Received: 19 December 2018 | Revised: 22 March 2019 | Accepted: 23 May 2019

DOI: 10.1002/qre.2520

SPECIAL ISSUE ARTICLE

WILEY

Modeling and detecting change in temporal networks via the degree corrected stochastic block model

James D. Wilson [1] | **Nathaniel T. Stevens** [2] | **William H. Woodall** [3]

[1]Department of Mathematics and Statistics, University of San Francisco, San Francisco, California

[2]Department of Statistics and Actuarial Sciences, University of Waterloo, Waterloo, Ontario, Canada

[3]Department of Statistics, Virginia Tech, Blacksburg, Virginia

Correspondence
James D. Wilson, Statistics and Data Science, 107B Harney Science Center, San Francisco, CA 94117.
Email: jdwilson4@usfca.edu

Funding information
Division of Mathematical Sciences, Grant/Award Number: DMS - 1830547

Abstract

In many applications, it is of interest to identify anomalous behavior within a dynamic interacting system. Such anomalous interactions are reflected by structural changes in the network representation of the system. We propose and investigate the use of the degree corrected stochastic block model (DCSBM) to model and monitor dynamic networks that undergo a significant structural change. We apply statistical process monitoring techniques to the estimated parameters of the DCSBM to identify significant structural changes in the network. We apply our surveillance strategy to a dynamic US Senate covoting network. We detect significant changes in the political network that reflect both times of cohesion and times of polarization among Republican and Democratic party members. Our analysis demonstrates that the DCSBM monitoring procedure effectively detects local and global structural changes in complex networks, providing useful insights into the modeled system. The DCSBM approach is an example of a general framework that combines parametric random graph models and statistical process monitoring techniques for network surveillance.

KEYWORDS

network surveillance, process monitoring, quality control, random graphs

1 | INTRODUCTION

Time-varying, or dynamic, networks are often used to model the interactions of a group of actors through time. In many applications, it is of interest to identify anomalous behavior among the actors within a dynamic network. For example, organizers of the Arab Spring uprisings in 2011 tended to interact with one another more frequently on Facebook at the onset of the uprisings.[1] Similarly, central players in the Enron scandal exchanged an increased number of emails prior to fraud investigations.[2] In both of these examples, anomalous activity occurred among the *interactions* of the actors of the system; as a result, these changes can be observed in the network describing the actors.

The monitoring of dynamic networks for anomalous changes through time is known as *network surveillance*. Network surveillance techniques have been successfully applied in a number of settings, including the detection of fraud in large online networks,[3-5] the identification of central players in terrorist groups,[6-8] and the detection of spammers in online social networks.[9] As recent applications of network surveillance have grown in complexity, there has been an increased interest in developing new scalable network surveillance techniques, especially in the area of social network monitoring (see previous studies[10-13] for recent reviews). A useful area to help guide network surveillance is statistical process monitoring (SPM). In general, statistical process monitoring provides a methodology for the real-time surveillance of any characteristic of interest. The philosophy behind SPM is that anomalous behavior in such a characteristic can be identified by distinguishing unusual variation from typical variation in an ordered sequence of observations. Stemming from

applications in industrial manufacturing and public health surveillance, SPM has a rich history and many methods have been developed (see previous studies[14-16] for reviews of methods and applications).

We propose a network surveillance framework that applies statistical process monitoring to the estimated parameters of a dynamic random graph model. We propose the use of a dynamic version of the degree corrected stochastic block model (DCSBM) from Karrer and Newman.[17] The DCSBM yields a probability distribution on the family of undirected graphs with discrete-valued edge weights. Importantly, the DCSBM dictates the propensity of connection between actors and captures two important aspects of social networks: heterogeneous connectivity and community structure. As many monitoring applications involve social communications, eg, the terrorist networks in other works,[3,5] the DCSBM can be used to simulate realistic networks.

The DCSBM is characterized by parameters for which closed-form maximum likelihood estimators (MLEs) can be readily derived. We use statistical process monitoring to identify time points at which the parameter estimates of the DCSBM change significantly. Here, we investigate two widely studied SPM methods for surveillance, the Shewhart control chart for individual observations and the exponentially weighted moving average (EWMA) control chart.[18] We apply our surveillance strategy to the dynamic covoting network of the US Senate, which models the voting behavior of US Senators from 1867 to 2015. We find that our surveillance strategy is able to identify eras of cohesion and division among the Republican and Democrat parties, and that these changes coincide with significant political events in US history. These analyses, as well as our simulation study, reveal that our network surveillance method with the DCSBM is an effective monitoring strategy for dynamic networks that undergo change.

Our proposed monitoring strategy establishes one practically useful technique among a general family of methods for surveillance. Our framework relies on two components: a parametric dynamic random graph model for modeling the features of the graph and control charts from statistical process monitoring for the detection of significant change in the model's parameters. Here, we consider the DCSBM random graph model and the Shewhart and EWMA control charts for surveillance. However, this same framework can be used for any parametric random graph model and any control chart of the user's choice. For example, one could investigate dynamic exponential random graph models like those described by other works[19,20] or dynamic latent space models such as that introduced in Sewell and Chen.[21] Furthermore, one could further investigate the use of other univariate SPM methods such as cumulative sum (CUSUM) control charts or control charts for attributes and perhaps multivariate SPM approaches such as Hotelling T^2 or multivariate EWMA control charts.[18] Our contribution serves as a first step in understanding the utility of this framework.

2 | THE NETWORK SURVEILLANCE PROBLEM

Consider a collection of actors or individuals $[n] = \{1, \ldots, n\}$, whose interactions have been recorded at times $t = 1, \ldots, T$. In many applications, it is convenient to represent the interactions of $[n]$ at time t by an undirected network $G_t = ([n], W_t)$. Here, the actors $[n]$ are treated as *nodes* or *vertices* in the graph, and $W_t = \{w_{u,v}(t) : u, v \in [n]\}$ is the set of *edge weights*, where $w_{u,v}(t)$ quantifies the strength of the relationship between nodes u and v at time t. A dynamic network model of the individuals $[n]$ over time $t = 1, \ldots, m$ is the ordered sequence of undirected graphs $\boldsymbol{G}(n, T) = \{G_1, \ldots, G_T\}$. The edge weight $w_{u,v}(t)$ may, for example, represent the number of communications between individuals u and v at time t in a dynamic social network, or the number of interactions between two genes u and v at time t in a biological network.[22] Note that an unweighted graph, where each edge weight is binary, is a special case where edges indicate the presence or absence of a specified level of connection between nodes u and v at time t.

The goal of network surveillance is to prospectively monitor the interactions of $[n]$ so as to detect abnormal behavior among the actors. To perform surveillance, one generally first specifies a statistic S_t, or more generally a vector of statistics $\boldsymbol{S}_t$, that provides some local or global summary of the network G_t based on the types of anomalies to be detected. The choice of S_t is flexible. In the simplest case, one can choose a statistic that summarizes some topological aspect of G_t, such as the connectivity of each node, the clustering of nodes, or the average shortest distance between each pair of nodes.[23-26] In many cases, the choice of statistic is driven by the application, such as the Enron email network analysis in.[26] Alternatively, one can model G_t by a family of probability distributions governed by parameters $\boldsymbol{\Psi}$ and specify S_t as an estimator associated with $\boldsymbol{\Psi}$. We discuss such model-based approaches in more detail in Section 3.

Once a statistic S_t has been chosen, SPM is used to distinguish unusual behavior from typical behavior. In network surveillance, this corresponds to the real-time identification of unusually large or small values of S_t. For this purpose, we use a *control chart*—a time series plot of S_t constructed with *control limits* that indicate boundaries of typical behavior.

An observed value of S_t is considered anomalous if it deviates significantly from what previous observations suggest is typical. Monitoring consists of two phases, *Phases I* and *II*, which are described below.

a. Phase I: The statistic S_t is calculated for the first $m < T$ graphs. The mean μ and variance σ^2 of S_t are estimated using the m sampled statistics. A tolerance region $\mathcal{R}(\hat{\mu}, \hat{\sigma}^2)$ is constructed on the basis of the estimated values for μ and σ^2. The upper and lower bounds of this region are referred to as upper and lower control limits, respectively. Variation within these limits defines typical behavior.
b. Phase II: For each new graph G_t, with $t > m$, S_t is calculated, and G_t is deemed "typical" if $S_t \in \mathcal{R}(\hat{\mu}, \hat{\sigma}^2)$ and deemed "anomalous" otherwise. When an observed value of S_t exceeds these limits, we say that the control chart has *signalled*; this serves as an indication that a structural change has occurred.

As the first m networks are used to determine the tolerance region $\mathcal{R}(\hat{\mu}, \hat{\sigma}^2)$, successful monitoring in Phase II requires that the data in Phase I provide an accurate representation of typical variation; if μ and σ^2 are not accurately estimated, then the control limits defined by $\mathcal{R}(\hat{\mu}, \hat{\sigma}^2)$ are unlikely to be applicable beyond the Phase I time frame. Ideally, the control limits will balance the need for a control chart that is sensitive enough to detect important changes, while not signalling too frequently when no anomaly is present and creating an excessive number of false alarms. Jones-Farmer et al[27] discussed the importance of effectively collecting and analyzing baseline data during Phase I. If the network being monitored is expected to evolve over time, then we recommend moving window approaches as opposed to a fixed Phase I sample as considered in other studies.[28,29]

The performance of a surveillance technique depends also on the definition of $\mathcal{R}(\hat{\mu}, \hat{\sigma}^2)$, which largely depends on the goal of the control chart and the type of data being plotted. Abnormal activity in Phase II networks may be brief—where as few as one or two anomalous graphs are observed, or it may persist over an extended period of time. To detect sudden large changes, a standard Shewhart control chart is typically used.[18] However, if sensitivity to sustained small and medium-sized changes is of interest, one might consider using an exponentially weighted moving average (EWMA) control chart. See other studies[30,31] for recent advances in EWMA control chart techniques.

In practice, the choice of statistic S_t and type of control chart will depend on the types of network changes one wishes to detect. For instance, if one seeks to detect a global change in the network (where there is an overall change in the structure, eg, communications on average increase or decrease over the entire network), the choice of statistic and chart will be different than if one wishes to detect a local change in the network (where there is a change in structure among some subgraph of the network, eg, communications on average increase or decrease within a particular community). In Section 7, we use simulation to evaluate the ability of our proposed methodology to detect a variety of different local and global network changes.

3 | RELATED WORK

There are other model-based approaches for network surveillance that have been recently developed. Azarnoush et al[28] proposed a longitudinal logistic model that describes the (binary) occurrence of an edge at time t as a function of time-varying edge attributes in the sequence of networks $\boldsymbol{G}([n], T)$. A likelihood ratio test is used to monitor changes before and after each time point. Peel and Clauset[32] developed a generalized hierarchical random graph model (GHRG) to model $\boldsymbol{G}([n], T)$. To detect anomalies, the authors used the GHRG as a null model to compare observed graphs in $\boldsymbol{G}([n], T)$ via a Bayes factor. At each time t, Bayesian posterior inference via Markov Chain Monte Carlo is used to fit the GHRG to the graph G_t. Anomalies are detected using a sliding window approach on the Bayes factor that compares observed graphs to the GHRG fit for previous observations.

In Heard et al,[33] the authors considered monitoring changes in communication volume between subgroups of targeted people over time. Their approach evaluates pairwise communication counts and determines whether these have significantly increased using a P value. The P value assesses the deviation of the communication rate at time t and what is considered normal behavior under conjugate Bayesian models describing the discrete-valued time series of communications up to time t. While their focus is detecting changes on the entire network, our approach considers detecting anomalies for members of a community within a dynamic network. Sparks and Wilson[31] considered the monitoring of abrupt changes among an unknown set of actors in a dynamic network. They establish an EWMA strategy for detecting such changes, which incorporates the uncertainty of the type and size of the subset of actors undergoing a change.

The change point approach developed in Barnett and Onnela[34] seeks to identify significant changes in correlation networks, where the correlation network at time t represents the correlation of some underlying multivariate stochastic

process at that time. For each t, the Frobenius distance $F(t, t^-)$ between the correlation network at time t and the average of the correlation networks from times $1, \ldots, t-1$ is calculated. The authors then generate a sample of "null" networks by bootstrapping a sample of t networks where no change is introduced. There are a number of sequential change point methodologies that have been developed for networks recently (see, for example, Zambon et al[35]). Chen[36] introduced a nonparametric method for detecting change via a k nearest neighbor approach. Roy et al[37] considered change point analysis in high-dimensional Markov fields, and Keshavarz et al[38] extended this work to Gaussian graphical models.

Recent work has begun to focus on the problem of network monitoring with the use of control charts. Yu et al[39] extended the present work to monitor varying degree propensity coefficients. Hosseini and Noorossana[40] considered the application of the CUSUM and EWMA control charts on the average and standard deviation of the degree of each node in an observed network. Perry[41] developed an EWMA control chart to monitor the hierarchy present in directed networks. Both Zhao et al[29] and Noorossana[42] analyzed the use of control charts for dynamic anomaly detection in social networks.

4 | THE DEGREE CORRECTED STOCHASTIC BLOCK MODEL

Let $G = ([n], W)$ be an undirected network that represents the interactions of actors $[n]$. The DCSBM models two important features of real networks: (a) community structure and (b) degree heterogeneity, which we now briefly discuss.

Empirically, the nodes of a network G can often be divided into $k \geq 1$ disjoint vertex sets as $[n] = V_1 \cup V_2 \ldots \cup V_k$ in such a way that the density of edges within each vertex set $V_j \subseteq [n]$ is substantially greater than the density between differing sets. The vertex sets are commonly referred to as *communities*. In many applications, the communities of a network provide structural or functional insights about the modeled complex system. For example, recently community structure has been used to help develop hypotheses about gene interactions and antibiotic resistance,[43] and about the dynamics of social interactions using cell phone data.[44] The substantial relevance of communities in network systems has lead to a large and growing literature about community structure and the identification of statistically meaningful communities (see other studies[45,46] for reviews).

In addition to naturally dividing into densely connected communities, actors in a network tend to have a highly variable propensity to make connections. In these situations, the *degree distribution* of the nodes are variable, where the degree d_u of a node $u \in [n]$ is the total number of interactions in which u takes part, namely, $d_u = \sum_{x \in [n]} w_{u,x}$.

The scale-free family of networks is one common family of networks with heterogenous degrees. In scale-free networks, the degree distribution approximately follows a power law.[47,48] Scale-free networks commonly arise in economic, social, and ecological networks (eg, Kasthurirathna and Piraveenan[49] studied a recent example).

4.1 | The model

Let $\mathcal{G}$ represent the family of all undirected networks with n nodes and k disjoint communities. The DCSBM yields a probability distribution $\mathbb{P}(\cdot) = \mathbb{P}(\cdot|\boldsymbol{\theta}, \boldsymbol{\pi}, P)$ on $\mathcal{G}$ that is characterized by (a) nonnegative degree parameters $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_n)$, which reflect the tendency of the nodes to connect, (b) containment probabilities $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_k)$ that satisfy $\pi_r > 0$ and $\sum_{r \in [k]} \pi_r = 1$, where π_r specifies the probability of a node belonging to community r, and (c) the $k \times k$ symmetric connectivity matrix $P = (P_{r,s})$, where entries $P_{r,s} > 0$ represent the propensity of connection between nodes in communities r and s.

Let $G \in \mathcal{G}$ be a random graph with n nodes and k communities generated under $\mathbb{P}$. Then G can be obtained by a simple generative procedure, described as follows. First, parameters $\boldsymbol{\theta}$, $\boldsymbol{\pi}$, and P are prespecified and fixed. These are chosen to control the degree variability, relative size of communities, and connection propensity between and within communities, respectively. Nodes are randomly assigned community labels $\boldsymbol{c} = (c_1, \ldots, c_n)$ according to the multinomial draws:

$$c_u \overset{\text{i.i.d}}{\sim} \text{Multinomial}(1, \boldsymbol{\pi}). \tag{1}$$

Given $\boldsymbol{\theta}$, $\boldsymbol{c}$, and P, edge weights $\{w_{u,v} : u, v \in [n]\}$ are assigned according to independent Poisson draws, where

$$\mathbb{E}[w_{u,v}|\boldsymbol{c}, \boldsymbol{\theta}, P] = \theta_u \theta_v P_{c_u, c_v}. \tag{2}$$

The graph G is then defined as the network with nodes $[n]$, community labels $\boldsymbol{c}$, and edge weights $\boldsymbol{w} = \{w_{u,v} : u, v \in [n]\}$ resulting from (1) and (2). For an observed network with community labels $\boldsymbol{c}$ and edge weights $\boldsymbol{w}$, we define n_r as the number of vertices in community r. Further define $m_{r,s} = \sum_{u,v} w_{u,v} \mathbb{I}(c_u = r, c_v = s)$ as the total weight of edges between

community r and s (twice the weight of edges when $r = s$). It follows by combining (1) and (2) that the joint distribution of the random graph G and community labels $\boldsymbol{c}$ is described by the joint probability mass function given by, when ignoring constants,

$$\begin{aligned}\mathbb{P}(G, \boldsymbol{c}|\boldsymbol{\theta}, \boldsymbol{\pi}, P) \propto & \prod_{r\in[k]} \pi_r^{n_r} \prod_{u\in[n]} \theta_u^{d_u} \prod_{u<v\in[n]} \frac{1}{w_{u,v}!} \\ & \times \prod_{r,s\in[k]} P_{r,s}^{\frac{m_{r,s}}{2}} e^{-\frac{n_r n_s P_{r,s}}{2}}.\end{aligned} \tag{3}$$

The distribution $\mathbb{P}(G|\boldsymbol{\theta}, \boldsymbol{\pi}, P)$ is obtained by summing the joint probability in (3) over all possible realizations of $\boldsymbol{c}$. We note that the model in (3) is not identifiable without some constraint on $\boldsymbol{\theta}$ since the likelihood is unaffected by certain opposing magnitude shifts in $\boldsymbol{\theta}$ and P.[50] To ensure that the model is identifiable, we require that the sum of θ_u in the same community equal the number nodes in that community, namely,

$$\sum_{u:c_u=r} \theta_u = n_r, \tag{4}$$

for all $r = 1, \ldots, k$. For simulation, it is often of interest to specify the community labels $\boldsymbol{c}$ deterministically rather than randomly as in (1). To distinguish these assignment strategies, we will write $\mathbb{P}(\cdot|\boldsymbol{\theta}, \boldsymbol{c}, P)$ to represent the probability distribution of the DCSBM when the community labels are prespecified a priori.

4.2 | Simulating a DCSBM with a structural change

We are interested in simulating an ordered sequence of graphs on the vertex set $[n]$ that demonstrate various types of significant structural change. The DCSBM $\mathbb{P}(\cdot|\boldsymbol{\theta}, \boldsymbol{c}, P)$ provides a flexible way to model change in a sequence of random graphs. To model a sequence of graphs with a significant structural change, we generate an ordered sequence of random graphs $\boldsymbol{G}(n, T) = \{G_1, \ldots G_{t^*}, \ldots, G_T\}$ according to

$$G_t \sim \begin{cases} \mathbb{P}(G|\boldsymbol{\theta}^0, \boldsymbol{c}^0, P^0), & t < t^* \\ \mathbb{P}(G|\boldsymbol{\theta}^*, \boldsymbol{c}^*, P^*) & t \geq t^* \end{cases}. \tag{5}$$

By simulating $\boldsymbol{G}(n, T)$ as in (5), we introduce a structural change in the graph at time t^* that persists across the remaining networks in the sequence. In this way, $\boldsymbol{G}_1 = \{G_1, \ldots, G_{t^*-1}\}$ are simulated as "typical" graphs, whereas $\boldsymbol{G}_2 = \{G_{t^*}, \ldots, G_T\}$ are "anomalous" graphs. The goal of a surveillance method then is to signal as quickly as possible following the time point of change t^*. For network monitoring simulations, we require $t^* > m$ so that the change occurs after Phase I. We note that in principle one can simulate networks with multiple changes, as well as networks with changes that affect a small number of networks.

To simulate a network $\boldsymbol{G}(n, T)$ according to (5), one first chooses $\boldsymbol{\theta}^0, \boldsymbol{c}^0, P^0$ and $\boldsymbol{\theta}^*, \boldsymbol{c}^*$, and P^* either stochastically or deterministically while maintaining constraints on each parameter. In particular, P^0 and P^* must contain entries on the unit interval, $\boldsymbol{c}^0$ and $\boldsymbol{c}^*$ must be positive discrete-valued labels $\{1, \ldots, k\}$, and $\boldsymbol{\theta}^0$ and $\boldsymbol{\theta}^*$ must satisfy (4) to ensure identifiability. For each $t \in \{1, \ldots, T\}$, let $(\theta_u, c_u, P_{u,v})$ be entries of $(\boldsymbol{\theta}^0, \boldsymbol{c}^0, P^0)$ if $t < t^*$ or $(\boldsymbol{\theta}^*, \boldsymbol{c}^*, P^*)$ otherwise. The edges of graph G_t is generated as independent Poisson draws

$$w_{u,v} \sim \text{Poisson}\left(\theta_u \quad \theta_v \quad P_{c_u,c_v}\right),$$

and each graph is generated independently.

The changes $\boldsymbol{\theta}^0 \to \boldsymbol{\theta}^*$, $\boldsymbol{c}^0 \to \boldsymbol{c}^*$, and $P^0 \to P^*$ each reflect a different type of structural change in the simulated dynamic network. By altering the parameters that dictate the DCSBM from time $t^* - 1$ to t^*, we are able to model several types of structural change among the actors $[n]$ in $\boldsymbol{G}(n, T)$, including the following:

a. Change in rates of interaction: In general, one can introduce a mean shift in interaction rate in community r by specifying $P^*_{r,r} \neq P^0_{r,r}$. Doing so will also affect the variance of the interaction rate in the community. In particular, the mean and variance of the number of interactions in community r will decrease at time t^* when $P^*_{r,r} < P^0_{r,r}$, and increase when $P^*_{r,r} > P^0_{r,r}$. One can introduce a change in variance of the interaction rate of vertices locally or globally by specifying $\boldsymbol{\theta}^* \neq \boldsymbol{\theta}^0$.

b. Communication outbreaks: In network surveillance, one is often interested in identifying "communication outbreaks" among the members of some subgraph $\Omega \subseteq [n]$ in the network. A communication outbreak corresponds to an increase in the average number of interactions among the members of Ω. Using the DCSBM, we can model communication outbreaks among any number of communities in the network. For example, a communication outbreak among the members of community j is modeled by specifying $P^*_{r,r} > P^0_{r,r}$ as the mean and variance of the interactions in community r will increase at time t^*. We can model a *global* communication outbreak by specifying $P^*_{r,s} > P^0_{r,s}$ for all $r, s \in [k]$.
c. Change in community structure: A change in community structure of a social network can signify an important transition in the modeled system. For example, in the political voting network we consider in Section 6, the network structure associated with the members of the US Senate significantly changes at times of extreme polarization of the Republicans and Democrats.[51,52] describe six general types of community structure changes in a network, including growth, shrinkage, birth, death, the merging of two communities, or the splitting of a single community into two or more communities. In general, each of these types of changes can be implemented at time t^* by specifying new community labels $\boldsymbol{c}^* \neq \boldsymbol{c}^0$.

5 | MONITORING THE DCSBM

Suppose that we observe a dynamic graph sequence $\boldsymbol{G}(n, T) = \{G_1, \ldots, G_T\}$ that is generated under the dynamic DCSBM according to (5). Our goal is to identify as quickly as possible any change in the distribution that generated $\boldsymbol{G}(n, T)$. To detect such changes, we propose a surveillance strategy that proceeds in two steps. First, the dynamic DCSBM is fitted to $\boldsymbol{G}(n, T)$ using maximum likelihood estimation. Next, control charts are applied to functions of these maximum likelihood estimators to detect changes. Here, we apply the Shewhart and EWMA control charts for individuals. We first describe estimation of the DCSBM and then our monitoring strategy.

5.1 | Fitting the DCSBM

5.1.1 | Estimation of communities

The estimation of the community labels $\boldsymbol{c}$, otherwise known as *community detection*, is known to be an NP hard problem; as a result one must estimate the labels using an approximate algorithm. Many detection methods have been developed for weighted and unweighted networks (see other studies[45,46] for reviews). The spectral clustering algorithm[53] is particularly well suited for this setting due to its theoretical guarantees,[54,55] which we now briefly mention.

Let m represent the number of Phase I graphs in $\boldsymbol{G}(n, T)$, and assume that $m < t^*$. Define the average Phase I graph by $\bar{G} = \frac{1}{m}\sum_{j=1}^{m} G_j$, where the sum of two graphs $G_1 = ([n], W_1)$ and $G_2 = ([n], W_2)$ is the graph with node set $[n]$ and edge weights $W_1 + W_2$. If the probability matrix P has no identical rows, then spectral clustering of the graph $\bar{G}$ will provide asymptotically consistent community label estimates $\hat{c}$, as $m \to \infty$. In other words, if the number of Phase I graphs is large enough, we obtain consistent estimators for the community structure for the sequence of graphs before t^*. This fact is a consequence of the main result presented in Han et al.[54] In practice, one should use as many Phase I graphs as possible, but the choice of m depends on the judgement of the practitioner and the availability of data.

For monitoring purposes, we suggest using the regularized spectral method from Qin and Rohe[55] on the Phase I graphs in the sequence and monitoring the parameter estimates conditional on the estimated community labels for the *entire* sequence of graphs. As we will see, in many cases changes in the community structure will be reflected by changes in the parameter estimates describing the DCSBM. Though we do not pursue it here, future work should investigate surveillance of community labels themselves.

5.1.2 | Maximum likelihood estimation of parameters

We now briefly summarize the maximum likelihood estimation of the parameters in the DCSBM, which was derived in Yan et al.[50] We assume that $\boldsymbol{c}$ is fixed for all t and is equal to the estimators $\hat{\boldsymbol{c}}$ obtained from spectral clustering described above. From (3), we can show that the log likelihood of $(\boldsymbol{\theta}, P)$ given an observed graph $G = ([n], W)$ and community labels is, when ignoring constants,

$$\ell(\boldsymbol{\theta}, P | G, \boldsymbol{c}) \propto \sum_{u \in [n]} d_u \log(\theta_u) + \frac{1}{2} \sum_{r,s \in [k]} \left(m_{r,s} \log(P_{r,s}) - n_r n_s P_{r,s} \right). \tag{6}$$

Taking derivatives, it is readily shown from (6) that the maximum likelihood estimator (MLE) for each parameter has a closed-form solution. For $u \in [n]$ and $r, s \in [k]$, the maximum likelihood estimators are given by

$$\hat{\theta}_u = \frac{d_u}{n_r^{-1} \sum_{w : c_w = c_u} d_w}, \qquad \hat{P}_{r,s} = \frac{m_{r,s}}{n_r n_s}. \tag{7}$$

5.2 | Monitoring strategy

To develop a monitoring strategy that detects local and global changes in a network, we first suppose that the number of communities, k, is fixed through time. The community labels at each time point, $\boldsymbol{c}_t$, is first estimated. Given $\boldsymbol{c}_t$, we directly monitor the MLE $\hat{P}$, where at each time t we estimate the $\binom{k}{2}$ unique entries of $\hat{P}$ for graph G_t. This statistic reflects the overall connection propensity among communities. To monitor for changes in $\boldsymbol{\theta}$, one could in principle monitor each statistic $\hat{\theta}_u$ separately; however, this can lead to a large number of control charts. Instead, for community $r = 1, \ldots, k$, we monitor the sample standard deviation of the propensity estimates in each community $\{\hat{\theta}_j : c_j = r\}$, denoted by s_r. Our choice in using the standard deviation is motivated by the fact that subject to (4), the expectation of $\{\theta_u : c_u = r\}$ is fixed to be exactly 1. Thus, we decide instead to monitor the variability in overall connection within community r.

We note that it is possible for s_r to remain fixed while the propensity parameters change. For example, in yet to be published work[39] define a θ value for each individual within a community, and treat these propensities as fixed parameters to be modeled and monitored. Their focus is the detection of change in individual connection propensities within communities.

In summary, our surveillance plan monitors $\binom{k}{2} + k$ statistics $\{\hat{P}_{q,r}, \quad s_q : q \le r \in [k]\}$ through time. We expect these statistics to capture community structure changes as well, since in this scenario the mean connectivity between pairs of nodes in the network is also likely to change. We note that we monitor the parameters with separate charts because the maximum likelihood estimators given by (7) are asymptotically independent. For small networks, it may be beneficial to apply a multivariate control chart; however, in our applications, we apply separate control charts for ease of interpretation. Furthermore, we note that since we monitor characteristics of communities over time and not the nodes themselves, we can directly apply our monitoring strategy to a sequence of networks with a varying number of nodes.

5.2.1 | Shewhart control chart

For each of the parameters that we estimate, we use Shewhart and EWMA control charts to determine what values of these parameters indicate a significant change in the network. Let S_t be a statistic at time t, and let m be the number of Phase I networks. For $t > m$, the Shewhart control chart for individual outcomes signals a change in the statistic if S_t lies outside of the control limits $\hat{\mu} \pm 3\hat{\sigma}$, where $\hat{\mu}$ is the sample mean of the m Phase I observations, and $\hat{\sigma}$ is the moving range estimate for the standard deviation of these m observations given by

$$\hat{\sigma} = \frac{\sqrt{\pi}}{2(m-1)} \sum_{j=2}^{m} |S_j - S_{j-1}|.$$

Note that the constant $2/\sqrt{\pi}$ is equivalent to d_2, the normalization constant used in the control chart literature.

5.2.2 | EWMA control chart

Whereas the Shewhart control chart is designed to detect sudden large changes in S_t, the width of the $\pm 3\hat{\sigma}$ limits and the use of only the most recent statistic value results in reduced sensitivity to persistent changes that are small to medium in size. When detection of these sorts of changes is of interest the EWMA control chart is to be preferred over the Shewhart control chart.

Instead of plotting the observed values of S_t directly, for $t > m$ the EWMA control chart is a time series plot of Z_t, the exponentially weighted moving average of S_t

$$Z_t = \lambda S_t + (1 - \lambda) Z_{t-1},$$

where $Z_0 = \hat{\mu}$ is a common choice for the starting value of the moving average and λ ($0 < \lambda \leq 1$) is a smoothing constant. Through empirical investigation, Crowder[56] provides guidance on the choice of λ that optimizes the performance of the EWMA control chart. Montgomery[18] suggests that values of λ in the interval $0.05 \leq \lambda \leq 0.25$ work well in practice with $\lambda = 0.2$ being a popular choice. The control limits of the EWMA control chart are given by

$$\hat{\mu} \pm 3\hat{\sigma}\sqrt{\frac{\lambda}{(2-\lambda)}[1-(1-\lambda)^{2t}]}.$$

Note that as t increases, ie, as the number of Phase II observations increases, these control limits approach the steady-state values given by

$$\hat{\mu} \pm 3\hat{\sigma}\sqrt{\frac{\lambda}{(2-\lambda)}}. \tag{8}$$

If Z_t lies outside these control limits, it signals that a small and persistent change has occurred. Because the current observation S_t is de-emphasized in this moving average, the EWMA control chart will not signal sudden large changes as quickly as a Shewhart control chart. Thus the nature of change one wishes to detect should dictate which control chart is used. In practice, it is sensible to simultaneously monitor S_t using both approaches. We explore the utility of both the Shewhart and EWMA control charts when applied to the U.S. Senate covoting network in Section 6. We also use simulation to investigate the detection properties of the Shewhart control chart more generally, in Section 7.

6 | APPLICATION TO THE US SENATE VOTING NETWORK

We now use the DCSBM surveillance procedure to investigate the dynamic relationship between Republican and Democrat Senators in the US Congress. We analyzed the covoting network of the US Senate from 1867 (Congress 40) to 2015 (Congress 113). This network was first analyzed in Moody and Mucha[51] and has been since investigated in Roy et al.[37] In Moody and Mucha,[51] the modularity, or extent of divisiveness, of the network was calculated over time, and it was found that generally Republicans and Democrats have become more polarized over time. The dynamic DCSBM framework provides a means to formally model this network and test for changes in the community structure and voting patterns among party members.

We generated a dynamic network to model the covoting patterns among US Senators in the following manner. We first collected the roll call voting data for each Congress from http://voteview.com. This data set contains the voting decision (either yay, nay, or abstain) of each Senator for every bill submitted to the Senate. For each Congress, we model the Senators in that Congress as the collection of nodes. Binary edges are placed between two Senators if they vote concurrently (either both yay or both nay) for at least 75% of the total number of bills on which either of them voted.

To analyze political polarization, we applied the DCSBM surveillance strategy with Shewhart and EWMA control charts to this dynamic network. Since node labels across graphs are not registered, ie, nodes do not represent the same Senators across time, estimating the community labels using the spectral clustering strategy mentioned in Section 5.1.1 is not appropriate. As we are interested in understanding political polarization, we instead set the community labels at time t according to the political affiliation of each Senator (1 for Democrat and 2 for Republican). We set the Phase I size to be $m = 25$ and computed the Shewhart and EWMA control charts for the estimators $\{\hat{P}_{q,r}, s_q : q, r = 1, 2\}$. For the EWMA chart, we calculated the control limits in (8) and set $\lambda = 0.2$. The Shewhart and EWMA control charts are shown in the bottom of Figure 1.

The control charts in Figure 1 reveal three interesting and relevant features about the US Senate voting patterns. First, both the Shewhart and EWMA control charts signal large values of $\hat{P}_{1,2}$ from Congress 91 (1969-1971) to Congress 94 (1975-1977). This finding suggests that Republicans and Democrats tended to vote concurrently more often than expected during this period of time. Furthermore, the EWMA control chart signals large values of s_1 during this time period. This suggests that the voting propensity of the Democratic party during this time is significantly more variable than expected. Interestingly, this time frame lies at the second half of the so-called "Rockefeller Republican" era, which lasted from 1960 to 1980. During this era, many Republican Senators had moderate views that reflected the ideals of the governor of New York, Nelson Rockefeller.[57,58] The Rockefeller Republicans were strong supporters of the civil rights movement, including the Civil Rights Act of 1968, and held especially moderate fiscal views under the Presidency of Richard Nixon (93rd Congress). Notably, this general cohesion among parties—marked by large values of $\hat{P}_{1,2}$ in the control charts—ended in

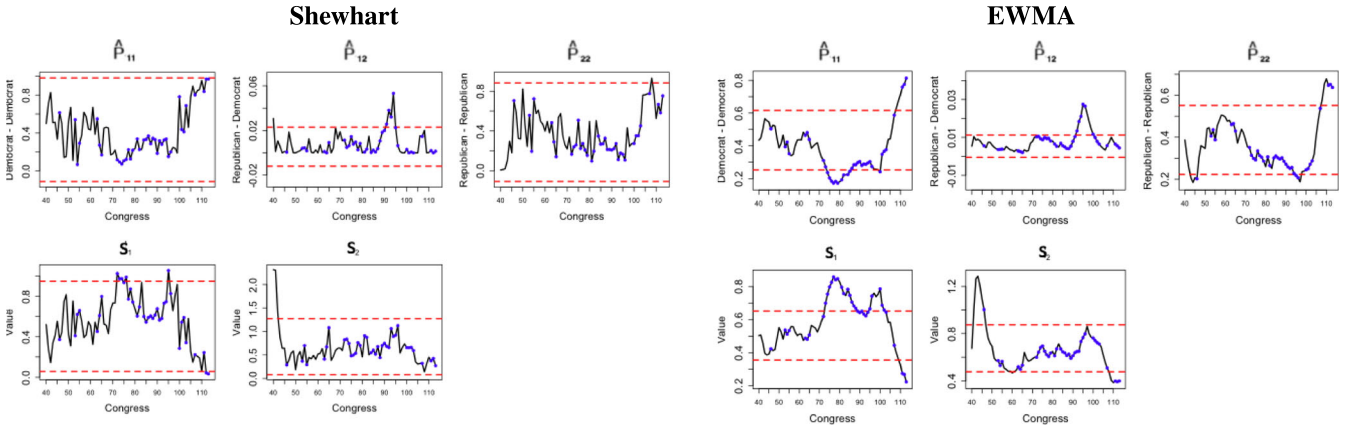

FIGURE 1 Shewhart and EWMA control charts for the DCSBM coefficient estimates for the US Senate covoting network. Red dashed lines represent the upper and lower control limits for each control chart. Red dots represent weeks that were signalled for that control chart. Blue dots signify when the Democratic party held the majority in the Senate [Colour figure can be viewed at wileyonlinelibrary.com]

Congress 94. This Congress coincides with the end of Nelson Rockefeller's role as Vice President of the United States in 1977. To the best of our knowledge, this is the first work to identify this political era using Senatorial covoting data.

Next, the EWMA control charts for $\hat{P}_{1,1}$ and $\hat{P}_{2,2}$ signal large values at Congress 104. This suggests that the intraparty covoting propensities for both the Democratic and Republican parties became exceedingly large at that time. This finding supports the theory of recent polarization of the parties at the beginning of Bill Clinton's first term as President (Congress 103). According to Moody and Mucha,[51] this time period marked an important transition at which conservative Democrats and liberal Republicans joined majority-party coalitions in both Congress 103 (Democratic majority) and Congress 104 (Republican majority). This transition left the middle ground between parties empty, which may have lead to an enduring polarization. These results also coincide with the findings of Roy et al.[37] The Shewhart control chart did not as clearly signal this change; however, in each of the charts, there is an increasing trend beginning in Congress 100.

Finally, the EWMA control charts for s_1 and s_2 signal a significantly small value of these statistics at Congress 105. This suggests that the variability of total interaction of the Senators steadily and significantly become lower during this period. This finding complements the polarization theory described above, and suggests that since Congress 105, each US Senator tends to vote according to his or her party, regardless of the bill.

7 | SIMULATION STUDY

In this section, we investigate the detection of structural changes in a network $\boldsymbol{G}(n, T) = \{G_1, \ldots, G_T\}$ generated under a DCSBM with a structural change. We consider local and global changes in the network as parameterized by changes in P, $\boldsymbol{\theta}$, and $\boldsymbol{c}$ at time t^*. In this study, we consider the Shewhart control chart for individuals; however, we expect the EWMA control charts to behave similarly.

In Section 7.1, we first evaluate this monitoring strategy on a collection of illustrative examples to gain an intuition of the DCSBM and the performance of the proposed methodology. In Section 7.2, we quantify the strengths and weaknesses of our method using an analysis of *average run lengths* under a variety of simulated conditions. To evaluate the performance of our detection strategy, we altered the network size and the magnitude of the change being introduced. This simulation strategy can be readily used to assess the performance of any network surveillance method.

We note that this simulation study assumes that the community membership of each node is known and so we do not use a community detection algorithm such as those described in Section 5.1.1 prior to surveillance. We make this choice deliberately for the following reason. In practice, a user of the proposed methodology begins by identifying a community partition with an algorithm of their choice. Then, trusting this partition is accurate, the user continues with surveillance assuming community membership is known. In line with this, our simulation results provide information about the expected performance of our methodology *after* community membership has been established. To investigate the influence of community detection on the surveillance strategy is to evaluate the community detection algorithm itself—which is not our focus. Thus, we provide useful insight for the realistic situation in which surveillance is performed assuming community labels, whether assigned deterministically or determined algorithmically, are known.

7.1 | Illustrative examples

We begin our simulation study by demonstrating the Shewhart control charts on a collection of six dynamic networks, each of which reflects a different structural change at time t^*. We investigated changes in the mean and variance of interaction rate, both locally and globally, as well as changes in community structure. For each simulation, we generated a dynamic network according to (5) with $n = 50$ nodes, $k = 2$ equally sized communities, $T = 50$ time points, and a change implemented at time $t^* = 30$. We use the first $m = 25$ simulated networks for Phase I, and implemented the Shewhart control chart for the statistics $\{\hat{P}_{q,r}, s_q : q, r = 1, 2\}$ using the surveillance strategy described in Section 5.2. In all six simulations, we generate $\boldsymbol{\theta}^0$ as random draws from a Uniform distribution where

$$\theta_u^0 \overset{i.i.d}{\sim} U(1 - \delta_{c_u}^0, 1 + \delta_{c_u}^0),$$

and δ_r is a constant multiple of the standard deviation of the connection propensity among the nodes in community r. Furthermore, we set

$$P^0 = \begin{pmatrix} 0.2 & 0.1 \\ 0.1 & 0.2 \end{pmatrix}, \delta_1^0 = \delta_2^0 = 0.5.$$

Control charts are shown for each simulation in Figure 2. Below, we describe the six simulated networks and the results of our monitoring plan. To conserve space, we do not present charts for s_2, and instead describe them qualitatively where appropriate. The implemented changes for each simulation are described in Table 1.

Simulations 1-2: mean interaction rate changes In the first two simulations, we monitored changes in the mean interaction rates in the network. In simulation 1, we introduce a local mean interaction outbreak in community 1 by setting $P_{1,1}^* = P_{1,1}^0 + \epsilon$ with $\epsilon = 0.10$. The top of Figure 2 reveals that the control chart for $\hat{P}_{1,1}$ efficiently signals a change at time 30, whereas, all other statistics remain in control over the entire time interval. In simulation 2, we introduce a global mean interaction outbreak by increasing all entries of P by $\epsilon = 0.10$. In this case, the probability estimates $\hat{P}_{1,1}$, $\hat{P}_{1,2}$ and $\hat{P}_{2,2}$ all lead to a signal for a change at time 30, and s_1 and s_2 remain in control. (The chart for s_2 is not shown here.) We note that $\hat{P}_{1,2}$ appears to signal the most dramatic change. This is due to the fact that the signal to noise ratio introduced by increasing the overall interaction rate in the network is highest for the inter community interactions.

Simulations 3-4: variance of interaction rate changes Next, we monitored changes in the variation of the interaction rate in the simulated network. In simulation 3, we increased δ_1^0 by $\tau = 0.25$, which results in a change in the variability of interaction in community 1. The top of Figure 2 reveals that this change is indeed signalled by the s_1 chart. We expect the reaction of the chart, and hence the signal delay, to depend on the magnitude of change. We investigate this further in the next section. In simulation 4, we simulated a global change in $\boldsymbol{\delta}^0 = (\delta_1^0, \delta_2^0)$, which increases the variability of interactions among all nodes. In this case, δ_1^0 and δ_2^0 were both increased by $\tau = 0.25$. The bottom of Figure 2 reveals that s_1 signals the change almost immediately. Although not shown here, the control chart for s_2 behaves similarly. Importantly, the connection probability estimates remain in control in these simulations suggesting, as desired, that the mean interaction rate in the network did not change.

Simulations 5-6: change in community structure In simulations 5 and 6, we consider two common changes in community structure: merging and splitting of communities. In simulation 5, we simulated networks with two equally sized communities up to time $t^* = 30$. At time t^*, we then merged the two communities into one and set the connection value to the average of the former connection probabilities, that is $P^* = 0.15$. Structurally, this change results in an increase of $P_{1,2}^0$ by 0.05 and a decrease in $P_{1,1}^0$ and $P_{2,2}^0$ by 0.05. Our control charts from Figure 2 signal and we see that the change is appropriately detected using $\hat{P}_{1,2}$. Although we witness a decrease in $\hat{P}_{1,1}$ and $\hat{P}_{2,2}$,

TABLE 1 A description of the changes introduced to the dynamic DCSBMs in our simulation study

Sim.	Change	Description
1	$P_{1,1}^* = P_{1,1}^0 + \epsilon$	Local outbreak in community 1
2	$P_{i,j}^* = P_{i,j}^0 + \epsilon$	Global outbreak ($i = 1, 2, j = 1, 2$)
3	$\delta_1^* = \delta_1^0 + \tau$	Local variability increase in community 1
4	$\delta_i^* = \delta_i^0 + \tau$	Global variability increase ($i = 1, 2$)
5	$\boldsymbol{c}^0 \to \boldsymbol{c}^*$	Merge communities
6	$\boldsymbol{c}^0 \to \boldsymbol{c}^*$	Split community 1 into 2 communities

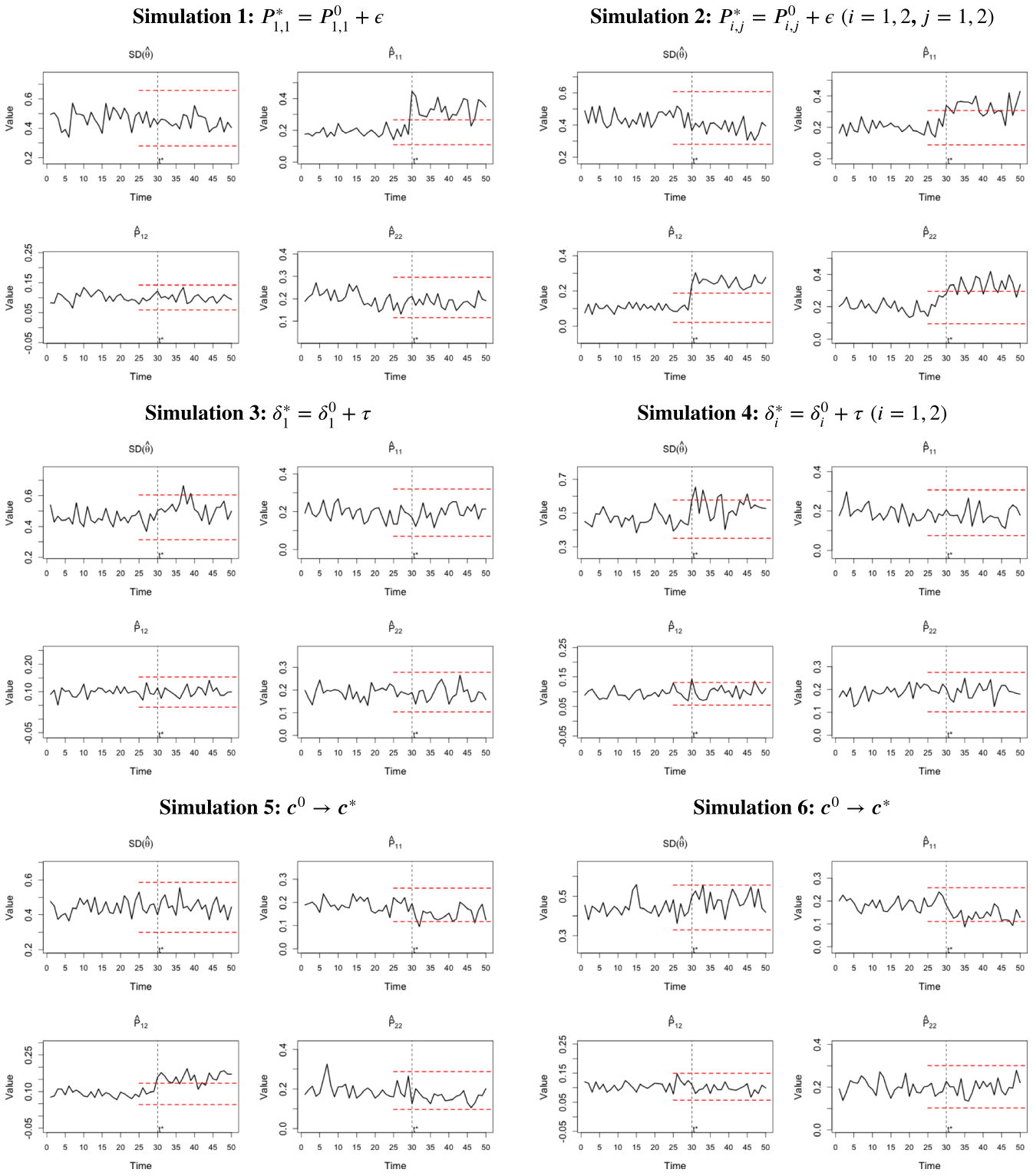

FIGURE 2 Shewhart control charts for the dynamic networks generated for simulations 1-6. Red dashed lines indicate control limits in Phase II [Colour figure can be viewed at wileyonlinelibrary.com]

the control chart does not signal a change immediately. Because this change is relatively small, we would expect that it would be better detected by EWMA control charts for $\hat{P}_{1,1}$ and $\hat{P}_{2,2}$.

In simulation 6, we once again begin with two equally sized communities. At time $t^* = 30$, we split community 1 into two communities of size 12 and 13, respectively. For the three communities after time t^*, we fixed $P^*_{i,i} = 0.20$ and

$P^*_{i,j} = 0.10$. Structurally such a change will be reflected by an overall decrease in $\hat{P}_{1,1}$. We see this behavior in the chart in the bottom of Figure 2; however, the change was not identified until time $t = 35$, where $\hat{P}_{1,1}$ went below the control limits. We expect that this type of change will be more readily detected in larger networks and in networks where the split community is large. We investigate this further in the next section.

7.2 | Average run length analysis

For each scenario described in Table 1, we evaluated our monitoring methodology by simulating the situation 1000 times. On each of these 1000 simulated runs, we calculated the number of networks until the control chart detects a change, ie, the run length, and we then estimate the average run length (ARL) from these 1000 simulations. Because μ and σ are estimated from Phase I, there will be practitioner-to-practioner sampling differences in observed ARL values, which is the basis for an ARL distribution. Thus, the average run lengths we report are estimates of the mean of this distribution, which we refer to as the average ARL (AARL) as in Saleh et al.[59] This AARL is the basis upon which different surveillance methods can be compared. In what follows, we describe the performance of our proposed surveillance technique.

In each of the scenarios discussed below, we assume the same initial form of P and $\boldsymbol{\theta}$ as discussed in the previous section, with $n = 100$ nodes in each network. We investigated the performance of the method with $m = 25$, $m = 50$ and $m = 1000$ Phase I samples. In all cases, we implemented the appropriate change 25 time periods in Phase II and thereafter generated as many networks as required to observe the first signal on each control chart. Here, we investigate the performance of control charts for $\hat{P}_{1,1}$, $\hat{P}_{1,2}$, $\hat{P}_{2,2}$, and s which is a pooled estimate of the standard deviation of $\widehat{\boldsymbol{\theta}}$ based on s_1 and s_2 since we assume $\delta_1 = \delta_2$ in Phase I. Note that this restricts what we can detect well.

We found comparable performance of our surveillance technique under Phase I sizes of $m = 25$, $m = 50$ and $m = 1000$. However, as Saleh et al[59] indicate, it is unwise to guarantee specific ARL values when the control chart parameters are estimated from small sample sizes. As such, we present the results of the $m = 1000$ case here and provide the results for the $m = 25$ and $m = 50$ cases in the Supporting Information. Note that when $m = 1000$, we gain insight into the performance of the methodology under favorable conditions (ie, when information about each statistic's distribution is ample).

Simulation 0: no change We begin by considering the performance of the methodology when no structural change has occurred. Doing so allows us to quanitify the prevalence of *false alarms*, ie, when the control chart incorrectly indicates a change has occurred. The AARLs associated with the control charts for s, $\hat{P}_{1,1}$, $\hat{P}_{1,2}$, and $\hat{P}_{2,2}$ are shown in Table 2. Although there will be variation in in-control ARLs, the large AARL values shown in the Simulation 0 row are reassuring; they indicate that false alarms are not expected to occur until hundreds of "in-control" networks have been observed. When structural changes have occurred, we expect much smaller AARLs to be associated with at least one of the four control charts. We discuss these scenarios below.

Simulations 1-2: mean interaction rate changes We quantify the method's ability to detect *local* changes in P, specifically in community 1, by adding $\epsilon = 0.01, 0.05, 0.10$ to $P^0_{1,1}$. As mentioned previously, such a change is expected to be detected on the $\hat{P}_{1,1}$ control chart. The Simulation 1 AARLs in the $\hat{P}_{1,1}$ column of Table 2 indicate that this is indeed the case; on average we expect the $\hat{P}_{1,1}$ control chart to detect such a change in roughly 10 networks for moderate sized changes in $P_{1,1}$, and roughly two networks for large changes. On the other hand, the large AARL values for the other three statistics indicate that none of them is likely to detect this change, as desired.
We similarly quantify the method's ability to detect *global* changes in P by adding $\epsilon = 0.01, 0.05, 0.10$ to each $P^0_{i,j}$. In this situation, we expect all entries of $\hat{P}$ to signal a change. The Simulation 2 AARLs in the $\hat{P}_{1,1}$, $\hat{P}_{1,2}$, and $\hat{P}_{2,2}$ columns of Table 2 support this hypothesis. As expected, we see that the $\hat{P}_{1,2}$ control chart signals this change fastest since ϵ is much larger relative to $P_{1,2}$ than it is to $P_{1,1}$ and $P_{2,2}$.

Simulations 3-4: variance of interaction rate changes We introduced *local* changes in interaction variability among the nodes in community 1 by adding $\tau = 0.05, 0.10, 0.25$ to δ^0_1, and we introduce *global* changes in interaction variability among all nodes in the network by adding $\tau = 0.05, 0.10, 0.25$ to both δ^0_1 and δ^0_2. In both cases, we expect the s control chart to signal this change. The AARLs in the Simulation 3 and Simulation 4 rows of Table 2 support this claim. In particular, we can expect this control chart to detect global changes more quickly than local changes, and in both cases large changes will be detected more quickly than small changes.

Simulations 5-6: change in community structure Simulation 5 corresponds to the merging of communities. Since $P_{1,2}$ is most affected by this change, we expect the $\hat{P}_{1,2}$ control chart to signal quickest. The AARLs in the Simulation 5 row of Table 2 support this intuition; while $\hat{P}_{1,1}$ and $\hat{P}_{2,2}$ tend to detect this change more quickly than s, the $\hat{P}_{1,2}$

Sim.		Change	s	$\hat{P}_{1,1}$	$\hat{P}_{1,2}$	$\hat{P}_{2,2}$
0		None	317.18	317.98	310.33	255.29
		$\epsilon = 0.01$	294.80	134.00	413.70	332.4
1	$P^*_{1,1} = P^0_{1,1} + \epsilon$	$\epsilon = 0.05$	284.90	9.87	257.27	207.70
		$\epsilon = 0.10$	524.40	2.23	289.90	325.90
		$\epsilon = 0.01$	498.80	140.90	64.65	142.30
2	$P^*_{i,j} = P^0_{i,j} + \epsilon$	$\epsilon = 0.05$	211.10	9.48	1.71	12.17
		$\epsilon = 0.10$	93.30	2.01	1.01	2.28
		$\tau = 0.05$	106.51	221.40	260.10	202.70
3	$\delta^*_1 = \delta^0_1 + \tau$	$\tau = 0.10$	115.70	152.33	305.29	544.60
		$\tau = 0.25$	18.81	63.35	107.20	431.00
		$\tau = 0.05$	93.58	232.30	246.10	216.10
4	$\delta^*_i = \delta^0_i + \tau$	$\tau = 0.10$	36.33	142.00	185.94	218.50
		$\tau = 0.25$	4.94	52.88	92.23	53.87
		$n = 50$	327.60	74.97	1.64	40.81
5	Merge comm.	$n = 100$	247.00	39.79	1.66	27.61
		$n = 500$	72.70	37.56	1.61	37.32
		$n = 50$	152.10	32.88	168.30	427.80
6	Split comm.	$n = 100$	127.50	33.90	313.39	426.20
		$n = 500$	72.70	33.37	315.50	446.50

TABLE 2 Average ARLs for simulations in Section 7.2 when $m = 1000$

chart detects the change almost immediately. Interestingly, this result does not appear to depend on the size of the network.

When community j is split into two (equally sized) communities, the illustrative example in Section 7.1 suggests that a control chart for $\hat{P}_{j,j}$ should signal most quickly. The results in the Simulation 6 row of Table 2 substantiate this; when community 1 is split into two communities, the control for $\hat{P}_{1,1}$ detects this more quickly than the other control charts, but perhaps not as quickly as a practitioner would like. This suggests that the proposed surveillance methodology may not be well suited for detecting community splitting, even though it is highly effective at detecting each of the other types of structural change considered.

8 | DISCUSSION

In this paper, we have illustrated the utility of the degree corrected stochastic block model (DCSBM) in detecting local and global structural changes in networks. Our proposed model is flexible, and can capture both degree heterogeneity and community structure in networks, two important features that are common in social and biological networks. We proposed a fast and effective monitoring methodology based on the surveillance of maximum likelihood estimates from the DCSBM using Shewhart and EWMA control charts for individuals. When applied to the US Senate covoting network, our methodology was able to identify relevant and significant changes in the bipartisan nature of the US Congress. Our analysis reveals that the dynamic DCSBM can effectively model a variety of dynamic networks with structural changes, and that our proposed surveillance strategy can detect relevant changes in a real dynamic system.

The majority of contemporary surveillance methodologies are based on the assumption that the observed dynamic graph is unweighted. As a consequence, model-based approaches generally model the existence of an edge as a Bernoulli random variable and often rely on some thresholding technique on count data. The DCSBM flexibly models the edge weight associated with each edge using a Poisson random variable. In future work, we plan to use the DCSBM to quantify the loss of information when count data is thresholded to binary outcomes. Furthermore, it would be interesting to generalize the work done here to random graph models with continuous-valued edge weights such as the weighted stochastic block model from Aicher et al[60] or the generalized exponential random graph model developed in previous studies.[61-63]

Our proposed monitoring strategy establishes one practically useful technique among a general family of methods for surveillance. Our framework relies on two components: a parametric dynamic random graph model for modeling the features of the graph, and a control chart from statistical process monitoring for the detection of changes in the parameters. We considered a dynamic DCSBM random graph model and the Shewhart and EWMA control charts for surveillance. This serves only as a first step in understanding the utility of our proposed surveillance strategy. In future work, it would be useful to explore the use of other parametric random graph models and control charts and to assess the advantages and disadvantages of each strategy. In particular, future work will explore the utility of dynamic latent space models like that

discussed in Sewell and Chen[21] as well as dynamic exponential random graph models like the TERGM family described in Hanneke et al.[19] Finally, as our proposed method is a model-based strategy, model misspecification is a real practical concern. Model selection for network data remains a relatively open area, although some initial work has been done.[50] Studying the effect of misspecification on monitoring is an important area of research that requires a considerable amount of work. We plan to pursue this area in future work.

ORCID

James D. Wilson https://orcid.org/0000-0002-2354-935X
Nathaniel T. Stevens https://orcid.org/0000-0001-6149-5797

REFERENCES

1. Vargas JA. Spring awakening: how an Egyptian revolution began on Facebook. The New York Times, Sunday Book Review; 2012.
2. Shetty J, Adibi J. Discovering important nodes through graph entropy the case of Enron email database. In: Proceedings of the 3rd international workshop on link discovery ACM; 2005; New York, NY, USA:74-81.
3. Akoglu L, Faloutsos C. Anomaly, event, and fraud detection in large network datasets. In: WSDM '13 Proceedings of the sixth ACM international conference on Web search and data mining ACM; 2013; New York, NY, USA:773-774.
4. Chau DH, Pandit S, Faloutsos C. *Detecting Fraudulent Personalities in Networks of Online Auctioneers*. Berlin, Heidelberg: Springer; 2006;103-114.
5. Pandit S, Chau DH, Wang S, Faloutsos C. Netprobe: a fast and scalable system for fraud detection in online auction networks. In: Proceedings of the 16th international conference on world wide web ACM; 2007; Raleigh NC:201-210.
6. Krebs VE. Mapping networks of terrorist cells. *Connections*. 2002;24(3):43-52.
7. Porter MD, White G. Self-exciting hurdle models for terrorist activity. *Ann Appl Stat*. 2012;6(1):106-124.
8. Reid E, Qin J, Zhou Y, Lai G, Sageman M, Weimann G, Chen H. Collecting and analyzing the presence of terrorists on the web: a case study of jihad websites. *ISI'05 Proceedings of the 2005 IEEE International Conference on Intelligence and Security Informatics*. Verlag Berlin, Heidelberg: Springer; 2005:402-411.
9. Fire M, Katz G, Elovici Y. Strangers intrusion detection—detecting spammers and fake profiles in social networks based on topology anomalies. *Human J*. 2012;1(1):26-39.
10. Savage D, Zhang X, Yu X, Chou P, Wang Q. Anomaly detection in online social networks. *Soc Networks*. 2014;39:62-70.
11. Bindu P, Thilagam PS. Mining social networks for anomalies: methods and challenges. *J Netw Comput Appl*. 2016;68:213-229.
12. Jeske DR, Stevens NT, Tartakovsky AG, Wilson JD. Statistical methods for network surveillance. *Appl Stoch Model Bus Ind*. 2018;34:457-459.
13. Woodall WH, Zhao MJ, Paynabar K, Sparks R, Wilson JD. An overview and perspective on social network monitoring. *IISE Transactions*. 2017;49(3):354-365.
14. Woodall WH, Montgomery DC. Research issues and ideas in statistical process control. *J Qual Technol*. 1999;31(4):376-386.
15. Frisén M. Optimal sequential surveillance for finance, public health, and other areas (with discussion). *Seq Anal*. 2009;28:310-337.
16. Woodall WH, Montgomery DC. Some current directions in the theory and application of statistical process monitoring. *J Qual Technol*. 2014;46(1):78-94.
17. Karrer B, Newman ME. Stochastic blockmodels and community structure in networks. *Phys Rev E*. 2011;83(1):16107.
18. Montgomery DC. *Introduction to Statistical Quality Control*. 7th ed.: John Wiley and Sons, Inc; 2013.
19. Hanneke S, Fu W, Xing EP. Discrete temporal models of social networks. *Electron J Statistics*. 2010;4:585-605.
20. Krivitsky PN, Handcock MS. A separable model for dynamic networks. *J R Stat Soc Ser B Stat Methodol*. 2014;76(1):29-46.
21. Sewell DK, Chen Y. Latent space models for dynamic networks. *J Am Stat Assoc*. 2015;110(512):1646-1657.
22. Szekely E, Pappa I, Wilson JD, Bhamidi S, Jaddoe VW, Verhulst FC, Tiemeier H, Shaw P. Childhood peer network characteristics: genetic influences and links with early mental health trajectories. *J Child Psychol Psychiatry*. 2016;57(6):687-694.
23. Marchette D. Scan statistics on graphs. *Wiley Interdiscip Rev Comput Stat*. 2012;4(5):466-473.
24. Neil J, Hash C, Brugh A, Fisk M, Storlie CB. Scan statistics for the online detection of locally anomalous subgraphs. *Technometrics*. 2013;55(4):403-414.
25. Park Y, Priebe CE, Youssef A. Anomaly detection in time series of graphs using fusion of graph invariants. *IEEE J Sel Top Sign Proces*. 2013;7(1):67-75.
26. Priebe CE, Conroy JM, Marchette DJ, Park Y. Scan statistics on Enron graphs. *Comput Math Organ Theory*. 2005;11(3):229-247.
27. Jones-Farmer LA, Woodall WH, Steiner SH, Champ CW. An overview of phase I analysis for process improvement and monitoring. *J Qual Technol*. 2014;46(3):265-280.
28. Azarnoush B, Paynabar K, Bekki J, Runger G. Monitoring temporal homogeneity in attributed network streams. *J Qual Technol*. 2016;48(1):28-43.
29. Zhao MJ, Driscoll AR, Sengupta S, Fricker RDJr, Spitzner DJ, Woodall WH. Performance evaluation of social network anomaly detection using a moving window-based scan method. *Qual Reliab Eng Int*. 2016;34:1699-1716.

30. Saleh NA, Mahmoud MA, Jones-Farmer LA, Zwetsloot I, Woodall WH. Another look at the EWMA control chart with estimated parameters. *J Qual Technol*. 2015;47(4):363-382.
31. Sparks R, Wilson JD. Monitoring communication outbreaks among an unknown team of actors in dynamic networks. arXiv preprint arXiv:1606.09308; 2016.
32. Peel L, Clauset A. Detecting change points in the large-scale structure of evolving networks. arXiv preprint arXiv:1403.0989; 2014.
33. Heard NA, Weston DJ, Platanioti K, Hand DJ. Bayesian anomaly detection methods for social networks. *Ann Appl Stat*. 2010;4(2):645-662.
34. Barnett I, Onnela J-P. Change point detection in correlation networks. *Sci Rep*. 2016;6:18893.
35. Zambon D, Alippi C, Livi L. Change point methods on a sequence of graphs. arXiv preprint arXiv:1805.07113; 2018.
36. Chen H. Sequential change-point detection based on nearest neighbors. arXiv preprint arXiv:1604.03611; 2016.
37. Roy S, Atchadé Y, Michailidis G. Change point estimation in high dimensional Markov random-field models. *J R Stat Soc Ser B Stat Methodol*. 2017;79(4):1187-1206.
38. Keshavarz H, Michailidis G, Atchade Y. Sequential change-point detection in high-dimensional Gaussian graphical models. arXiv preprint arXiv:1806.07870; 2018.
39. Yu L, Woodall WH, Tsui K-L. Detecting node propensity changes in the dynamic degree corrected stochastic block model. *Soc Networks*. 2018;54:209-227.
40. Hosseini SS, Noorossana R. Performance evaluation of EWMA and CUSUM control charts to detect anomalies in social networks using average and standard deviation of degree measures. *Qual Reliab Eng Int*. 2018;34(4):477-500.
41. Perry MB. An EWMA control chart for categorical processes with applications to social network monitoring. *J Qual Technol*. 2019:1-16.
42. Noorossana R, Hosseini SS, Heydarzade A. An overview of dynamic anomaly detection in social networks via control charts. *Qual Reliab Eng Int*. 2018;34(4):641-648.
43. Parker KS, Wilson JD, Marschall J, Mucha PJ, Henderson JP. Network analysis reveals sex-and antibiotic resistance-associated antivirulence targets in clinical uropathogens. *ACS Infectious Diseases*. 2015;1(11):523-532.
44. Greene D, Doyle D, Cunningham P. Tracking the evolution of communities in dynamic social networks. In: ASONAM '10 Proceedings of the 2010 International Conference on Advances in Social Networks Analysis and Mining IEEE; 2010:176-183.
45. Porter MA, Onnela J-P, Mucha PJ. Communities in networks. *Notices of the AMS*. 2009;56(9):1082-1097.
46. Fortunato S. Community detection in graphs. *Phys Rep*. 2010;486(3):75-174.
47. Barabási A-L, Albert R. Emergence of scaling in random networks. *Science*. 1999;286(5439):509-512.
48. Clauset A, Shalizi CR, Newman ME. Power-law distributions in empirical data. *SIAM Review*. 2009;51(4):661-703.
49. Kasthurirathna D, Piraveenan M. Emergence of scale-free characteristics in socio-ecological systems with bounded rationality. *Sci Rep*. 2015;5:10448.
50. Yan X, Shalizi C, Jensen JE, et al. Model selection for degree-corrected block models. *J Stat Mech: Theory Exp*. 2014;2014(5):P05007.
51. Moody J, Mucha PJ. Portrait of political party polarization. *Network Science*. 2013;1(01):119-121.
52. Chen Z, Hendrix W, Samatova NF. Community-based anomaly detection in evolutionary networks. *J Intell Inf Syst*. 2012;39(1):59-85.
53. Von Luxburg U. A tutorial on spectral clustering. *Stat Comput*. 2007;17(4):395-416.
54. Han Q, Xu K, Airoldi E. Consistent estimation of dynamic and multi-layer block models. In: ICML'15 Proceedings of the 32nd International Conference on International Conference on Machine Learning; 2015:1511-1520.
55. Qin T, Rohe K. Regularized spectral clustering under the degree-corrected stochastic blockmodel. *Adv Neural Inf Proces Syst*. 2013:3120-3128.
56. Crowder SV. Design of exponentially weighted moving average schemes. *J Qual Technol*. 1989;21(3):155-162.
57. Rae NC. *The Decline and Fall of the Liberal Republicans: From 1952 to the Present*. USA: Oxford University Press; 1989.
58. Smith RN. *On His Own Terms: A Life of Nelson Rockefeller*. New york: Random House; 2014.
59. Saleh NA, Mahmoud MA, Keefe MJ, Woodall WH. The difficulty in designing Shewhart xbar and x control charts with estimated parameters. *J Qual Technol*. 2015;47(2):127-138.
60. Aicher C, Jacobs AZ, Clauset A. Adapting the stochastic block model to edge-weighted networks. arXiv preprint arXiv:1305.5782; 2013.
61. Desmarais BA, Cranmer SJ. Statistical inference for valued-edge networks: the generalized exponential random graph model. *PloS one*. 2012;7(1):e30136.
62. Wilson JD, Denny MJ, Bhamidi S, Cranmer SJ, Desmarais BA. Stochastic weighted graphs: flexible model specification and simulation. *Soc Networks*. 2017;49:37-47.
63. Stillman PE, Wilson JD, Denny MJ, et al. Statistical modeling of the default mode brain network reveals a segregated highway structure. *Sci Rep*. 2017;7(1):11694.

AUTHOR BIOGRAPHIES

James D. Wilson is an Assistant Professor of Statistics and Data Science at the University of San Francisco. He is also Director of Data Science and Codirector of Research at the Data Institute at the university. He received his PhD in Statistics and Operations Research from the University of North Carolina at Chapel Hill in 2015. His research focuses on the development and analysis of computational techniques for the study of networks.

Nathaniel T. Stevens is an Assistant Professor of Statistics at the University of San Francisco. He received his PhD in Statistics from the University of Waterloo in 2014. His research interests include experimental design, the assessment and comparison of measurement systems, reliability analysis, and variation reduction.

William H. Woodall is a Professor of Statistics at Virginia Tech. He received his PhD in Statistics from Virginia Tech in 1980. His research interests include statistical quality control and improvement, all aspects of control charting, prospective public health surveillance, and critiques of fuzzy logic. He is a Fellow of the American Statistical Association and the American Society for Quality, and an elected member of the International Statistical Institute.

SUPPORTING INFORMATION

Additional supporting information may be found online in the Supporting Information section at the end of the article.

How to cite this article: Wilson JD, Stevens NT, Woodall WH. Modeling and detecting change in temporal networks via the degree corrected stochastic block model. *Qual Reliab Engng Int*. 2019;35:1363–1378. https://doi.org/10.1002/qre.2520