SCALABLE MCMC IN DEGREE CORRECTED STOCHASTIC BLOCK MODEL

Soumyasundar Pal and Mark Coates

Dept. of Electrical and Computer Engineering, McGill University, Montréal, Québec, Canada

ABSTRACT

Community detection from graphs has many applications in machine learning, biological and social sciences. While there is a broad spectrum of literature based on various approaches, recently there has been a significant focus on inference algorithms for statistical models of community structure. These algorithms strive to solve an inference problem based on a generative model of the network. Recent advances in stochastic gradient MCMC have played a crucial role in improving the scalability of these techniques. In this paper, we propose a version of a degree corrected stochastic block model and present an MCMC based inference algorithm. Experimental results on several real world networks demonstrate the effectiveness of the proposed approach.

Index Terms— Markov Chain Monte Carlo, Overlapping community detection, Degree corrected block models.

I. INTRODUCTION

In many applications, networks are used to model relationships among objects. Detection of community memberships of the nodes of graph structured data has many applications, including analyzing collaboration networks [1], protein interaction networks [2], and social networks [3]. Communities can be loosely defined as sets of nodes which have dense internal connectivity and few external connections [4]. Traditional approaches use heuristic algorithms like graph partitioning [5], spectral clustering [6] and hierarchical clustering [7]. Other techniques are based on edge betweenness [8], modularity optimization [9], link clustering [10], or clique percolation [11]. As these algorithms often have a heuristic objective function or perform greedy optimization of a global criterion over all possible partitions, some are scalable to large networks. Comprehensive reviews of these techniques are provided in [12], [13] and different quality metrics are throughly surveyed in [14].

An alternate avenue of research consists of hypothesizing a well-defined statistical model for the graphs and using inference techniques to estimate the model parameters [15]. The stochastic blockmodel (SBM) is one such model that is used widely in the literature [16]–[18]. This model assumes that each node can only participate in one of the communities and different node pairs having the same community memberships are stochastically equivalent. While this model is simple to understand, in real world networks, often a node can be a part of more than one community with varying membership strengths. To address this possibility, the mixed membership stochastic blockmodel (MMSB) is proposed in [19]. Exact inference in this model becomes intractable in large networks and several algorithms for approximate inference have been developed [20], [21].

A major weakness of the SBM is that it cannot model a heavy tailed degree distribution [22] well. Such degree distributions are observed in many real world networks. Several modifications to tackle degree heterogeneity within a community have been proposed and studied [16], [22]–[25]. A more detailed review of these papers is conducted in Section II. However, none of these techniques considers mixed memberships for the nodes. In this paper, we formulate the problem of overlapping community detection as a Bayesian inference problem on a mixed membership DCB (MMDCB) and develop a scalable MCMC algorithm for the inference of the posterior distribution. The intra-community edge probabilities use a simpler parametrization compared to [25], which allows our algorithm to scale linearly with the number of communities. Experimental results show that the proposed approach attains lower perplexity and better link prediction for several real world networks.

The paper is organized as follows. Section II reviews the related work. Section III introduces the model and states the inference task we address. Section IV provides a brief review of the SGLD algorithm. Section V describes the MCMC based Bayesian inference algorithm and Section VI presents and discusses the results of numerical experiments. The conclusion is provided in Section VII.

II. RELATED WORK

In [20], a scalable stochastic variational inference algorithm for an assortative MMSB (a-MMSB) is employed which outperforms many scalable techniques like [1], [7], [10]. It is shown in [21] that Stochastic Gradient Langevin Dynamics (SGLD) [26] based MCMC is significantly faster and performs much better compared to variational methods, as the latter class of algorithms has a relatively high bias due to variational approximations. As the SGLD uses stochastic gradient via mini-batch sampling for the MCMC update of the parameters and eliminates the acceptance ratio calculation step at the expense of introducing larger variance in small samples, the computation for each MCMC update does not grow severely with the number of nodes in the

graph. This allows this approach to obtain state-of-the-art performance for large networks.

In [16], the number of edges between any two nodes is assumed to be Poisson distributed, with a mean that depends both on community specific and node specific parameters. A heuristic algorithm is developed for maximum likelihood estimation. Similar models under the Bernoulli distribution assumption are studied in [22], [23]. In [22], a general theory for checking consistency of community detection in a degree corrected blockmodel (DCB) is developed, whereas [23] derives asymptotic minimax risks under a misclassification proportion loss and proposes a polynomial time algorithm for performing consistent community detection in an adaptive manner. In [24], a fast pseudo-likelihood method is introduced for learning parameters of DCBs and spectral clustering with perturbations is proposed as an initialization strategy. If there are only two communities, the resulting algorithm is guaranteed to recover the communities consistently under some mild assumptions on the initialization of memberships. A different version of the DCB is introduced in [25]; a logistic regression formulation with node correction terms is used. The authors employ a Gibbs sampler based on data augmentation strategy for inference of the parameters and develop a mapped consensus estimator to address label permutation of the nodes.

III. PROBLEM STATEMENT

We propose a MMDCB to model the community structure in an undirected graph of N nodes and K communities. The observed graph is denoted by $\mathbf{Y} = \{y_{ab} \in \{0,1\} : 1 \leq a < b \leq N\}$. The parameter $y_{ab} = 0$ or 1 indicates the absence or presence of a link between node a and node b. Similar to [19]–[21], each node a has a K dimensional community membership probability distribution $\pi_a = [\pi_{a1}, ...\pi_{aK}]^T$. The intra and inter community edge probabilities are parametrized via logit models with $q = \{q_k, 1 \leq k \leq K\}$ and $r = \{r_a, 1 \leq a \leq N\}$. The generative model is then: For any two nodes a and b,

i) Sample $z_{ab} \sim \pi_a$ and $z_{ba} \sim \pi_b$.
ii) If $z_{ab} = z_{ba} = k$, sample the link $y_{ab} \sim$ *Bernoulli*(β_k^{ab}), where $\beta_k^{ab} = logit^{-1}(q_k + r_a + r_b)$, otherwise sample the link $y_{ab} \sim$ *Bernoulli*(δ^{ab}), where $\delta^{ab} = logit^{-1}(r_a + r_b)$.

Since an edge between two nodes is more likely when both of them belong to the same community, we restrict $q_k \geq 0$. Note that the key difference between the MMDCB and the a-MMSB is that β_k^{ab} and δ^{ab} are different for different (a, b) pairs. The joint posterior of the parameters π, q and r is obtained via marginalization over $\{z_{ab} : 1 \leq a, b \leq N\}$:

$$p(\pi, q, r|\mathbf{Y}) \propto p(\pi)p(q)p(r)p(\mathbf{Y}|\pi, q, r)\,,$$

$$= \prod_{a=1}^{N} p(\pi_a)p(r_a) \prod_{k=1}^{K} p(q_k) \prod_{1 \leq a < b \leq N} \sum_{z_{ab}, z_{ba}} p(y_{ab}, z_{ab}, z_{ba}|\pi_a, \pi_b, q_{1:K}, r_a, r_b)\,. \quad (1)$$

We assume a symmetric Dirichlet prior, $Dir(\alpha)$ for π_a, whereas $q_k \sim \mathbf{1}(q_k \geq 0)\mathcal{N}(0, \sigma^2)$ and $r_a \sim \mathcal{N}(0, \sigma^2)$ have truncated normal and normal prior respectively. Here $\mathbf{1}(\cdot)$ denotes an indicator function that takes the value 1 when the condition holds and 0 otherwise. α and σ are hyper-parameters. Inside this Bayesian framework, the goal is to generate samples of (π, q, r) from the joint posterior, which can be used to form Monte Carlo estimates and assess uncertainty.

IV. STOCHASTIC GRADIENT LANGEVIN DYNAMICS

In an estimation problem of parameter θ with prior $p(\theta)$ and observed data $\mathbf{X} = \{x_1, x_2, ..., x_N\}$ with a generative model $p(\mathbf{X}|\theta) = \prod_{i=1}^{N} p(x_i|\theta)$, Bayesian inference involves computation of the posterior distribution, given as:

$$p(\theta|\mathbf{X}) \propto p(\theta) \prod_{i=1}^{N} p(x_i|\theta)\,. \quad (2)$$

When the posterior is not analytically tractable, an approximation can be obtained by sampling from the posterior using MCMC. One such technique uses Langevin Dynamics (LD) [27] to propose samples in a Metropolis-Hastings (MH) algorithm, which leads to a proposal distribution $q(\theta^*|\theta)$ as follows:

$$\theta^* = \theta + \frac{\epsilon}{2}\Big(\nabla_\theta \log p(\theta) + \sum_{i=1}^{N} \nabla_\theta \log p(x_i|\theta)\Big) + \zeta\,. \quad (3)$$

Here ϵ is the step size and $\zeta \sim \mathcal{N}(0, \epsilon I)$ is the random noise which prevents the algorithm from collapsing to one of the modes of the posterior. In order to correct for the discretization error in solving the Langevin equation, θ^* is accepted as the new sample with an MH acceptance probability of $\min\Big(1, \frac{p(\theta^*|\mathbf{x})q(\theta|\theta^*)}{p(\theta|\mathbf{x})q(\theta^*|\theta)}\Big)$. If different dimensions of θ have very different scales or high correlation, the isotropic proposal in (3) can potentially lead to slow mixing, particularly in high dimensional problems. A locally adaptive preconditioning using a Riemannian metric tensor [28] leads to Riemannian Langevin Dynamics (RLD) algorithm, which explores the posterior distribution more efficiently.

A major weakness of any standard MCMC algorithm is that computing the proposal distribution and the acceptance ratio has complexity $\mathcal{O}(N)$, since we need to process all of the observations at each iteration. In recent years, a significant amount of research has aimed to develop stochastic gradient versions [26], [29]–[31] of these algorithms. In [26], the exact computation of the gradient of the likelihood in (3) was replaced by a stochastic approximation and the

acceptance ratio calculation was ignored. More specifically, at iteration t, the gradient term was approximated by an unbiased estimator with $\mathcal{O}(n)$ complexity ($n \ll N$) considering a random mini-batch $\mathbf{X_t} = \{x_{t1}, x_{t2}, ..., x_{tn}\}$ as follows:

$$\nabla_\theta \log p(\mathbf{X}|\theta) \approx \frac{N}{n} \sum_{x_{ti} \in \mathbf{X_t}} \nabla_\theta \log p(x_{ti}|\theta)\,. \tag{4}$$

The resulting SGLD algorithm still guarantees convergence to the true posterior, if an annealed step-size schedule ϵ_t is employed which satisfies $\sum_{t=1}^{\infty} \epsilon_t = \infty$ and $\sum_{t=1}^{\infty} \epsilon_t^2 < \infty$.

V. METHODOLOGY

In this section, we derive the novel MCMC based algorithm for sampling the model parameters. Direct use of the SGLD update for π does not guarantee that all samples of π_a's lie on the probability simplex. Hence we consider an expanded mean parametrization [29] as follows. We introduce a new parameter $\phi_a \in \mathbb{R}_+^K$ and adopt a product of independent $Gamma(\alpha, 1)$ distributions as its prior. We define $\pi_{ak} = \dfrac{\phi_{ak}}{\sum_{l=1}^{K} \phi_{al}}$, which results in a symmetric Dirichlet prior, $Dir(\alpha)$ for π_a. The boundary conditions $\phi_{ak} \geq 0$ can be satisfied by simply taking the absolute value of the update at each iteration. If X is a Bernoulli random variable with parameter ρ, we denote $p(X = x|\rho) = \rho^x(1-\rho)^{(1-x)}$ as $\mathcal{B}(x;\rho)$ for $x \in \{0, 1\}$, to simplify notation. We first define the following probabilities:

$$\begin{aligned} f_{ab}^{(y)}(k,l) &= p(y_{ab}, z_{ab} = k, z_{ab} = l|\pi_a, \pi_b, r_a, r_b, q_k, q_l)\,, \\ &= \begin{cases} \pi_{ak}\pi_{bk}\mathcal{B}(y_{ab};\beta_k^{ab})\,, & \text{if } k = l \\ \pi_{ak}\pi_{bl}\mathcal{B}(y_{ab};\delta^{ab})\,, & \text{if } k \neq l, \end{cases} \end{aligned} \tag{5}$$

$$\begin{aligned} Z_{ab}^{(y)} &= p(y_{ab}|\pi_a, \pi_b, r_a, r_b, q) = \sum_{k=1}^{K}\sum_{l=1}^{K} f_{ab}^{(y)}(k,l)\,, \\ &= \mathcal{B}(y_{ab};\delta^{ab}) + \sum_{k=1}^{K} \Big(\mathcal{B}(y_{ab};\beta_k^{ab}) - \mathcal{B}(y_{ab};\delta^{ab})\Big)\pi_{ak}\pi_{bk}\,, \end{aligned} \tag{6}$$

and

$$\begin{aligned} f_{ab}^{(y)}(k) &= p(y_{ab}, z_{ab} = k|\pi_a, \pi_b, r_a, r_b, q) = \sum_{l=1}^{K} f_{ab}^{(y)}(k,l)\,, \\ &= \pi_{ak}\Big(\mathcal{B}(y_{ab};\beta_k^{ab})\pi_{bk} + \mathcal{B}(y_{ab};\delta^{ab})(1-\pi_{bk})\Big)\,. \end{aligned} \tag{7}$$

Based on these probabilities, we compute the partial derivatives of the log likelihood of y_{ab} as follows:

$$g_{ab}(q_k) = \nabla_{q_k} \log p(y_{ab}|\pi, q, r) = \frac{f_{ab}^{(y)}(k,k)}{Z_{ab}^{(y)}}(y_{ab} - \beta_k^{ab})\,.$$

$$\begin{aligned} g_{ab}(r_a) &= \nabla_{r_a} \log p(y_{ab}|\pi, q, r)\,, \\ &= (y_{ab} - \delta^{ab}) + \frac{\sum_{k=1}^{K} f_{ab}^{(y)}(k,k)(\delta^{ab} - \beta_k^{ab})}{Z_{ab}^{(y)}}\,, \end{aligned}$$

and

$$\begin{aligned} g_{ab}(\phi_{ak}) &= \nabla_{\phi_{ak}} \log p(y_{ab}|\pi, q, r)\,, \\ &= \frac{f_{ab}^{(y)}(k)}{Z_{ab}^{(y)}\phi_{ak}} - \frac{1}{\sum_{l=1}^{K}\phi_{al}}\,. \end{aligned} \tag{8}$$

Based on these partial derivatives, the LD updates of q and r are:

$$q_k^{(t+1)} = \Big|q_k^{(t)} + \frac{\epsilon_t}{2}\Big(-\frac{q_k^{(t)}}{\sigma^2} + \sum_{a=1}^{N}\sum_{b=a+1}^{N} g_{ab}(q_k^{(t)})\Big) + \zeta_k^{(t)}\Big|\,, \tag{9}$$

$$r_a^{(t+1)} = r_a^{(t)} + \frac{\epsilon_t}{2}\Big(-\frac{r_a^{(t)}}{\sigma^2} + \sum_{b=1, b\neq a}^{N} g_{ab}(r_a^{(t)})\Big) + \zeta'^{(t)}_a\,, \tag{10}$$

where $\zeta_k^{(t)}, \zeta'^{(t)}_a \sim \mathcal{N}(0, \epsilon_t)$. For ϕ_a, we follow the RLD rule with the Riemannian metric tensor $G(\phi_a) = diag(\phi_a)^{-1}$ of [21], [29]:

$$\phi_{ak}^{(t+1)} = \Big|\phi_{ak}^{(t)} + \frac{\epsilon_t}{2}\Big(\alpha - \phi_{ak}^{(t)} + \phi_{ak}^{(t)}\sum_{b=1, b\neq a}^{N} g_{ab}(\phi_{ak}^{(t)})\Big) + \sqrt{\phi_{ak}^{(t)}}\zeta''^{(t)}_{ak}\Big|\,, \tag{11}$$

where $\zeta''^{(t)}_{ak} \sim \mathcal{N}(0, \epsilon_t)$. Naive implementation of (9), (10) and (11) is $\mathcal{O}(N^2K)$ per iteration, which is infeasible for large graphs. We instead employ stochastic approximation as follows. For the update of the q_k values in eq. (9), a random mini-batch contains too few links, because in most of the real world datasets, the number of edges is closer to $\mathcal{O}(N)$ than $\mathcal{O}(N^2)$. Hence, we split the sum over all node pairs into two separate terms involving all links and non-links and use a separate mini-batch to approximate each of them. At any single iteration, we update r_a and the ϕ_{ak} values at only n_1 randomly sampled nodes ($n_1 \ll N$), keeping the rest of them fixed. At any of the randomly selected n_1 nodes, we split the sum in eq. (10) and (11) into two separate terms corresponding to the neighbours and non-neighbours of node a and separate mini-batches are used such that each update is computed based on n_2 nodes ($n_2 \ll n_1$) in total. Overall, the update of r and ϕ involves $\mathcal{O}(n_1 n_2 K)$ operations instead of $\mathcal{O}(N^2K)$ complexity for a full batch update.

VI. EXPERIMENTAL RESULTS

We evaluate our algorithm and compare it to the SG-MCMC algorithm on the a-MMSB [21] using four real world academic collaboration datasets: NETSCIENCE (1589 nodes, 2742 edges) [32], RELATIVITY (5242 nodes, 14496 edges), HEP-TH (9877 nodes, 25998 edges) and HEP-PH (12008 nodes, 118521 edges) [33]. We consider each dataset as an undirected graph with a symmetric adjacency matrix. For hyper-parameters of the prior distributions, we use $\alpha = \frac{1}{K}, \sigma = 10$ and a decreasing step size of $\epsilon_t = \epsilon_0(t+\tau)^{-\kappa}$ with $\epsilon_0 = 10, \tau = 1000$ and $\kappa = 0.75$ is employed. We set $n_1 = 500, n_2 = 50$ so that for the datasets considered here, $n_2 \ll n_1 \ll N$ is satisfied. Since the true number

of communities is unknown, we conduct experiments with four different values of K. We construct a held-out test set $\mathbf{Y_{test}}$ by randomly sampling 10% of the links and same number of non-links. The predictive performance of the algorithm is measured by computing the average perplexity. This is defined as the exponential of the negative average predictive probabilities on the test set. Based on T samples of parameters $\{\pi^{(i)}, q^{(i)}, r^{(i)}\}_{i=1}^{T}$, we estimate the expectation over the posterior by a Monte Carlo average to obtain perplexity as follows:

$$perp_{avg}(\mathbf{Y_{test}}|\{\pi^{(i)}, q^{(i)}, r^{(i)}\}_{i=1}^{T}) = \exp\Big(-\frac{\sum_{y_{ab}\in\mathbf{Y_{test}}} \log\{\frac{1}{T}\sum_{i=1}^{T} p(y_{ab}|\pi^{(i)}, q^{(i)}, r^{(i)})\}}{|\mathbf{Y_{test}}|}\Big). \quad (12)$$

We compute perplexity at regular intervals based on the most recent $T = 500$ samples to determine convergence of the MCMC algorithms. If the perplexity does not drop by more than 0.1% in 500 iterations, we declare convergence and stop running the algorithms. From Figure 1, we observe that SG-MCMC achieves a lower perplexity for the MMDCB compared to the a-MMSB in the HEP-PH dataset with $K = 50$ communities, which indicates that the proposed MMDCB generalizes better to unseen links and non-links for this dataset. We obtain similar results for all the datasets considered here. Apart from measuring perplexity, at convergence, we conduct a link prediction experiment on the test set using the MC estimates of the predictive probabilities and report the area under the receiver operating characteristic (ROC) curve (AUC) as the performance metric. For the four datasets considered here, the perplexity and AUC values are shown for different numbers of communities (K) in Figures 2 and 3 respectively. We observe that for almost all of the cases, the proposed MCMC algorithm based on the MMDCB obtains lower perplexity and higher AUC compared to the a-MMSB.

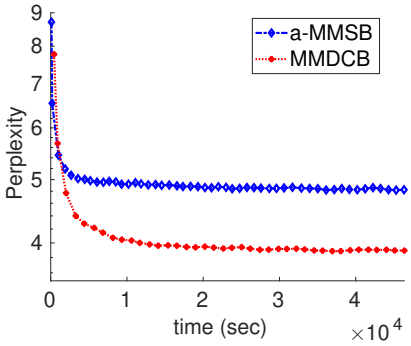

Fig. 1. Convergence of perplexity for the HEP-PH dataset with $K = 50$.

VII. CONCLUSION

In this paper, we introduce a mixed membership DCB as a generative model for networks and propose a scalable MCMC algorithm for the inference task. Experimental results on several real world datasets show improvement compared to the a-MMSB of [21]. This technique can be employed to model large scale graph data in a Bayesian setting. Future research endeavours will investigate ways to propose better generative models, to reduce computational complexity by employing more efficient minibatch sampling strategies and to incorporate better MCMC techniques like stochastic gradient Hamiltonian Monte Carlo [30], [31].

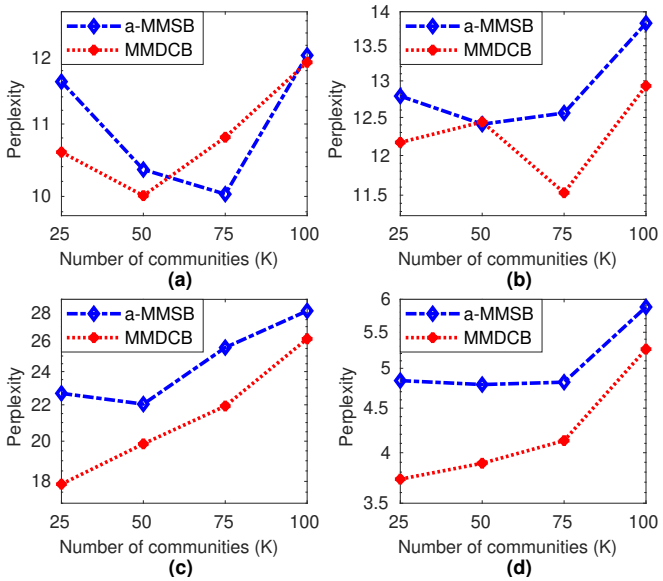

Fig. 2. Perplexity vs the number of communities (K) for (a) NETSCIENCE, (b) RELATIVITY, (c) HEP-TH and (d) HEP-PH datasets.

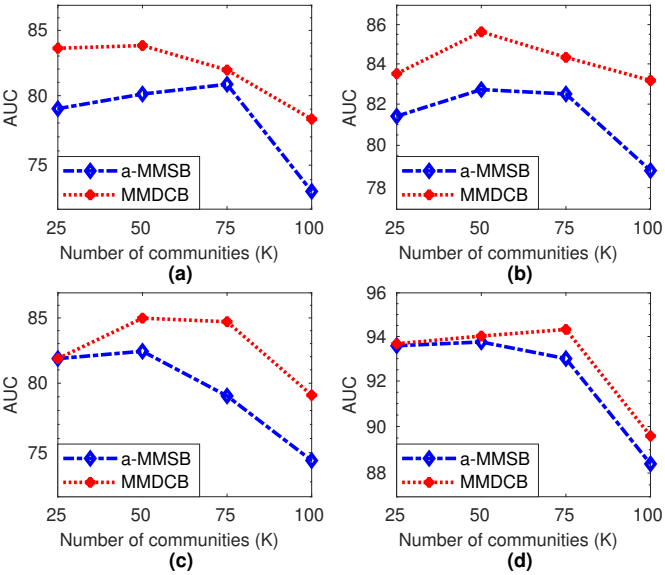

Fig. 3. AUC vs the number of communities (K) for (a) NETSCIENCE, (b) RELATIVITY, (c) HEP-TH and (d) HEP-PH datasets.

VIII. REFERENCES

[1] B. Ball, B. Karrer, and M. E. J. Newman, "Efficient and principled method for detecting communities in networks," *Phys. Rev. E*, vol. 84, no. 3, pp. 036103, Sep. 2011.

[2] H. Mahmoud, F. Masulli, S. Rovetta, and G. Russo, "Community detection in protein-protein interaction networks using spectral and graph approaches," in *Proc. Int. Meeting Comput. Intell. Methods for Bioinformatics and Biostatistics*, Jul. 2014, pp. 62–75.

[3] K. He, Y. Li, S. Soundarajan, and J. E. Hopcroft, "Hidden community detection in social networks," *Inform. Sci.*, vol. 425, pp. 92–106, Jan. 2018.

[4] M. E. J. Newman, "Modularity and community structure in networks," *Proc. Nat. Academy of Sci.*, vol. 103, no. 23, pp. 8577–8582, Jun. 2006.

[5] E. Barnes, "An algorithm for partitioning the nodes of a graph," *SIAM J. Algebraic Discrete Methods*, vol. 3, no. 4, pp. 541–550, 1982.

[6] U. von Luxburg, "A tutorial on spectral clustering," *Statist. and Comput.*, vol. 17, no. 4, pp. 395–416, Aug. 2007.

[7] L. Donetti and M. A. Muñoz, "Detecting network communities: a new systematic and efficient algorithm," *J. Statist. Mechanics: Theory and Experiment*, vol. 10, pp. 10012, Oct. 2004.

[8] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys. rev. E*, vol. 69, pp. 026113, Mar. 2004.

[9] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Phys. Rev. E*, vol. 69, no. 6, pp. 066133, Jun. 2004.

[10] Y. Y. Ahn, J. P. Bagrow, and S. Lehmann, "Link communities reveal multiscale complexity in networks," *Nature*, vol. 466, pp. 761–764, Jun. 2010.

[11] I. Derényi, G. Palla, and T. Vicsek, "Clique percolation in random networks," *Phys. Rev. Lett.*, vol. 94, pp. 160202, Apr. 2005.

[12] S. Fortunato, "Community detection in graphs," *Phys. Rep.*, vol. 486, pp. 75–174, Feb. 2010.

[13] S. Parthasarathy, Y. Ruan, and V. Satuluri, "Community discovery in social networks: Applications, methods and emerging trends," in *Social Network Data Analytics*, pp. 79–113. Springer US, Boston, MA, Mar. 2011.

[14] T. Chakraborty, A. Dalmia, A. Mukherjee, and N. Ganguly, "Metrics for community analysis: a survey," *ACM Comput. Surv.*, vol. 50, no. 4, pp. 1–37, Aug. 2017.

[15] P. Doreian, V. Batagelj, and A. Ferligoj, "Generalized blockmodeling," *J. Classif.*, vol. 24, no. 2, pp. 308–311, Sep. 2007.

[16] B. Karrer and M. E. J. Newman, "Stochastic blockmodels and community structure in networks," *Phys. Rev. E*, vol. 83, no. 1, pp. 016107, Jan. 2011.

[17] T. P. Peixoto, "Bayesian stochastic blockmodeling," *ArXiv e-print arXiv: 1705.10225*, May 2017.

[18] E. Abbe, "Community detection and stochastic block models," *Found. and Trends Commun. and Inform. Theory*, vol. 14, no. 1-2, pp. 1–162, Jun. 2018.

[19] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing, "Mixed membership stochastic blockmodels," *J. Mach. Learn. Res.*, vol. 9, pp. 1981–2014, Jun. 2008.

[20] P. K. Gopalan, S. Gerrish, M. Freedman, D. M. Blei, and D. M. Mimno, "Scalable inference of overlapping communities," in *Proc. Adv. Neural Inf. Proc. Systems*, Dec. 2012.

[21] W. Li, S. Ahn, and M. Welling, "Scalable MCMC for mixed membership stochastic blockmodels," in *Proc. Artificial Intell. and Statist.*, May 2016, pp. 723–731.

[22] Y. Zhao, E. Levina, and J. Zhu, "Consistency of community detection in networks under degree-corrected stochastic block models," *Ann. Statist*, vol. 40, no. 4, pp. 2266–2292, Aug. 2012.

[23] C. Gao, Z. Ma, A. Y. Zhang, and H. H. Zhou, "Community detection in degree-corrected block models," *Ann. Statist.*, vol. 46, no. 5, pp. 2153–2185, Oct. 2018.

[24] A. A. Amini, A. Chen, P. J. Bickel, and E. Levina, "Pseudo-likelihood methods for community detection in large sparse networks," *Ann. Statist.*, vol. 41, no. 4, pp. 2097–2122, Aug. 2013.

[25] L. Peng and L. Carvalho, "Bayesian degree-corrected stochastic blockmodels for community detection," *Electron. J. Statist.*, vol. 10, no. 2, pp. 2746–2779, Sep. 2016.

[26] M. Welling and Y. W. Teh, "Bayesian learning via stochastic gradient Langevin dynamics," in *Proc. Int. Conf. Machine Learning*, Jul. 2011, pp. 681–688.

[27] T. Xifara, C. Sherlock, S. Livingstone, S. Byrne, and M. Girolami, "Langevin diffusions and the Metropolis-adjusted Langevin algorithm," *Statist. and Probab. Lett.*, vol. 91, pp. 14–19, Sep. 2014.

[28] M. Girolami and B. Calderhead, "Riemann manifold Langevin and Hamiltonian Monte Carlo methods," *J. R. Stat. Soc. B*, vol. 73, no. 2, pp. 123–214, Mar. 2011.

[29] S. Patterson and Y. W. Teh, "Stochastic gradient Riemannian Langevin dynamics on the probability simplex," in *Proc. Adv. Neural Inf. Proc. Systems*, Dec. 2013.

[30] T. Chen, E. B. Fox, and C. Guestrin, "Stochastic gradient Hamiltonian Monte Carlo," in *Proc. Int. Conf. Machine Learning*, June 2014, vol. 32, pp. 1683–1691.

[31] Y.-A. Ma, T. Chen, and E. B. Fox, "A complete recipe for stochastic gradient MCMC," in *Proc. Int. Conf. Neural Inf. Proc. Systems*, Jul. 2015, vol. 2, pp. 2917–2925.

[32] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Phys. Rev. E*, vol. 74, no. 3, pp. 036104, Sep. 2006.

[33] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graph evolution: densification and shrinking diameters," *ACM Trans. Knowl. Discov. Data*, vol. 1, no. 1, Mar. 2007.