d\theta')\\ &\quad + I(\theta\in A)\rho(\theta),\end{aligned}$$

where $I(\cdot)$ is the indicator function and $\beta_* = \min_{\boldsymbol{x},\theta,\theta'}\beta_l(\theta,\boldsymbol{x},\theta')$. By the conditions (i) and (ii), we have $\beta_*>0$ and thus the implication is true for $k=1$. Assume the induction assumption is true up to some $k=n\ge 1$. Now, for some $\theta\in\Theta$, let $A\in\mathcal{B}(\Theta)$ such that $P_v^{n+1}(\theta,A)>0$ and assume that

$$\int_\Theta\tilde{P}_l^n(\theta,d\theta')\tilde{P}_l(\theta',A) = 0,$$

which implies $\tilde{P}_l(\theta',A) = 0$, $\tilde{P}_l^n(\theta,\cdot)$-a.s. and hence $P_v(\theta',A)=0$, $\tilde{P}_l^n(\theta,\cdot)$-a.s. from the induction assumption for $k=1$. From this and the induction assumption for $k=n$, we deduce that $P_v(\theta',A)=0$, $P_v^n(\theta,\cdot)$-a.s. (by contradiction), which contradicts the fact $P_v^{n+1}(\theta,A)>0$. □

Proof of Theorem 3.1. For any $k\ge 1$ and any $\phi:\Theta\to[-1,1]$, we have

$$\tilde{P}_l^k\phi(\theta_0) - \pi(\phi|\boldsymbol{y}) = S_1(k) + S_2(k),$$

where $\pi(\phi|\boldsymbol{y}) = \pi(\phi(\theta)|\boldsymbol{y})$ for notational simplicity, and

$$\begin{aligned}S_1(k) &= P^k\phi(\theta_0) - \pi(\phi|\boldsymbol{y}),\\ S_2(k) &= \tilde{P}_l^k\phi(\theta_0) - P^k\phi(\theta_0),\end{aligned}$$

where P denotes the transition kernel defined in equation (14) of the original manuscript.

For the term $S_2(k)$, we can further decompose it as follows. For any k_0 $(1 \le k_0 < k)$,

(25)
$$\begin{aligned}
|S_2(k)| &\le |\tilde{P}_l^k \phi(\theta_0) - \tilde{P}_l^{k_0} \phi(\theta_0)| \\
&\quad + |\tilde{P}_l^{k_0} \phi(\theta_0) - P^{k_0} \phi(\theta_0)| + |P^{k_0} \phi(\theta_0) - P^k \phi(\theta_0)| \\
&= \left| \sum_{m=0}^{k_0-1} [P^m \tilde{P}_l^{k_0-m} \phi(\theta_0) - P^{m+1} \tilde{P}_l^{k_0-(m+1)} \phi(\theta_0)] \right| \\
&\quad + |\tilde{P}_l^k \phi(\theta_0) - \tilde{P}_l^{k_0} \phi(\theta_0)| + |P^k \phi(\theta_0) - P^{k_0} \phi(\theta_0)| \\
&= \left| \sum_{m=0}^{k_0-1} P^m (\tilde{P}_l - P) \tilde{P}_l^{k_0-(m+1)} \phi(\theta_0) \right| \\
&\quad + |\tilde{P}_l^k \phi(\theta_0) - \tilde{P}_l^{k_0} \phi(\theta_0)| + |P^k \phi(\theta_0) - P^{k_0} \phi(\theta_0)|.
\end{aligned}$$

For any $\epsilon > 0$, it follows from equation (13) of the original manuscript that there exists an $L(\epsilon, \theta_0)$ such that for any $l > L(\epsilon, \theta_0)$,

$$\begin{aligned}
|S_2(k)| &\le k_0 \epsilon + |\tilde{P}_l^k \phi(\theta_0) - \tilde{P}_l^{k_0} \phi(\theta_0)| \\
&\quad + |P^k \phi(\theta_0) - P^{k_0} \phi(\theta_0)| \\
&= k_0 \epsilon + S_3(l, k, k_0) + S_4(k, k_0).
\end{aligned}$$

The magnitudes of $S_1(k)$, $S_4(k, k_0)$ and $S_3(l, k, k_0)$ can be controlled following from the convergence of the transition kernel P and Lemma 3.2. For any $\epsilon > 0$, there exists $k_0 = K(\epsilon, \theta_0, l)$ such that for any $k > k_0$,

$$|S_1(k)| \le \epsilon, \quad S_3(l, k, k_0) \le \epsilon, \quad S_4(k, k_0) \le \epsilon.$$

Setting $\epsilon = \epsilon'/(k_0 + 3)$ and summarizing the results of $S_1(k)$ and $S_2(k)$, we conclude the following: For any $\epsilon' > 0$ and any $\theta_0 \in \Theta$, there exists $L(\epsilon', \theta_0) \in \mathbb{N}$ and $K(\epsilon', \theta_0, l) \in \mathbb{N}$ such that for any $l > L(\epsilon', \theta_0)$ and $k > K(\epsilon', \theta_0, l)$,

$$\|\tilde{P}_l^k(\theta_0, \cdot) - \pi(\cdot|\boldsymbol{y})\| \le \epsilon'. \qquad \square$$

APPENDIX C

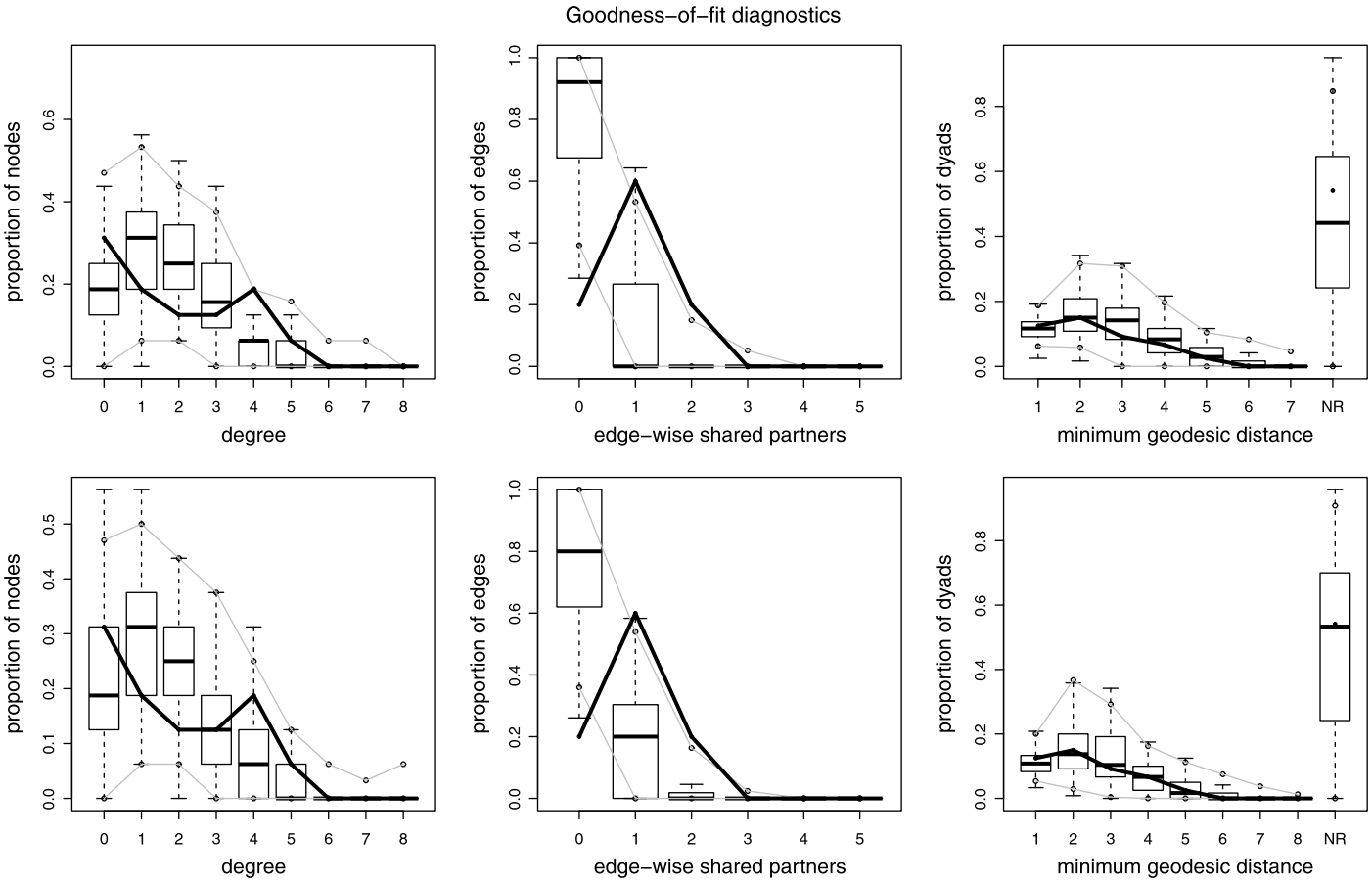

Figure 5. Goodness-of-fit (GOF) plots for the Florentine business network: Row 1: AEX; Row 2: plain-DMH with $\Pi = 200$*. The solid line shows the observed network statistics and the box-plots represent the distribution of simulated network statistics.*

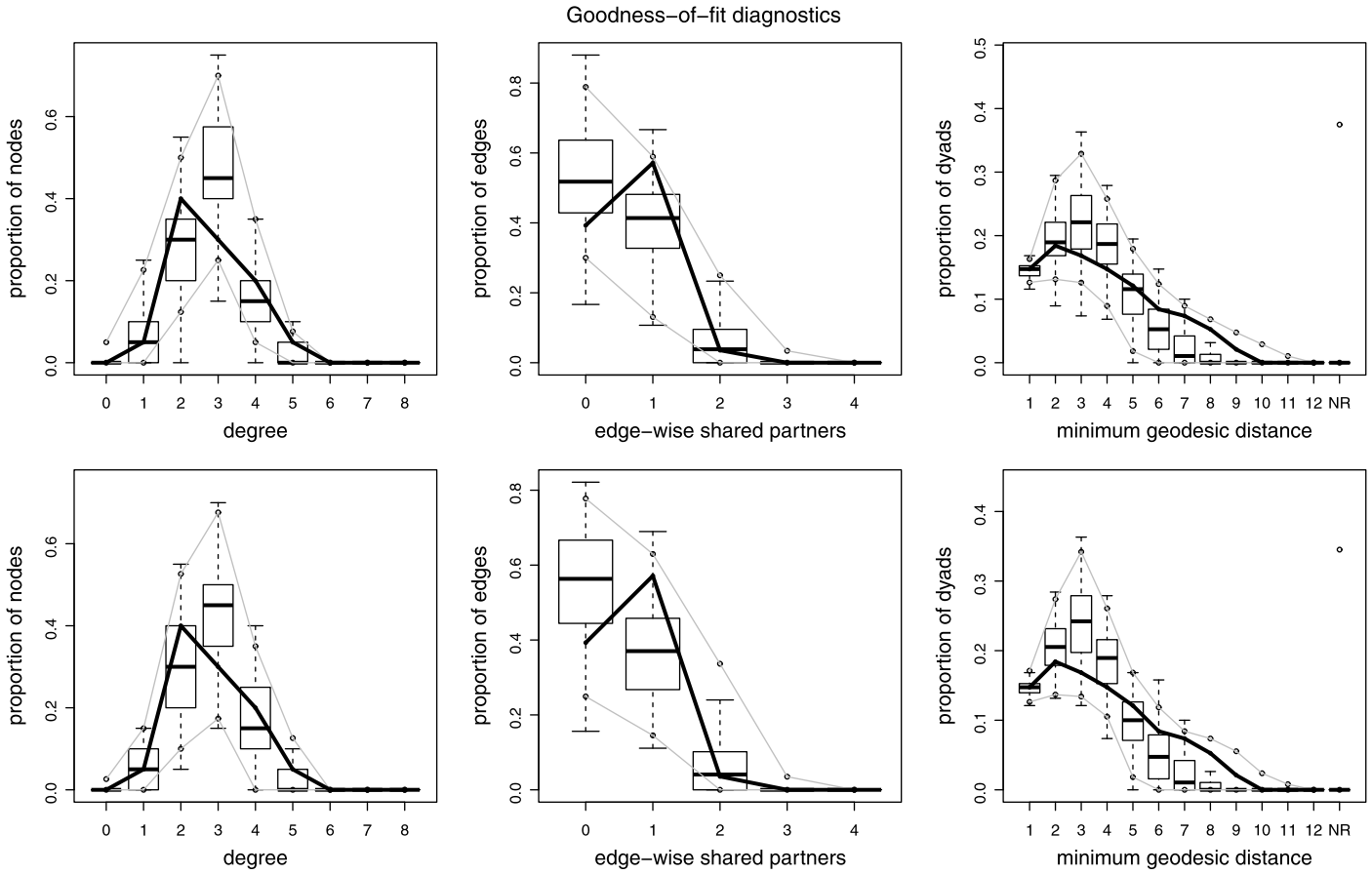

Figure 6. Goodness-of-fit (GOF) plots for the molecule synthetic network: Row 1: AEX; Row 2: CF-DMH. The solid line shows the observed network statistics and the box-plots represent the distribution of simulated network statistics.

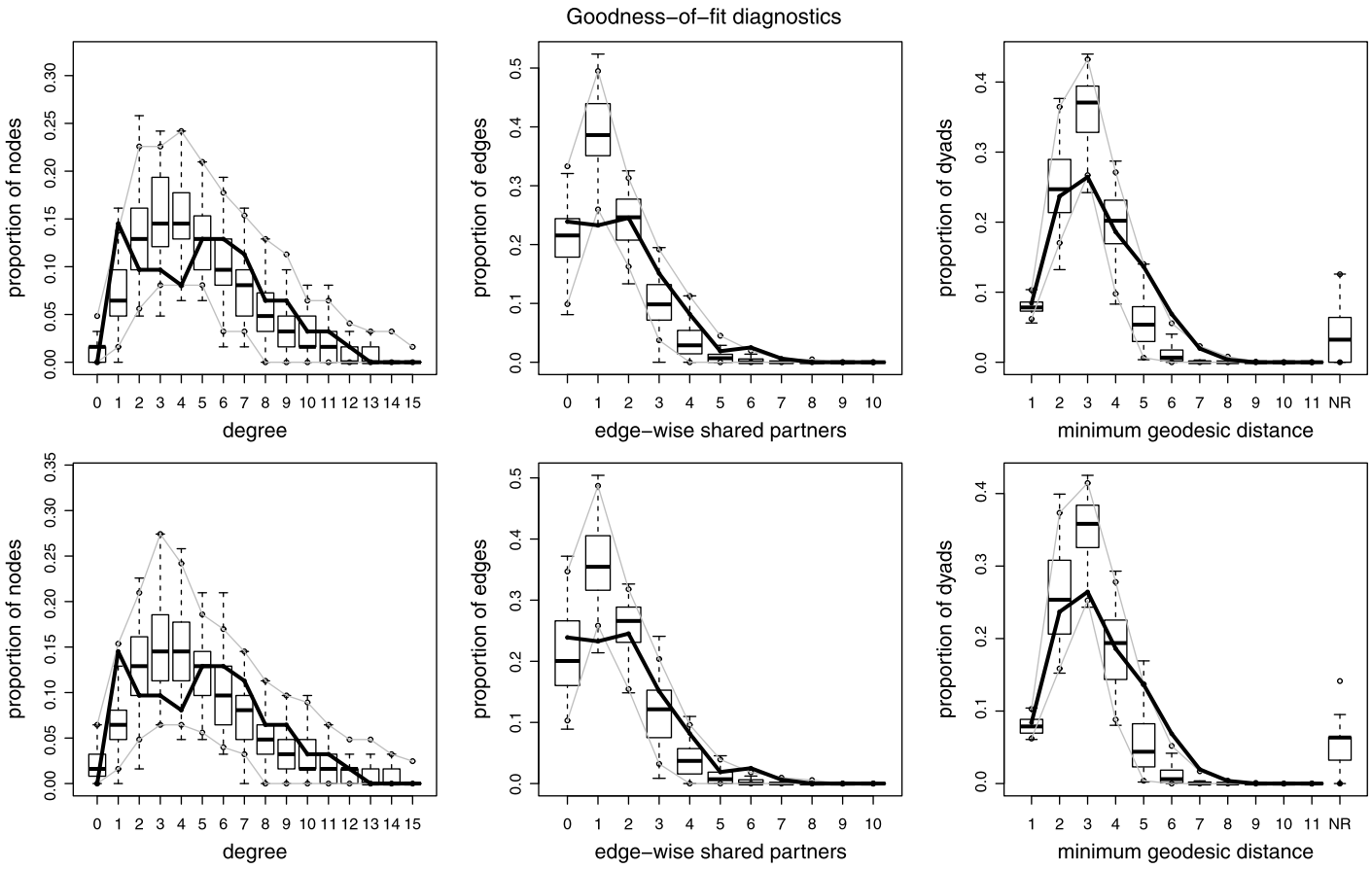

Figure 7. Goodness-of-fit plots for the dolphin network: Row 1: AEX; Row 2: CF-DMH. The solid line shows the observed network statistics and the box-plots represent the distribution of simulated network statistics.

Received 29 December 2012

REFERENCES

[1] Andrieu, C., Moulines, E. and Priouret, P. (2005). Stability of stochastic approximation under verifiable conditions. *SIAM Journal of Control and Optimization* **44** 283–312.

[2] Atchadé, Y. F. and Liu, J. S. (2010). The Wang-Landau algorithm in general state spaces: Applications and convergence analysis. *Statistica Sinica* **20** 209–233.

[3] Atchadé, Y. F., Lartillot, N. and Robert, C. (2012). Bayesian computation for statistical models with intractable normalizing constants. *Brazilian Journal of Probability and Statistics* **27** 416–436, page 13.

[4] Bartz, K., Blitzstein, J. and Liu, J. (2008). Monte Carlo maximum likelihood for exponential random graph models: From snowballs to umbrella densities. Technical Report, Harvard University.

[5] Beaumont, M. A., Zhang, W. and Balding, D. J. (2002). Approximate Bayesian computation in population genetics. *Genetics* **162** 2025–2035.

[6] Caimo, A. and Friel, N. (2011). Bayesian inference for exponential random graph models. *Social Networks* **33** 41–55.

[7] Cranmer, S. J. and Desmarais, B. A. (2011). Inferential network analysis with exponential random graph models. *Political Analysis* **19** 66–86.

[8] Efron, B. and Tibshirani, R. J. (1993). *An Introduction to the Bootstrap*. Chapman and Hall.

[9] Erdös, P. and Rényi, A. (1959). On random graph. *Publicationes Mathematicae* **6** 290–297.

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

[11] Geyer, C. J. (1991). Markov chain Monte Carlo maximum likelihood. In: *Computer Science and Statistics: Proceedings of the 23rd Symposium on the Interface*, E. M. Keramigas, ed., Interface Foundation, Fairfax, pp. 153–163.

[12] Gilks, W. R., Roberts, G. O. and George, E. I. (1994). Adaptive direction sampling. *The Statistician* **43** 179–189.

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

[14] Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika* **82** 711–732.

[15] Gumpertz, M. L., Graham, J. M. and Ristaino, J. B. (1997). Autologistic model of spatial pattern of Phytophthora epidemic in bell pepper: Effects of soil variables on disease presence. *Journal of Agricultural, Biological, and Environmental Statistics* **2** 131–156.

[16] Haario, H., Saksmaan, E. and Tamminen, J. (2001). An adaptive Metropolis algorithm. *Bernoulli* **7** 223–242.

[17] Handcock, M. S. (2003). Statistical models for social networks: Degeneracy and inference. In: *Dynamic Social Network Modeling and Analysis*, R. L. Breiger, K. M. Carley and P. E. Pattison, eds, National Academies Press, Washington, DC, pp. 229–240.

[18] Holland, P. W. and Leinhardt, S. (1981). An exponential family of probabilitic distributions for directed networks. *Journal of the American Statistical Association* **76** 33–65.

[19] Hunter, D. R. (2007). Curved exponential family models for social network. *Social Networks* **29** 216–230.

[20] Hunter, D. R., Goodreau, S. M. and Handcock, M. S. (2008). Goodness of fit of social network models. *Journal of the American Statistical Association* **103** 248–258.

[21] Hunter, D. R. and Handcock, M. S. (2006). Inference in curved exponential family models for network. *Journal of Computational and Graphical Statistics* **15** 565–583.

[22] Hunter, D. R., Handcock, M. S., Butts, C. T., Goodreau, S. M. and Morris, M. (2008). `ergm`: A package to fit, simulate and diagnose exponential-family models for networks. *Journal of Statistical Software* **24** 3.

[23] Jin, I. H. and Liang, F. (2012). Fitting social network models using varying truncation stochastic approximation MCMC algorithm. *Journal of Computational and Graphical Statistics*, in press.

[24] Kaiser, M. S., Caragea, P. C. and Furukawa, K. (2012). Centered parameterizations and dependence limitations in Markov random field models. *Journal of Statistical Planning and Inference* **142** 1855–1863.

[25] Koskinen, J. H. (2008). The linked importance sampler auxiliary variable Metropolis-Hastings algorithm for distributions with intractable normalizing constants. Technical Report, University of Melbourne.

[26] Liang, F. (2010). A double Metropolis-Hastings sampler for spatial models with intractable normalizing constants. *Journal of Statistical Computing and Simulation* **80** 1007–1022.

[27] Liang, F. and Jin, I. H. (2013). A Monte Carlo metropolis-hastings algorithm for sampling from distributions with intractable normalizing constants. *Neural Computation* **25** 2199–2234.

[28] Liang, F., Jin, I. H., Yuan, Y. and Song, Q. (2012). An adaptive exchange algorithm for sampling from distribution with intractable normalizing constants. Technical Report, Texas A&M University.

[29] Liang, F., Liu, C. and Carroll, R. J. (2007). Stochastic approximation in Monte Carlo computation. *Journal of the American Statistical Association* **102** 305–320.

[30] Liang, F., Liu, C. and Carroll, R. J. (2010). *Advanced Markov Chain Monte Carlo Methods: Learning from Past Samples*. John Wiley & Sons, Ltd.

[31] Liang, F. and Wong, W. H. (2001). Real parameter evolutionary Monte Carlo with applications in Bayesian mixture models. *Journal of the American Statistical Association* **96** 653–666.

[32] Lusseau, D., Schneider, K., Boisseau, O. J., Haase, P., Slooten, E. and Dawson, S. M. (2003). The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations. *Behavioral Ecology and Sociobiology* **54** 396–405.

[33] Marjoram, P., Molitor, J., Plagnol, V. and Tavare, S. (2003). Markov Chain Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences of the United States of America* **100** 15324–15328.

[34] Meyn, S. P. and Tweedie, R. L. (1995). *Markov Chains and Stochastic Stability*. Springer, New York.

[35] Møller, J., Pettitt, A. N., Reeves, R. W. and Berthelsen, K. K. (2006). An efficient Markov chain Monte Carlo method for distributions with intractable normalising constants. *Biometrika* **93** 451–459.

[36] Murray, I., Ghahramani, Z. and MacKay, D. J. C. (2006). MCMC for doubly-intractable distributions. In: *Proc. of 22nd Annual Conference on Uncertainty in Artificial Intelligence (UAI)*.

[37] Padgett, J. F. (1994). Marriage and elite structure in Renaissance Florence, 1282–1500. Social Science History Association.

[38] Propp, J. G. and Wilson, D. B. (1996). Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures and Algorithms* **9** 223–252.

[39] Roberts, G. O. and Gilks, W. R. (1994). Convergence of adaptive direction sampling. *Journal of Multivariate Analysis* **49** 287–298.

[40] Robins, G. E., Pattison, P. E., Kalish, Y. and Lusher, D. (2007). An introduction to exponential random graph (p^*) models for social networks. *Social Networks* **29** 173–191.

[41] Robins, G. E., Snijers, T. A. B., Wang, P., Handcock, M. S. and Pattison, P. E. (2007). Recent development in exponential random graph models for social networks. *Social Networks* **29** 192–215.

[42] Saul, Z. and Filkov, V. (2007). Exploring biological network structure using exponential random graph models. *Bioinformatics* **23** 2604–2611.
[43] Schweinberger, M. (2011). Instability, sensitivity, and degeneracy of discrete exponential families. *Journal of the American Statistical Association* **106** 1361–1370.
[44] Snijders, T. A. B. (2002). Markov chain Monte Carlo estimation of exponential random graph models. *Journal of Social Structure* **3**.
[45] Snijders, T. A. B., Pattison, P. E., Robins, G. L. and Handcock, M. S. (2006). New specification for exponential random graph models. *Sociological Methodology* **36** 99–153.
[46] Strauss, D. and Ikeda, M. (1990). Pseudo-likelihood estimation for social network. *Journal of the American Statistical Association* **82** 204–212.
[47] Tierney, L. (1994). Markov chains for exploring posterior distributions (with discussion). *Annals of Statistics* **22** 1701–1762.
[48] van Duijn, M. A. J., Snijders, T. A. B. and Zijlstra, B. H. (2004). p_2: A random effects model with covariates for directed graphs. *Statistica Neerlandica* **58** 234–254.
[49] van Duijn, M. A. J., Gile, K. J. and Handcock, M. S. (2009). A framework for the comparison of maximum pseudo likelihood and maximum likelihood estimation of exponential family random graph models. *Social Networks* **31** 52–62.

Ick Hoon Jin
Postdoctoral Fellow, Department of Biostatistics
The University of Texas MD Anderson Cancer Center
Houston, TX 77030-4009
USA
E-mail address: ijin@mdanderson.org

Ying Yuan
Associate Professor, Department of Biostatistics
The University of Texas MD Anderson Cancer Center
Houston, TX 77030-4009
USA
E-mail address: yyuan@mdanderson.org

Faming Liang
Professor, Department of Statistics
Texas A&M University
College Station, TX, 77843-3143
USA
E-mail address: fliang@stat.tamu.edu