Statistica Sinica **30** (2020), 251-284
doi:https://doi.org/10.5705/ss.202017.0218

TIME-VARYING ESTIMATION AND DYNAMIC MODEL SELECTION WITH AN APPLICATION OF NETWORK DATA

Lan Xue, Xinxin Shu and Annie Qu

Oregon State University, Merck and
University of Illinois at Urbana-Champaign

Abstract: In many biomedical and social science studies, it is important to identify and predict the dynamic changes of associations among network data over time. We propose a varying-coefficient model to incorporate time-varying network data, and impose a piecewise penalty function to capture local features of the network associations. The proposed approach is semi-parametric, and therefore flexible in modeling dynamic changes of association in network data problems. Furthermore, the approach can identify the time regions when dynamic changes of associations occur. To achieve a sparse network estimation at local time intervals, we implement a group penalization strategy involving parameters that overlap between groups. However, this makes the optimization process challenging for large-dimensional network data observed at many time points. We develop a fast algorithm, based on the smoothing proximal-gradient method, that is computationally efficient and accurate. We illustrate the proposed method through simulation studies and children's attention deficit hyperactivity disorder fMRI data, showing that the proposed method and algorithm recover dynamic network changes over time efficiently.

Key words and phrases: B-spline, dynamic network, model selection consistency, proximal gradient method, varying-coefficient model.

1. Introduction

In social science, genomic, environmental, and biomedical studies, it is scientifically important to identify and predict associations and interactions between genes, spatial locations, or social structures effectively. Network modeling (e.g., Kolaczyk (2009)) can effectively quantify the associations between variables. Our method is motivated by a study on children's attention deficit hyperactivity disorder. The data are available from the ADHD-200 sample initiative website: `http://fcon_1000.projects.nitrc.org/indi/adhd200/`. The test samples contain fMRI data from regions of interest (ROIs) in the brains of children with ADHD. These data are measured repeatedly at many time-points. We are

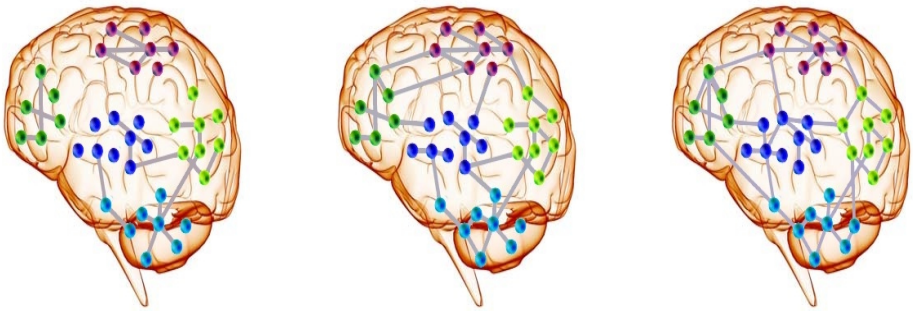


Figure 1. Changes of associations between different sites of a brain over three time-points.

interested in identifying associations and interactions between ROIs of the brain over time in order to better understand how ADHD patients' brains function.

Figure 1 illustrates the dynamic changes of associations between several ROIs of a brain over three time-points. We are interested in extracting the underlying signals of associations by modeling the responses of brain activities over time. This can be formulated as a time-varying network problem, where the ROIs are variables or nodes in the network, and the associations between ROIs represent edges connecting nodes of the network.

Recent development in the field of network modeling includes the high-dimensional graphical models of Meinshausen and Bühlmann (2006); Friedman, Hastie and Tibshirani (2007), and Peng et al. (2009). The central idea of these approaches is to estimate the precision matrix or the inverse of the covariance matrix, which provides a conditional correlation interpretation for the variables in the graph, where a zero partial correlation implies pairwise conditional independence. In addition, Shen, Huang and Pan (2012) and Zhu, Shen and Pan (2013) both develop methods for simultaneous grouping pursuit and feature selection in high-dimensional graphs. For multiple graphs, Guo et al. (2011) jointly estimate graphical models to capture the dependence between multiple graphs and their common structure. In addition, Zhu, Shen and Pan (2014) propose a maximum penalized likelihood approach to model the structural changes over multiple graphs, thus incorporating the dependencies between interacting units.

Most of the existing literature focuses on the network data problem observed at one time-point only. However, networks can be observed at multiple time-points; where dynamic changes of associations are of scientific interest and require quantification. For example, in gene-expression data, functional magnetic resonance imaging (fMRI), and social network data, associations often change

over time. Therefore it is important to model and estimate the dynamic changes to the network structure.

Modeling time-varying network data can be statistically and computationally challenging. This is because the network structures can become complex over time, involve large-dimensional parameter estimations, and be computationally intensive with high-dimensional matrix operations. Existing approaches for time-course network data include using linear mixed-effect modeling to incorporate temporal correlations (Shojaie and Michailidis (2010)), the kernel-reweighted logistic regression method for network structure that change over time (Song, Kolar and Xing (2009); Kolar, Parikh and Xing (2010)), and time-varying Markov random fields (Kolar and Xing (2009)). However, these approaches are mainly used to estimate time-varying networks, and are not designed to select models that capture changes of associations in local time regions.

We propose a dynamic network model that captures the changes of associations using a varying-coefficient model (Hastie and Tibshirani (1993); Huang, Wu and Zhou (2002); Cheng, Honda and Zhang (2016)). The model of the dynamics of the partial correlations is semiparametric and, therefore, flexible in modeling the nonlinear changes of the coefficients. In addition, we propose a one-step penalized polynomial spline method to detect zero regions in the varying coefficients. Therefore, we are able to locate the time regions when dynamic changes of associations occur. This method can be used to identify the changes of associations between ROIs over time, as in the example of fMRI data for ADHD patients, which could be useful for detecting dynamic changes in brain functions.

The one-step penalized polynomial spline method proposed here is quite different from the penalization methods (Xue (2009); Wei, Huang and Li (2011); Xue and Qu (2012)) developed recently for variable selection in semiparametric models. The latter approaches were developed to determine whether a nonparametric function is zero in the entire region. Therefore, an L_2-norm of the spline coefficients is penalized to shrink the function to zero on the entire region. In contrast, our one-step penalized polynomial spline method aims to detect local zero regions in the varying coefficients, thus locating the time regions when dynamic changes of associations occur. We use the local property of polynomial splines that the spline functions on a given local interval depend only on the neighboring B-spline bases. Therefore, we propose penalizing only those coefficients relevant to a given local interval in a groupwise fashion. This new form of penalization raises challenges in terms of both computation and theory development, as discussed in Sections 3 and 4.

In order to achieve sparse network data at local time intervals, we propose a piecewise penalized loss function that incorporates the local features of the varying-coefficient models in the dynamic modeling. The piecewise penalization strategy involves spline-coefficient parameters that overlap between different penalty groups. However, the popular coordinate-wise descent algorithm cannot be applied in our optimization. Thus we propose an alternative algorithm that is computationally efficient and accurate, based on the proximal-gradient method. This approach does not involve large-dimensional matrix inversion and is capable of handling large-dimensional network data.

One computational challenge we face when using time-varying network data is that the volume of data is extremely large, because they include observations for many nodes over many time points. For example, when the network comprises about 100 nodes and is observed over 50 time points, the dimension of the matrix operation could reach 10^5 in the iteration process. Existing methods for handling time-varying networks mainly target relatively small network sizes and limited time points. Therefore, there is great demand for computationally efficient and fast algorithms to solve large-dimensional time-varying network problems. The proposed group penalization strategy effectively ensures sparsity at local time intervals. However, it incurs an additional computational cost in the optimization process, because it requires a high degree of memory storage and the use of matrix operations when solving the dynamic network problem. In theory, it is also more challenging to establish local-feature than global-feature model selection consistency. We show that the proposed method identifies zero estimators in the nonsignal time regions, and estimates the partial correlation functions uniformly and consistently in the signal regions.

Recent works on the dynamic modeling of network changes include the reversible jump MCMC (Lebre et al. (2010)), time-series model for covariance matrix (Zhou, Lafferty and Wasserman (2010)), piecewise constant varying-coefficient varying-structure (VCVS) models (Kolar, Song and Xing (2009); Kolar and Xing (2011, 2012)), and nonparametric model for the dynamic covariance matrix (Chen and Leng (2016)). Our approach differs from these approaches because we use a penalized polynomial spline function to model the network changes, allowing us to accommodate many time points at a scalable computing cost. In contrast, the reversible jump MCMC approach is mainly applicable for a limited number of time points, and the piecewise constant VCVS approach is used to model abrupt change rather than smooth changes to the network structure. The method of Zhou, Lafferty and Wasserman (2010) is based on the penalized

maximum likelihood approach, where the covariance matrix is estimated using a kernel smoother. However, they do not establish the sparsistency property, by which all zero parameters are estimated as zero with probability approaching one. In contrast, we establish the sparsistency property for the proposed method, which is important for detecting dynamic changes on the network structure. The approach of Chen and Leng (2016) is nonparametric, in that they make no assumption on the covariance matrix. In contrast, our method is semiparametric in that we model each partial correlation function as a semiparametric varying-coefficient function.

In addition, dynamic brain network models are receiving much attention. The study of neural connectivity disruptions caused by disease pathology requires models that capture the temporal connectivity of brain networks. Current dynamic brain network models include the dynamic causal models (DCMs) (Friston, Harrison and Penny (2003)) and a nonlinear extension of a DCM (Stephan et al. (2008)) that builds on a causal neuronal model. The latter takes dynamic specified input, state, and output variables, corresponding to the stimulus functions, neuronal activities or biophysical variables, and outcomes measured from the brain ROIS, respectively. In addition, Wang, Lin and Wu (2015) investigate the important role of the dynamic temporal–topological structure of the ADHD brain network using sliding time-window correlation coefficients. Wee et al. (2016) propose a fused sparse learning algorithm to jointly estimate temporal networks, while encouraging temporally correlated networks to form similar network structures using the fused LASSO (Tibshirani et al. (2005)). Furthermore, Lee et al. (2011) recover the sparse brain network derived from partial correlations when the sample size is relatively small, but the dimension of the parameters is high. Wee et al. (2012) also consider a constrained sparse linear regression model using the LASSO penalty when there is a relatively small number of connections within a brain network. However, the sparse network models do not incorporate dynamic changes to the brain network.

Furthermore, the diffusion wavelet has been proposed to analyze time-varying brain networks. It provides a framework within which to study the properties and structures of a graph in the spectral domain, and provides multi-resolution and interpretable basis representations of network data. Chung (1997) gives a comprehensive overview of spectral graph theory. Leonardi and Van De Ville (2011) applied a spectral graph wavelet transform (SGWT) to brain functional-connectivity data. They decomposed fMRI data using the SGWT, and then used wavelet coefficients to understand the connectivity of the network. However, this

connectivity can only be interpreted in a specific frequency band. Kim et al. (2013) applied a diffusion wavelet to conduct a multi-resolution analysis on brain networks, comparing the connectivity differences between healthy and bipolar patients. The aforementioned works represent information contained in a graph using a few interpretable wavelet bases, that capture structural differences in brain networks. In general, diffusion wavelets are to reduce the dimensionality, while appropriately incorporating the network topology information. In contrast, our work aims to model the pairwise connectivity of the network. In future research, we will first use our method to estimate the network connectivity, after which we will conduct a multi-resolution analysis using the diffusion wavelet to understand the differences between such networks.

The reminder of the paper is organized as follows. Section 2 proposes the penalized polynomial spline method for time-varying network data. Section 3 provides the smoothing proximal-gradient (SPG) algorithm that captures dynamic changes in the network data over time. Section 4 presents the asymptotic theory of model selection local consistency. In Section 5, we compare the numerical performance of the proposed SPG algorithm with that of existing approaches. Section 6 illustrates the proposed method using the fMRI data on ADHD patients. The final section concludes the paper.

2. Time-varying Networks

In this study, we focus on time-varying network data and model the dynamic changes in its partial correlations or structural changes of the network over time. Both the correlation function and the partial correlation function can be used to characterize associations between variables of interest. We focus on the partial correlation function, mainly because we are interested in the **conditional** dependence/independence between variables in a network. This correlation measures the direct relationship between two variables, while removing the influence of other variables.

Let $\mathbf{y}(t) = \big(y_1(t), \ldots, y_p(t)\big)'$ be a set of time-varying variables observed at time t, and $\{\mathbf{y}(t), t \in \mathbf{I}\}$ be the corresponding continuous stochastic process defined on a compact interval $\mathbf{I}$. Without loss of generality, let $\mathbf{I} = [0, 1]$. Suppose the data consist of n subjects with measurements taken at m discrete time-points $0 \leq t_{k1} < \cdots < t_{km} \leq 1$, for each subject $k = 1, \ldots, n$. For each subject, the observation $\mathbf{y}^k(\mathbf{t}_k) = \big(\mathbf{y}^k(t_{k1}), \ldots, \mathbf{y}^k(t_{km})\big)'$ is a discrete realization of the continuous stochastic process $\{\mathbf{y}(t), t \in \mathbf{I}\}$ at m subject-specific time-points

$\mathbf{t}_k = (t_{k1}, \ldots, t_{km})$. Here, $\mathbf{y}^k(t_{ku}) = \left(y_1^k(t_{ku}), \ldots, y_p^k(t_{ku})\right)'$, for $u = 1, \ldots, m$, are p variables observed at time t_{ku} for the kth subject.

Let $\boldsymbol{\rho}(t) = \{\rho^{12}(t), \ldots, \rho^{(p-1)p}(t)\}'$ be the partial correlation function of $\mathbf{y}(t)$. Suppose each partial coefficient function $\rho^{ij}(t)$ varies in time smoothly. Then we can apply the polynomial spline to approximate the time-varying coefficients, because this provides a good approximation of any smooth function, even with a small number of knots. Let $\{\nu_h\}_{h=1}^{N_n}$ be N_n interior knots within the interval [0, 1], and let Υ be a partition of the interval [0, 1] with N_n knots. That is, $\Upsilon_n = \{0 = \nu_0 < \nu_1 < \cdots < \nu_{N_n} < \nu_{N_n+1} = 1\}$. The polynomial splines of order $q+1$ are functions with q-degree polynomials on intervals $[\nu_{h-1}, \nu_h), h = 1, \ldots, N_n$, and $[\nu_{N_n}, \nu_{N_n+1}]$, and $q-1$ continuous derivatives globally. We denote the space of such spline functions by G_n. Let $\{B_h(\cdot)\}_{h=1}^{J_n}$ be a set of B-spline bases of G_n, where $J_n = N_n + q + 1$ and the function $\rho^{ij}(t)$ for any $1 \le i < j \le p$ can be approximated by

$$\rho^{ij}(t) \approx g^{ij}(t) = \sum_{h=1}^{J_n} \beta_h^{ij} B_h(t) = (\beta^{ij})' \mathbf{B}(t),$$

where $\beta^{ij} = (\beta_1^{ij}, \ldots, \beta_{J_n}^{ij})'$ is a set of coefficients, and $\mathbf{B}(t) = (B_1(t), \ldots, B_{J_n}(t))'$ are B-spline bases. In practice, different B-spline bases can be used to approximate $\rho^{ij}(t)$. For simplicity, the same set of B-spline bases is used for the various partial correlation functions presented here.

In addition to polynomial splines, other basis functions can be used to approximate unknown functions, including wavelet and trigonometric polynomials. Sections 2.5 and 2.6 of Fan and Gijbels (1996) provide a review of the basis choices. We chose the polynomial spline owing to its sound numerical properties and excellent approximation power. Given a sufficient number of knots, any continuous function can be approximated arbitrarily well by polynomial splines, assuming it is reasonably smooth. However, in general, polynomial splines do not approximate functions with discontinuities and rapid variations sufficiently well. In such cases, other basis functions, such as the wavelet, might be more suitable.

Suppose $\mathbf{y}(t)$ has mean $\mathbf{0}$ and covariance $\boldsymbol{\Sigma}(t)$. Denote the concentration matrix $\boldsymbol{\Sigma}^{-1}(t)$ by $\left(\sigma^{ij}(t)\right)_{p\times p}$. Then we can express $y_i(t)$ by a varying-coefficient model, as

$$y_i(t) = \sum_{j \ne i} \beta_{ij}(t) y_j(t) + \varepsilon_i(t), \tag{2.1}$$

where $\beta_{ij}(t) = \rho^{ij}(t)\sqrt{\sigma^{jj}(t)/\sigma^{ii}(t)}$, and $Var(\varepsilon_i(t)) = 1/\sigma^{ii}(t)$. The errors $\varepsilon_i(t)$

can be correlated over time. However, in the following, longitudinal correlation is not incorporated. Instead we assume that $\varepsilon_i(t)$ is independent over time. We develop a method for identifying the local sparsity of the coefficient functions $\{\beta_{ij}(t)\}$ over time. In a traditional polynomial spline estimation, we would replace $\rho^{ij}(t)$ with $g^{ij}(t)$, and then estimate the spline coefficients $\beta = \{\beta^{ij}, 1 \le i < j \le p\}$ by minimizing the weighted sum of squares in (2.2). The benefit of using a spline approximation for the time-varying coefficient model is that it is computationally fast and efficient.

In this study we are interested in locally sparse estimators of the partial correlations that characterize the dynamic changes of network associations over time. The B-spline basis function has a desirable local property. Denote any interval constructed by two consecutive knots as (ν_{h-1}, ν_h), for $1 \le h \le N_n + 1$. If $t \in (\nu_{h-1}, \nu_h)$, the spline function $g^{ij}(t)$ is only affected by the basis functions $B_h, \ldots, B_{h+q}$. Therefore, the spline function $g^{ij}(t)$ is locally zero within the interval (ν_{h-1}, ν_h), if and only if the spline coefficients $\boldsymbol{\gamma}_h^{ij} = (\beta_h^{ij}, \ldots, \beta_{(h+q)}^{ij})'$ are all zero. In addition, the whole region [0, 1] can be divided into $N_n + 1$ intervals by the spline knots. Therefore, we penalize the group of spline coefficients associated with each local interval $[\nu_{h-1}, \nu_h]$ in a groupwise fashion. Consequently, this provides locally sparse spline estimators $\widetilde{\rho}_{ij}(t)$, which can be completely zero on certain time intervals spanned by the knot sequence.

We propose the following piecewise penalized loss function to achieve sparse network data:

$$\begin{aligned} &PL(\boldsymbol{\beta}, \boldsymbol{\sigma}, \mathbf{t}, \mathbf{y}) \\ &= \frac{1}{2nm} \sum_{k=1}^{n} \sum_{i=1}^{p} \sum_{u=1}^{m} w_{i_{ku}} \left(y_i^k(t_{ku}) - \sum_{j \ne i}^{p} \sum_{h=1}^{J_n} \beta_h^{ij} B_h(t_{ku}) \sqrt{\frac{\sigma^{jj}(t_{ku})}{\sigma^{ii}(t_{ku})}} y_j^k(t_{ku}) \right)^2 \\ &\quad + \sum_{i<j}^{p} \sum_{h=1}^{N_n+1} P_{\lambda_n}(\|\boldsymbol{\gamma}_h^{ij}\|), \end{aligned} \tag{2.2}$$

where $\mathbf{y} = \{\mathbf{y}^k(\mathbf{t}_k)\}_{k=1}^n$, $\boldsymbol{\beta} = (\beta_1^{1,2}, \ldots, \beta_{J_n}^{1,2}, \ldots, \beta_1^{p-1,p}, \ldots, \beta_{J_n}^{p-1,p})'$ is a $p(p-1)J_n/2$-dimensional spline coefficient, $\boldsymbol{\sigma} = \{\sigma^{ii}(\mathbf{t})\}_{i=1}^p$ with $\mathbf{t} = (\mathbf{t}_1, \ldots, \mathbf{t}_n)'$, and $w_{i_{ku}}$ are nonnegative weights, typically chosen as $\sigma^{ii}(t_{ku})$. In addition, $\|\cdot\|$ is the vector L_2-norm. Note that in contrast to the loss function of Peng et al. (2009), both the weights and the components in the concentration matrix vary over time.

The first term of (2.2) is the weighted sum of squares, and the second term

P_{λ_n} is the penalty function, which can be chosen from the LASSO, SCAD, or adaptive LASSO, described in subsection 3.1. The performance of the penalty function depends on the tuning parameter λ_n, the selection of which is discussed in subsection 3.2. Intuitively, if $\|\gamma_h^{ij}\|$ shrinks to zero, then all elements of γ_h^{ij} are zero and the spline function $g^{ij}(t)$ is locally zero on the corresponding interval. The penalty term in (2.2) differs from the typical penalty for global model selection in semiparametric models, such as those proposed in Xue (2009) and Xue and Qu (2012). Here, we incorporate the local features of varying-coefficient models and ensure local sparsity of the dynamic modeling. Zhou, Wang and Wang (2013) incorporated a similar idea to detect zero subregions for the functional coefficients in a functional linear regression model using a two-step procedure.

Both $\boldsymbol{\beta}$ and $\boldsymbol{\sigma}$ are unknown parameters, but $\boldsymbol{\beta}$ is the main parameter of o interest. We need to specify $\boldsymbol{\sigma}$ to estimate $\boldsymbol{\beta}$ in the penalized loss (2.2). A two-step iterative procedure is proposed in the algorithm in the next section.

Let $\mathbf{y}_{iu} = (y_i^1(t_{1u}), \ldots, y_i^n(t_{nu}))'$, $\widetilde{\mathbf{y}}_{iu} = \sqrt{w_{iu}/nm}\ \mathbf{y}_{iu}$, $\widetilde{\mathbf{y}}_i = (\widetilde{\mathbf{y}}'_{i1}, \ldots, \widetilde{\mathbf{y}}'_{im})'$, and $\mathcal{Y}_n = (\widetilde{\mathbf{y}}'_1, \ldots, \widetilde{\mathbf{y}}'_p)'$ be an nmp-dimensional vector. Let $\mathcal{X}_n = (\widetilde{\mathbf{x}}'_{(1,2)}, \ldots, \widetilde{\mathbf{x}}'_{(p-1,p)})$ be an $(nmp) \times \{p(p-1)J_n/2\}$-dimensional matrix, with $\widetilde{\mathbf{x}}_{(i,j)} = (\mathbf{0}_1, \ldots, \mathbf{0}_{i-1}, \mathbf{z}^j_{(i,j)}, \mathbf{0}_{i+1}, \ldots, \mathbf{0}_{j-1}, \mathbf{z}^i_{(i,j)}, \ldots, \mathbf{0}_p)'$, where $\mathbf{0}_k = \{0\}_{J_n \times nm}$, and $\mathbf{z}^j_{(i,j)} = (\mathbf{z}^j_{(i,j),1}, \ldots, \mathbf{z}^j_{(i,j),m})'$, with

$$\mathbf{z}^j_{(i,j),u} = \left(\mathbf{B}(t_{1u}) \sqrt{\frac{\widetilde{\sigma}^{jj}(t_{1u})}{\widetilde{\sigma}^{ii}(t_{1u})}} y_j^1(t_{1u}), \ldots, \mathbf{B}(t_{nu}) \sqrt{\frac{\widetilde{\sigma}^{jj}(t_{nu})}{\widetilde{\sigma}^{ii}(t_{nu})}} y_j^n(t_{nu}) \right),$$

for $u = 1, \ldots, m$, and $\widetilde{\sigma}^{ii}(t_u) = \sigma^{ii}(t_u)/w_{iu}$. Then, the corresponding loss function (2.2) is equivalent to

$$L(\boldsymbol{\beta}, \boldsymbol{\sigma}, \mathcal{Y}_n) = \frac{1}{2} \|\mathcal{Y}_n - \mathcal{X}_n \boldsymbol{\beta}\|^2 + \sum_{i<j}^{p} \sum_{h=1}^{N_n+1} P_{\lambda_n}(\|\boldsymbol{\gamma}_h^{ij}\|). \tag{2.3}$$

Let $\hat{\boldsymbol{\beta}}$ be the minimizer of objective functions (2.2) or (2.3). Then, the resulting estimator for the partial correlation function $\rho^{ij}(t)$ is defined as $\hat{\rho}^{ij}(t) = (\hat{\beta}^{ij})^T \mathbf{B}(t)$.

3. Implementation

3.1. Algorithms

In this section, we propose an algorithm that determines an optimal solution for the objective function (2.3). Let the penalty function $P_{\lambda_n}(\|\gamma_h^{ij}\|)$ in (2.3) follow the adaptive LASSO penalty (Tibshirani (1996); Zou (2006)); that

is, $P_\lambda(\|\gamma_h^{ij}\|) = \lambda_n \tau_h^{ij} \|\gamma_h^{ij}\|$, where $\tau_h^{ij} = 1/\|\tilde{\gamma}_h^{ij}\|^r$ with $r > 0$, and $\tilde{\gamma}_h^{ij}$ is a consistent estimator of γ_h^{ij}. So the penalty term can be considered as an adaptive group LASSO with overlapping groups. When the groups overlap, if one group is shrunk to zero, all coefficients in the group shrink to zero as well, even though some belong to other nonzero-coefficient groups. The solution space and theoretical properties of the group LASSO with overlaps are discussed in Jenatton, Audibert and Bach (2011) and Obozinski, Jacob and Vert (2011), who indicate that traditional LASSO algorithms cannot be applied directly to the penalized loss function in (2.2).

However, because the dual norm of the L_2-norm is still the L_2-norm, the L_2-norm $\boldsymbol{\gamma}_h^{ij}$ can be formulated as $\max_{\|\boldsymbol{\alpha}_h^{ij}\| \le 1} (\boldsymbol{\alpha}_h^{ij})' \boldsymbol{\gamma}_h^{ij}$, where $\boldsymbol{\alpha}_h^{ij} \in R^{(q+1)}$ is an auxiliary vector associated with $\boldsymbol{\gamma}_h^{ij}$. A similar transformation and its properties is discussed in Chen et al. (2012); Jacob, Obozinski and Vert (2009), and Obozinski, Jacob and Vert (2011). Let $Q = \{\boldsymbol{\alpha} | \|\boldsymbol{\alpha}_h^{ij}\| \le 1, 1 \le i < j \le p, h = 1, \ldots, N_n + 1\}$. We can rewrite the group adaptive LASSO penalty for the overlapping parameters in (2.2) as follows:

$$g_0(\boldsymbol{\beta}) = \lambda_n \sum_{i<j}^{p} \sum_{h=1}^{N_n+1} \tau_h^{ij} \|\boldsymbol{\gamma}_h^{ij}\| = \max_{\boldsymbol{\alpha} \in \mathbf{Q}} \sum_{i<j}^{p} \sum_{h=1}^{N_n+1} \lambda_n \tau_h^{ij} (\boldsymbol{\alpha}_h^{ij})' \boldsymbol{\gamma}_h^{ij} = \max_{\boldsymbol{\alpha} \in \mathbf{Q}} \boldsymbol{\alpha}' C \boldsymbol{\beta}, \tag{3.1}$$

where $C \in R^{[(q+1)(N_n+1)p(p-1)/2] \times [p(p-1)J_n/2]}$ is an indicator matrix, with each element defined as

$$C_{(k,l)} = \begin{cases} \lambda_n \tau_h^{ij} & k = (r-1)(N_n+1)(q+1) + (h-1)(q+1) + v, \\ & l = (r-1)J_n + (h-1) + v, \\ 0 & \text{otherwise}, \end{cases}$$

where $r = (i-1)(p-i+2) + (j-i-1)$ and $v = 1, \ldots, (q+1)$. Note that C is a very sparse matrix, with only one nonzero element in each row. Thus it only requires a relatively small amount of memory storage in the optimization procedure. As a result of the transformation, the group penalization terms no longer present overlapping parameters.

However, this introduces a new problem, because the penalty function $g_0(\boldsymbol{\beta})$ in (3.1) is a nonsmooth function of $\boldsymbol{\beta}$. To resolve this problem, we need to build a smooth function to approximate $g_0(\boldsymbol{\beta})$. Let $D = \max_{\boldsymbol{\alpha} \in \mathbf{Q}} \|\boldsymbol{\alpha}\|^2/2$ and

$$g_\mu(\boldsymbol{\beta}) = \max_{\boldsymbol{\alpha} \in \mathbf{Q}} \left(\boldsymbol{\alpha}' C \boldsymbol{\beta} - \frac{\mu}{2} \|\boldsymbol{\alpha}\|^2 \right), \tag{3.2}$$

where μ is the tolerance parameter. Then, $g_\mu(\boldsymbol{\beta})$ is a quadratic approximation

for $g_0(\boldsymbol{\beta})$, with a maximum difference of μD. That is,

$$g_0(\boldsymbol{\beta}) - \mu D \leq g_\mu(\boldsymbol{\beta}) \leq g_0(\boldsymbol{\beta}).$$

In order to control the maximum difference, we choose the tolerance level $\epsilon = \mu D$ or, equivalently, $\mu = \epsilon/D$. Consequently, the loss function in (2.3) can be approximated by

$$\widetilde{PL}(\mu, \boldsymbol{\beta}, \boldsymbol{\sigma}) = \frac{1}{2}\|\mathcal{Y}_n - \mathcal{X}_n\boldsymbol{\beta}\|^2 + g_\mu(\boldsymbol{\beta}).$$

To minimize the loss function $\widetilde{PL}(\mu, \boldsymbol{\beta})$, we need to calculate the gradient of $\widetilde{PL}(\mu, \boldsymbol{\beta})$. For any $\mu > 0$, $g_\mu(\boldsymbol{\beta})$ is convex and continuously differentiable and the corresponding gradient function $\nabla g_\mu(\boldsymbol{\beta})$ is $C'\boldsymbol{\alpha}^*$, where $\boldsymbol{\alpha}^*$ is the optimal solution to (3.2). Let $\boldsymbol{u}_h^{ij} = \lambda_n \tau_h^{ij} \boldsymbol{\gamma}_h^{ij}/\mu$. Then the closed form of $\boldsymbol{\alpha}^*$ can be expressed as

$$(\boldsymbol{\alpha}_h^{ij})^* = \begin{cases} \dfrac{\boldsymbol{u}_h^{ij}}{\|\boldsymbol{u}_h^{ij}\|}, & \text{if } \left\|\boldsymbol{u}_h^{ij}\right\| > 1, \\ \boldsymbol{u}_h^{ij}, & \text{if } \left\|\boldsymbol{u}_h^{ij}\right\| \leq 1. \end{cases} \tag{3.3}$$

Therefore, the partial derivative $\nabla\widetilde{PL}(\mu, \boldsymbol{\beta}, \boldsymbol{\sigma})$ with respect to $\boldsymbol{\beta}$ can be calculated as $\mathcal{X}_n' (\mathcal{X}_n\beta - \mathcal{Y}_n) + C'\alpha^*$. Moreover, $\nabla\widetilde{PL}(\mu, \boldsymbol{\beta}, \boldsymbol{\sigma})$ is Lipschitz-continuous, with the Lipschitz constant

$$M = \lambda_{\max}\left(\mathcal{X}_n'\mathcal{X}_n\right) + \frac{\|C\|^2}{\mu},$$

where $\lambda_{\max}$ is the largest eigenvalue of $(\mathcal{X}_n)'\mathcal{X}_n$ and $\|C\| = \max_{\|\boldsymbol{\alpha}\|\leq 1} \|C\boldsymbol{\alpha}\|$. The proximal operator can be defined as

$$Q_L(\boldsymbol{\beta}, \boldsymbol{\beta}', \boldsymbol{\sigma}) = \left\{\widetilde{PL}(\mu, \boldsymbol{\beta}', \boldsymbol{\sigma}) + \nabla PL(\mu, \boldsymbol{\beta}', \boldsymbol{\sigma})(\boldsymbol{\beta} - \boldsymbol{\beta}') + \frac{M}{2}\|\boldsymbol{\beta} - \boldsymbol{\beta}'\|^2\right\},$$

and $\boldsymbol{\beta}$ can be updated at the $(l+1)$th iteration by applying the proximal-gradient algorithm through

$$\begin{aligned}\boldsymbol{\beta}^{(l+1)} = \operatorname{argmin}_{\boldsymbol{\beta}} Q_L(\boldsymbol{\beta}, \boldsymbol{\beta}^{(l)}, \boldsymbol{\sigma}) = \operatorname{argmin}_{\boldsymbol{\beta}} \Big\{&\widetilde{PL}(\mu, \boldsymbol{\beta}^{(l)}, \boldsymbol{\sigma}^{(l)}) \\ &+\nabla PL(\mu, \boldsymbol{\beta}^{(l)}, \boldsymbol{\sigma}^{(l)})(\boldsymbol{\beta} - \boldsymbol{\beta}^{(l)}) + \frac{M}{2}\|\boldsymbol{\beta} - \boldsymbol{\beta}^{(l)}\|^2\Big\}.\end{aligned} \tag{3.4}$$

Convergence is guaranteed because the inequality $\widetilde{PL}(\mu, \boldsymbol{\beta}^{(l+1)}, \boldsymbol{\sigma}^{(l)}) \leq Q_L(\boldsymbol{\beta}, \boldsymbol{\beta}^{(l)}, \boldsymbol{\sigma}^{(l)})$ holds for each iteration. It is not difficult to check whether the inequality holds; see Chen et al. (2012) for a detailed discussion. The above penalization strategy achieves sparsity corresponding to the group parameters $\boldsymbol{\gamma}_h$; however, it does not guarantee the sparsity of each element in $\hat{\boldsymbol{\beta}}$ obtained

Algorithm 1 Proximal-gradient algorithm for estimating partial correlation networks

Input: Set desired tolerance levels ϵ and ϵ^* (set to be 10^{-3}), obtain $\mu = \epsilon/D$ and matrix C, and calculate the step size M; initialize the parameters $\boldsymbol{\beta}, \boldsymbol{\sigma}$ as $\boldsymbol{\beta}^{(0)}$ and $\boldsymbol{\sigma}^{(0)}$, respectively.

Output: $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{\sigma}}$.

1: Compute $\boldsymbol{\alpha}^*$ according to (3.3) and calculate $\nabla\widetilde{PL}(\boldsymbol{\beta}^{(l)}, \mu) = \mathcal{X}_n'(\mathcal{X}_n\boldsymbol{\beta}^{(l)} - \mathcal{Y}_n) + C'\boldsymbol{\alpha}^*$;

2: Obtain $\boldsymbol{\beta}^{(l+1)}$ by minimizing (3.4), i.e., $\boldsymbol{\beta}^{(l+1)} = arg\min_{\boldsymbol{\beta}} Q_L(\boldsymbol{\beta}^{(l)}, \boldsymbol{\beta})$, and set the elements in $\boldsymbol{\beta}^{(l+1)}$ less than ϵ^* as zero;

3: Update $\boldsymbol{\sigma}^{(l+1)}$ and $\mathbf{w}^{(l+1)}$ by calculating (3.5);

4: Return to Step 1 if $\|Q_L(\boldsymbol{\beta}^{(l+1)}, \boldsymbol{\beta}^{(l)}, \boldsymbol{\sigma}^{(l+1)}) - Q_L(\boldsymbol{\beta}^{(l)}, \boldsymbol{\beta}^{(l-1)}, \boldsymbol{\sigma}^{(l)})\| > \epsilon$.

from (3.4). Alternatively, we can set $\beta_h^{ij} = 0$ if $\|\beta_h^{ij}\| < \epsilon^*$ for a small tolerance level ϵ^*. For σ, if each subject is observed at the same time over m time-points (i.e., $t_{ku} = t_u$, for any $k = 1, \ldots, n$ and $u = 1, \ldots, m$), then each component of $\boldsymbol{\sigma}^{(l+1)} = \left\{\left((\sigma^{11})^{(l+1)}(t_u), \ldots, (\sigma^{pp})^{(l+1)}(t_u)\right)\right\}_{u=1}^{m}$ at the $(l+1)$th iteration can be updated by

$$\frac{1}{(\sigma^{ii})^{(l+1)}(t_u)} = \frac{1}{n}\sum_{k=1}^{n}\left(y_i^k(t_u) - \sum_{j\neq i}^{p}\sum_{h=1}^{J_n}(\beta_h^{ij})^{(l)}B_h^{ij}(t_u)\sqrt{\frac{(\sigma^{jj})^{(l)}(t_u)}{(\sigma^{ii})^{(l)}(t_u)}}y_j^k(t_u)\right)^2, \tag{3.5}$$

and the weight component for the ith subject is $w_{iu}^{(l+1)} = (\sigma^{ii})^{(l+1)}$. If each subject is observed at the m time-points, we can update $(\sigma^{ii})^{(l+1)}(t)$ using a polynomial spline estimation method. Let $\hat{\varepsilon}_i^2(t_{ku}) = (y_i^k(t_{ku}) - \sum_{j\neq i}^{p}\sum_{h=1}^{J_m}(\beta_h^{ij})^{(l)} B_h^{ij}(t_{ku})\sqrt{((\sigma^{jj})^{(l)}(t_{ku}))/((\sigma^{ii})^{(l)}(t_{ku}))}y_j^k(t_{ku}))^2$. For each $i = 1, \ldots, p$, we can estimate $\sigma^{ii}(t)$ by a polynomial spline regression, using $\left\{1/\hat{\varepsilon}_i^2\,(t_{ku})\right\}_{k=1,u=1}^{n,m}$ as the response variables and the spline basis generated on time-points $\{(t_{ku})\}_{k=1,u=1}^{n,m}$ as the explanatory variables. We summarize the algorithm as follows.

Algorithm 2 Alternating direction method of multipliers for estimating partial correlation networks

Input: Set desired tolerance levels ϵ, ϵ^*, and scalar κ, obtain $\mu = \epsilon/D$ and matrix C; initialize the parameters $\boldsymbol{\beta}, \boldsymbol{\sigma}$ as $\boldsymbol{\beta}^{(0)}$ and $\boldsymbol{\sigma}^{(0)}$.

Output: $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{\sigma}}$, respectively.

1: Compute $\boldsymbol{\alpha}^{*(l)}$ according to (3.3);

2: Obtain $\boldsymbol{\beta}^{(l+1)}, \boldsymbol{\beta}^{*(l+1)}, \boldsymbol{\eta}^{(l+1)}$ according to (3.8), and set the elements in $\boldsymbol{\beta}^{(l+1)}$ less than ϵ^* as zero;

3: Update $\boldsymbol{\sigma}^{(l+1)}$ and $\mathbf{w}^{(l+1)}$ by calculating (3.5);

4: Return to Step 1 if $\|\boldsymbol{\beta}^{(l+1)} - \boldsymbol{\beta}^{*(l+1)}\| > \epsilon$.

We can also apply the alternating direction method of multipliers (ADMM) (Boyd et al. (2011)) to approximate $g_0(\boldsymbol{\beta})$ by $g_\mu(\boldsymbol{\beta})$ in (3.2), as follows. The adaptive LASSO with overlapping group penalty can be solved using a constrained optimization:

$$\begin{aligned} &\min_{\boldsymbol{\beta},\boldsymbol{\beta}^*} \frac{1}{2}\|\mathcal{Y} - \mathcal{X}\boldsymbol{\beta}\|^2 + g_\mu(\boldsymbol{\beta}^*), \\ &s.t. \quad \boldsymbol{\beta} = \boldsymbol{\beta}^*. \end{aligned} \tag{3.6}$$

This can be further formulated as a scaled augmented Lagrangian problem:

$$L_\rho = \frac{1}{2}\|\mathcal{Y} - \mathcal{X}\boldsymbol{\beta}\|^2 + g_\mu(\boldsymbol{\beta}^*) + \frac{\kappa}{2}\|\boldsymbol{\beta} - \boldsymbol{\beta}^* + \boldsymbol{\eta}\|_2^2, \tag{3.7}$$

where $\boldsymbol{\eta}$ are dual variables, and κ is a scalar and can be preset. Therefore, the ADMM algorithm solving (3.7) leads to three iteration steps for $\boldsymbol{\beta}, \boldsymbol{\beta}^*$, and $\boldsymbol{\eta}$. That is, at the $(l+1)$th iteration,

$$\begin{aligned} \boldsymbol{\beta}^{(l+1)} &= arg\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathcal{Y} - \mathcal{X}\boldsymbol{\beta}\|^2 + \frac{\kappa}{2}\|\boldsymbol{\beta} - \boldsymbol{\beta}^{*(l)} + \boldsymbol{\eta}^{(l)}\|_2^2, \\ \boldsymbol{\beta}^{*(l+1)} &= arg\min_{\boldsymbol{\beta}} g_\mu(\boldsymbol{\beta}^*) + \frac{\kappa}{2}\|\boldsymbol{\beta}^{(l+1)} - \boldsymbol{\beta}^* + \boldsymbol{\eta}^{(l)}\|_2^2, \\ \boldsymbol{\eta}^{(l+1)} &= \boldsymbol{\eta}^{(l)} + (\boldsymbol{\beta}^{(l+1)} - \boldsymbol{\beta}^{*(l+1)}). \end{aligned} \tag{3.8}$$

The first minimization problem in (3.8) is easy to solve because the objective function is quadratic. The function $g_\mu(\boldsymbol{\beta}^*)$ in the second minimization is a smoothing function and, thus, can be approximated by the Taylor expansion at $\boldsymbol{\beta}^{*(l)}$; that is $g_\mu(\boldsymbol{\beta}^*) \approx g_\mu(\boldsymbol{\beta}^{*(l)}) + 1/2\nabla g_\mu(\boldsymbol{\beta}^{*(l)})(\boldsymbol{\beta}^* - \boldsymbol{\beta}^{*(l)})$. Thus, $\nabla g_\mu(\boldsymbol{\beta}^*) \approx \nabla g_\mu(\boldsymbol{\beta}^{*(l)})/2 = C'\boldsymbol{\alpha}^{*(l)}/2$, where $\boldsymbol{\alpha}^{*(l)}$ can be calculated by (3.3) corresponding to $\boldsymbol{\beta}^{*(l)}$. Therefore, the solution is $\boldsymbol{\beta}^{*(l+1)} = \boldsymbol{\beta}^{(l+1)} + \boldsymbol{\eta}^{(l)} - \lambda C'\boldsymbol{\alpha}^{*(l)}/(2\kappa)$. The algorithm is summarized as Algorithm 2:

Both the SPG and ADMM provide approximations of (3.1); however, they use different approximation methods and, therefore, yield different the final solutions. The proximal-gradient method has the following advantages: (1) we can construct a smoothing approximation to the objective function, which makes the convergence fast; and (2) it does not require a large matrix inversion and only involves sparse matrix operations. These benefits reduce algorithm complexity and improve the computational speed significantly. On the other hand, the ADMM requires the inversion of a matrix, which may be infeasible when the network size is large. More details are provided in Section 5.

3.2. Tuning parameter selection

The choice of tuning parameters is critical, because this determines the performance of the proposed method. The tuning parameter selection for the varying-coefficient model involves two parts. First, we select the sequence of knots for the polynomial spline. Second, we select the tuning parameter in the penalty function. For simplicity, we set the number of knots to be the same order of $n^{1/(2q+3)}$, where n is the sample size and q is the order of the polynomial spline. This choice of the number of knots balances between the variance and the squared bias of the polynomial spline estimators (Huang (1998); Xue and Yang (2006); Huang, Zhang and Zhou (2007)). We can also use a data-driven knot number, which can be selected using a similar a BIC procedure that described below. A more detailed discussion on knot selection can be found in Huang, Wu and Zhou (2004); Xue, Qu and Zhou (2010); Xue and Qu (2012). However, for convenience, we select equally spaced knots in our numerical studies. Nevertheless, our theory is developed under a more general setup that allows for more flexible choices of knot sequence.

To select tuning parameters associated with the penalty function, we use the BIC, which is documented in the model selection literature (e.g., Qu and Li (2006); Wang, Li and Tsai (2007)). Specifically, given the tuning parameters λ_n, denote the estimator $\widehat{\boldsymbol{\beta}}_{\lambda_n}$, and calculate the estimators $\widehat{\boldsymbol{\sigma}}_{\lambda_n}$ and $\widehat{w}_{\lambda_n}$ using (3.5). Let κ_n be the total number of nonzero elements in $\widehat{\boldsymbol{\beta}}_{\lambda_n}$. Then, the BIC is given as $BIC(\lambda_n) = nm \log\{MSE(\lambda_n)\} + \kappa_n \log(nm)$, with

$$MSE(\lambda_n) = \frac{1}{nm}\sum_{k=1}^{n}\sum_{i=1}^{p}\sum_{u=1}^{m}\widehat{w}_{iu,\lambda_n}\left(y_i^k(t_{ku}) - \sum_{j\neq i}^{p}\sum_{h=1}^{J_n}\widehat{\beta}_{h,\lambda_n}^{ij}B_h^{ij}(t_{ku})\sqrt{\frac{\widehat{\sigma}_{\lambda_n}^{jj}(t_{ku})}{\widehat{\sigma}_{\lambda_n}^{ii}(t_{ku})}}y_j^k(t_{ku})\right)^2.$$

The optimal tuning parameter $\widehat{\lambda}_n$ is selected by minimizing $BIC(\lambda_n)$.

4. Asymptotic Theory

In this section, we investigate the asymptotic properties of the varying-coefficient estimator $\widehat{\boldsymbol{\rho}}(t)$ based on the polynomial spline approximation. Because a distinct feature of our approach is the estimation and selection of local features in dynamic network modeling, we focus on establishing the local-feature model selection consistency of $\widehat{\boldsymbol{\rho}}(t)$. That is, if the true $\boldsymbol{\rho}(t)$ is zero for any given region, the estimator of $\boldsymbol{\rho}(t)$ is zero with probability approaching one.

Before presenting the asymptotic properties of the proposed model, we first introduce the following regularity conditions, which are required to establish the asymptotic properties.

C1: The weights $\{w_{it}\}_{i=1}^{p}$ are uniformly finite for $t \in I$. That is, there exist positive constants w_0 and w_∞ such that $0 < w_0 \leq \min_i\{w_{it}\} \leq \max_i\{w_{it}\} \leq w_\infty < \infty$, for any $t \in I$.

C2: There exists a constant c such that $\max_{1\leq i\leq p} \sup_{t\in\mathbf{I}} |\widehat{\boldsymbol{\sigma}}^{ii}(t) - \boldsymbol{\sigma}^{ii}(t)| \leq c\sqrt{(log(nm)N_n)/(nm)}$ holds, with probability approaching to one as the sample size $n \to \infty$.

C3: We assume that for any $t \in \mathbf{I}$, $\mathbf{y}(t)$ has mean $\mathbf{0}$ and covariance matrix $\Sigma(t)$, the eigenvalues of which are assumed to be uniformly bounded for $t \in I$. That is, $0 < \inf_{t\in\mathbf{I}} \lambda_{\min}(\Sigma(t)) \leq \sup_{t\in\mathbf{I}} \lambda_{\max}(\Sigma(t)) < \infty$, where $\lambda_{\min}$ and $\lambda_{\max}$ are the minimum and maximum eigenvalues of $\Sigma(t)$, respectively. Furthermore, for some sufficiently large $l > 0$, $\sup_{t\in\mathbf{I}} E\,|Y_i\,(t)|^l < +\infty$, for $i = 1, \ldots, p$.

C4: The observation times $\{t_{ku}\}_{k=1,u=1}^{n,m}$ are independent and follow a distribution $f_T(t)$ on I, and $f_T(t)$ is absolutely continuous and bounded away from zero and infinity.

C5: For $1 \leq i \neq j \leq p$, the partial correlation function $\rho^{ij}(\cdot)$ has q continuous derivatives, with $q \geq 1$.

C6: For $1 \leq i \neq j \leq p$, let $E^{ij} \subset I$ be the null region such that $\rho^{ij}(t) = 0$ if $t \in E^{ij}$, and $\rho^{ij}(t) \neq 0$ if $t \in (E^{ij})^c$. If $E^{ij} \neq \emptyset$, we assume that $E^{ij} = [e_1^{ij}, e_2^{ij}]$ is a closed interval. Let $\dot{\rho}^{ij}(t)$ be the first-order derivative of $\rho^{ij}(t)$. We assume there exists a constant $C > 0$ such that $\left|\dot{\rho}^{ij}(t)\right| \geq C$, for any $t \in [e_1^{ij} - \epsilon, e_1^{ij}] \cup [e_2^{ij}, e_2^{ij} + \epsilon]$ and a small constant $\epsilon > 0$.

C7: The set of knots, denoted as $\Upsilon_n = \{0 = \nu_0 < \nu_1 < \cdots < \nu_{N_n} < \nu_{N_n+1} = 1\}$, is quasi-uniform; that is, there exists $b > 0$ such that

$$\frac{\max(\nu_{h+1} - \nu_h, h = 0, \ldots, N_n)}{\min(\nu_{h+1} - \nu_h, h = 0, \ldots, N_n)} \leq b.$$

C8: The number of interior knots N_n and tuning parameters λ_n satisfy

$$\frac{\lambda_n N_n}{\alpha_n} \to 0, \frac{\lambda_n N_n^2}{\alpha_n} \to \infty, \frac{\lambda_n\sqrt{N_n nm/\log(nm)}}{\alpha_n} \to \infty,$$

where $\alpha_n = \sqrt{N_n/nm} + N_n^{-1}$.

Condition C1 indicates that the weights are bounded away from zero and infinity. Condition C2 assumes there exists a consistent estimator for $\sigma^{ii}(t)$, for each $i = 1, \ldots, p$. Similar conditions to C1 and C2 can also be found in Peng et al. (2009). In the Supplementary Material, we propose an estimator that meets this condition by kernel smoothing the residuals of a least-square fitting, as discussed in the algorithm. Conditions C3, C4, C5, and C7 are standard conditions in a polynomial spline framework, and are required to ensure the consistency of the spline estimation in the varying coefficient model. Similar conditions can be found in Huang, Wu and Zhou (2002); Xue and Qu (2012), and Wang et al. (2014). Condition C6 divides the time regions into those with zero and nonzero correlations, leading consistency of the partial correlation estimators.

To present our theoretical results, we first introduce an oracle estimator, which estimates each $\rho^{ij}(t)$ under the assumption that the null regions of each $\rho^{ij}(t)$ are known. It is constructed only for the proof of the asymptotic results, and is not useful for analyzing real data. Note that, for each end point of the null region $E^{ij} = [e_1^{ij}, e_2^{ij}]$ in condition (C6), there exist knots $\nu_{l_1^{ij}}$ and $\nu_{l_2^{ij}}$ in the knot sequence $\Upsilon = \{0 = \nu_0 < \nu_1 < \cdots < \nu_{N_n} < \nu_{N_n+1} = 1\}$, such that $e_1^{ij} \in [\nu_{l_1^{ij}}, \nu_{l_1^{ij}+1})$ and $e_2^{ij} \in [\nu_{l_2^{ij}-1}, \nu_{l_2^{ij}})$. Let $J_{ij} = \left\{1, \ldots, \nu_{l_1^{ij}} - 2, \nu_{l_2^{ij}} + q + 2, \ldots, J_n\right\}$. An oracle estimator $\widetilde{\beta}^{(o)} = \left\{\widetilde{\beta}_h^{ij(o)}, 1 \le h \le J_n, 1 \le i < j \le p\right\}$ is constructed by taking all coefficients $\widetilde{\beta}_h^{ij(o)} = 0$, for $h = \nu_{l_1^{ij}-1}, \ldots, \nu_{l_2^{ij}} + q + 1$, and estimating the remaining coefficients by minimizing the sum of the squares

$$\frac{1}{2nm}\sum_{i=1}^{p}\sum_{k=1}^{n}\sum_{u=1}^{m} w_{iu}\left(y_i^k(t_{ku}) - \sum_{j\neq i}^{p}\sum_{h\in J_{ij}} \beta_h^{ij} B_h(t_{ku})\sqrt{\frac{\widehat{\sigma}^{jj}(t_{ku})}{\widehat{\sigma}^{ii}(t_{ku})}} y_j^k(t_{ku})\right)^2. \quad (4.1)$$

Denote the resulting oracle estimator of the partial coefficient functions by $\widetilde{\rho}^{ij}(t)$, for $1 \le i < j \le p$. Then, the oracle estimators enjoy both estimation consistency and null-region selection consistency, as indicated in the following theorem.

Theorem 1. *Under conditions* (C1)–(C8)*, for any* $1 \le i < j \le p$*, the oracle estimators satisfy*

$$\left\|\widetilde{\rho}^{ij(o)} - \rho^{ij}\right\|_2 = O_p\left(\sqrt{\frac{N_n}{nm}} + N_n^{-1}\right),$$

$$\sup_{t\in\mathbf{I}}\left|\widetilde{\rho}^{ij(o)}(t) - \rho^{ij}(t)\right| = O_p\left(\frac{N_n^{3/2}}{\sqrt{nm}} + N_n^{-1}\right). \quad (4.2)$$

In addition, let $\widetilde{E}^{ij} = \left\{t \in I, \widetilde{\rho}^{ij}(t) = 0\right\}$ *be the corresponding null region of*

$\widetilde{\rho}^{ij(o)}(t)$. *Then,* $E^{ij} \subset \widetilde{E}^{ij}$, *and the set* $\widetilde{E}^{ij} \backslash E^{ij}$ *converges to the empty set with probability approaching one as* $n \to \infty$.

Theorem 2. *Under conditions* (C1)–(C8), *when* n *is sufficiently large, the minimizer* $\left\{\widehat{\rho}^{ij}\right\}_{1 \le i < j \le p}$ *of the penalized likelihood function in* (2.2) *satisfies* $\|\widehat{\rho}^{ij} - \rho^{ij}\|_2 = O_p(\sqrt{N_n/(nm)} + N_n^{-1})$, *for any* $1 \le i < j \le p$.

Theorem 3. *Under conditions* (C1)–(C8), *for any* $1 \le i < j \le p$, *let* $\widehat{E}^{ij} = \left\{t \in I, \widehat{\rho}^{ij}(t) = 0\right\}$ *be the corresponding null region of* $\widehat{\rho}^{ij}(t)$. *Then,* $E^{ij} \subset \widehat{E}^{ij}$, *and the set* $\widehat{E}^{ij} \backslash E^{ij}$ *converges to the empty set with probability approaching one as* $n \to \infty$.

Theorem 2 shows that the estimator that minimizes the penalized loss function (2.2) is L_2-consistent when estimating the partial correlation functions. Furthermore, Theorem 3 shows that, with probability approaching one, the estimator correctly identifies zero estimators in the nonsignal time regions. Therefore, the proposed method can correctly produce a locally sparse network and efficiently model the dynamic changes in large volumes of network data. The proof of the theorem is provided in the Supplementary Material.

Note that Theorems 2 and 3 assume that the network structure changes smoothly over time (e.g., Condition C5). Therefore, the proposed spline method is developed for networks with smooth changes.

5. Simulation

In this section, we conduct simulation studies to illustrate the performance of the proposed SPG described in Section 3. We first compare the performance of the SPG method using different degrees of polynomial spline. Then, the proposed approach with the best order of B-spline approximation is selected for the comparison with other existing approaches, such as SPACE (Peng et al. (2009)), the kernel-based method (Kolar, Parikh and Xing (2010)), and the ADMM. Note that the ADMM does not apply directly to our dynamic partial correlation networks, because the original ADMM is not formulated for overlapping parameters from penalty terms. Therefore, we provide an adaptation of the ADMM approach to accommodate our setting. We also compare the proposed method with the time-varying undirected graph (TVUG) model proposed by Zhou, Lafferty and Wasserman (2010) and the varying-coefficient and varying-structure graphic model (VCVS) proposed by Kolar and Xing (2012). Specifically, Zhou, Lafferty and Wasserman (2010) develop a kernel-based nonparametric method for esti-

mating time-varying covariance matrices for multivariate Gaussian distributions using an l_1-regularization. As such, the authors show that the TVUG model is able to obtain l_1-penalized maximum likelihood estimators at each time-point, as long as the covariances change smoothly over time. The VCVS model is based on the neighborhood selection procedure (Meinshausen and Bühlmann (2006)), which allows the coefficients of the precision matrix to change in a piecewise constant fashion. That is, their model assumes that the network structures change abruptly, rather than changing smoothly, by incorporating both a modified fused LASSO penalty and a LASSO penalty.

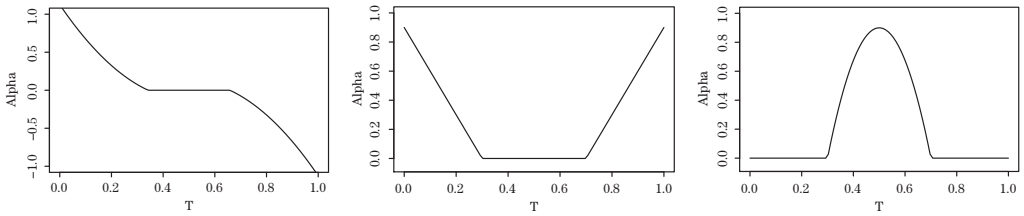
We generate dynamic networks by assuming that the network structures have disjoint blocks. Such networks are quite common in applications where networks are connected within blocks, but are not associated with each other between blocks. See Girvan and Newman (2002) and Valencia et al. (2009) for more examples on brain and biological functions, gene expressions, and social, sports, and computer network associations. In the following simulations, the number of disjoint blocks is three. To generate the concentration matrix at time t, we first create an initial matrix $(A_t)_{p\times p}$ with three blocks, as

$$\begin{pmatrix} A_t^1 & & \\ & A_t^2 & \\ & & A_t^3 \end{pmatrix},$$

where the diagonal entries for each block $A_t^k (k = 1, 2, 3)$ are set to one, and each off-diagonal entry of A^k is set to $f_k(t)U$, where U follows a Bernoulli distribution with $Pr(U = 1) = \omega$. The blocks A_t^k are exchangeable, because the partial correlations between the nodes of networks are undirected and interchangeable. We use ω to control the number of nonzero elements in A_t^k and, thus, the sparsity within each block, such that the networks are sparse if ω is small. We consider moderate associations between the nodes of the network, and therefore choose $\omega = 0.8$ in our settings. The functions $f_k(t)$, for $k = 1, 2, 3$, are defined as follows:

$$f_1(t) = \begin{cases} 5(t-0.5)^2 - 0.125, & \text{if} \quad 1 \le t \le 0.342, \\ 0, & \text{if} \quad 0.342 < t \le 0.658, \\ -5(t-0.5)^2 + 0.125, & \text{if} \quad 0.658 < t \le 1, \end{cases}$$

$$f_2(t) = \begin{cases} -3t + 0.9, & \text{if} \quad 0 \le t \le 0.3, \\ 0, & \text{if} \quad 0.3 < t \le 0.7, \\ 3t - 2.1, & \text{if} \quad 0.7 < t \le 1, \end{cases}$$



(a) $f_1(t)$ (b) $f_2(t)$ (c) $f_3(t)$

Figure 2. The function $f(t)$ at time interval $t \in [0, 1]$.

Table 1. Model selection performance of the SPG method for three-block disjoint networks with time-points $T = 50$ and sample size 200, based on 100 simulation runs.

	Network size	C	O	U	Sensitivity	Specificity	Time per run (seconds)
Linear	p=18	0.920	0.056	0.024	0.802	0.967	27.46
	p=54	0.859	0.071	0.070	0.679	0.910	467.71
	p=108	0.830	0.023	0.147	0.772	0.836	3,726.48
Quadratic	p=18	0.887	0.063	0.050	0.760	0.932	41.87
	p=54	0.838	0.073	0.089	0.642	0.888	670.89
	p=108	0.799	0.088	0.113	0.560	0.859	7,510.36
Cubic	p=18	0.860	0.091	0.049	0.688	0.931	60.13
	p=54	0.791	0.099	0.110	0.526	0.861	1,192.30
	p=108	0.764	0.113	0.123	0.474	0.843	14,102.38

and

$$f_3(t) = \begin{cases} -22.5(t-0.5)^2 + 0.9, & \text{if} \quad 0.3 \le t \le 0.7, \\ 0, & \text{if} \quad \text{o.w.}. \end{cases}$$

The plots of $f_k(t)$ are provided in Figure 2. After constructing a concentration matrix, we follow a similar strategy to that in Peng et al. (2009) to ensure that the simulated covariance matrix is positive-definite.

We first compare the performance of the local signal selection using the linear, quadratic, and cubic spline approximations in the simulation studies. We consider network sizes of $p = 18, 54$, and 108, and time length $T = 50$. The sample size is chosen as $n = 200$.

Table 1 compares the model selection performance of the SPG method in detecting the true time-varying signals under different orders of spline approximations. Here correct-fitting (C), over-fitting (O), and under-fitting (U) are calculated as the percentage of T equally spaced time-points in the interval [0, 1], where the true signal and nonsignal points are identified correctly, true nonsignal

points are misclassified as signal points; and true signal points are not selected, respectively. In addition, we calculate the sensitivity and specificity, as defined by Peng et al. (2009). Here the sensitivity is the ratio of the number of correctly detected signals to the number of true signals; the specificity is the ratio of the number of correctly detected signals to the number of detected signals.

Table 1 indicates that the SPG with a linear spline tends to select correct edges with the highest frequency, compared with the quadratic and cubic splines. When the network size increases from 18 to 108, the percentage of selecting correct associations decreases by about 9.8% in the linear spline approach. When the network size is 108, the percentage of selecting correct edges based on the SPG is about 83.0% for the linear spline approach. In addition, the overall sensitivity and specificity rates are best when using the linear spline approach. This simulation indicates that the SPG with a linear spline performs best in terms of detecting local changes in network associations, compared with the quadratic and cubic splines.

We further compare the performance of the proposed model with the SPACE, kernel-based method (KEN), ADMM approach, TVUG model, and VCVS method. We compare the performance of these methods under the network sizes of 18, 54, and 108, with a sample size $n = 200$ and time length $T = 50$, based on 100 simulations. Because Table 1 indicates that the SPG method with a linear spline outperforms the quadratic and cubic splines, we use the linear spline for the SPG in the following comparison.

Table 2 provides the model selection performance of the SPG, ADMM, SPACE, KEN, TVUG, and VCVS under various network sizes. The SPG and ADMM exhibit similar performance, and are best in terms of selecting the true model with the highest frequency when the network size is 18 or 54. When the network size increases to 108, the rates of selecting the correct model for SPACE and VCVS decrease to 51.2% and 66.7%, respectively. This is probably due to the over-fitting problem. For the TVUG, the correct-fitting rate is down to 75.4%. In comparison, the SPG still has a correct-fitting rate of 83.0%. However, neither the ADMM nor the KEN is feasible owing to the problem of a high-dimensional matrix inversion for the ADMM approach and a highly intensive computing procedure for the kernel method. The ADMM requires inverting large-dimensional matrices if p is large. We tried the SparseM package in R, and the Eigen package and SparseLib++ in C++, which are designed for large-dimensional matrix operations. However, when the dimension of a matrix is beyond that which the package can handle, the ADMM approach becomes infeasible.

Table 2. Model selection performance of SPG, ADMM, SPACE, KEN, and VCVS for three-block disjoint networks with time-points $T = 50$ and sample size 200, based on 100 simulation runs.

Network size	Methods	C	O	U	Sensitivity	Specificity	Time per run (seconds)
p=18	SPG	0.920	0.056	0.024	0.802	0.967	27.46
	ADMM	0.920	0.055	0.025	0.804	0.965	10.53
	SPACE	0.907	0.082	0.011	0.745	0.984	1.33
	KEN	0.909	0.065	0.026	0.775	0.963	109.35
	TVUG	0.880	0.079	0.041	0.726	0.942	2.03
	VCVS	0.901	0.052	0.047	0.796	0.937	25.49
p=54	SPG	0.859	0.071	0.070	0.679	0.910	467.71
	ADMM	0.860	0.068	0.072	0.685	0.908	286.87
	SPACE	0.691	0.220	0.089	0.373	0.863	36.39
	KEN	0.786	0.123	0.091	0.512	0.878	14,328.74
	TVUG	0.820	0.096	0.084	0.586	0.891	26.79
	VCVS	0.748	0.127	0.124	0.430	0.840	123.94
p=108	SPG	0.830	0.023	0.147	0.772	0.836	3,726.48
	ADMM	NA	NA	NA	NA	NA	NA
	SPACE	0.512	0.418	0.070	0.271	0.836	349.98
	KEN	NA	NA	NA	NA	NA	NA
	TVUG	0.754	0.136	0.110	0.458	0.853	383.76
	VCVS	0.667	0.220	0.113	0.337	0.831	944.03

For the ADMM, the required number of iterations is $O(1/\epsilon)$ (Wang and Banerjee (2014)), given a desired accuracy ϵ. For the SPG, the convergence rate is also $O(1/\epsilon)$ (Chen et al. (2012)). The SPACE and TVUG are basically LASSO approaches; the computational complexity is the same as that of the quadratic programming algorithm, which is $O(n^3)$ in the worst case, where n is the sample size. For Kolar and Xing's (2012) approach, the accelerated gradient method also has a convergence rate of $O(1/\epsilon)$. For the kernel-based method, the computation complexity is due to the number of iterations, because the method only updates one parameter in each iteration. That is, if we have p nodes and m time-points, the model has $p(p-1)/2 * m$-dimensional parameters, where the number of parameters increases as the number of time points increases. This leads to an intensive calculation, because each iteration requires $p(p-1)/2 * m$ updates.

Table 2 also provides the average computing time per simulation run for each method. We performed our simulations on a cluster server running Linux equipped with a 2.67 GHz CPU and 48 GB memory. The computing time in-

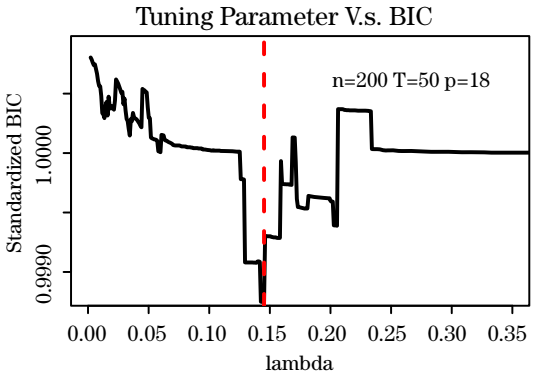


Figure 3. Plot of the moving tuning parameter versus the BIC for the SPG algorithm when $n = 200$, $T = 50$, and $p = 18$.

creases significantly because the dimension of the matrix operations increases exponentially from 10^2 to 10^5 when the network size increases from 18 to 108. The SPACE and TVUG are the fastest of the methods. This is because the SPACE does not utilize neighboring information of the time-points observed from the same subject. In the case of TVUG, the kernel-based sample covariance matrices can be preprocessed before minimization, and the covariance matrix is penalized through its determinant rather than for each element. KEN is the slowest of the methods, because it requires updating the neighborhood information for each nonparametric coefficient estimation at each iteration. The computing time ranges from 27.46 seconds to 1.04 hours per run for the SPG algorithm, and 25.49 seconds to 15.8 minutes per run for the VCVS method. We were not able to record the times for the KEN and ADMM when $p = 108$ owning to feasibility issues for these two approaches. In summary, the SPG performs best in terms of computational feasibility and correct-fitting performance.

We also compare the number of edges correctly identified by the SPG, KEN, SPACE, and ADMM with a moving tuning parameter. The TVUG and VCVS are not provided here, because they require two tuning parameters, making a comparison unsuitable. Figure 3 shows that the BIC reaches the minimum if the tuning parameter is selected as $\lambda = 0.145$ when the network size is 18, the sample size is 200, and number of time-points is 50. In addition, Figure 4 indicates that both the SPG and the ADMM have the highest ratio of correctly identified edges over total detected edges, for any given tuning parameter. For example, when the number of total detected edges is equal to the number of true edges (1,876), the SPG and ADMM are able to identify 1,444 and 1,441 correct edges, respectively,

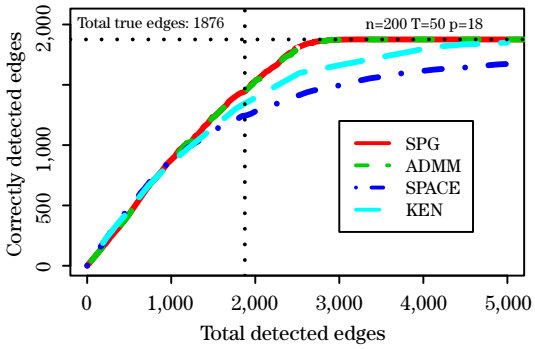


Figure 4. Correctly detected edges versus total detected edges using the four methods.

whereas the KEN detects 1,345 correct edges and the SPACE detects only 1,243 correct edges.

6. Application

In this section, we analyze data obtained from an attention deficit hyperactivity disorder (ADHD) study. ADHD is a mental disorder found in children and adolescents, and common symptoms include being easily distracted, impulsiveness, and restlessness. To better understand how ADHD patients' brains function and react to stimuli, we focus on identifying associations and interactions between different ROIs of the brain. A distinct feature of ADHD patients is that they have high variability of brain function over time; therefore, it is scientifically important to identify the dynamic changes in the ROIs of the brain in order to locate the ADHD pathology.

The ADHD-200 samples contain fMRI data, measured repeatedly over time. The data are available from `http://www.nitrc.org/frs/?group_id=383|`, which contains resting-state fMRI (rs-fMRI) data on 78 patients (mean age = 9.0 and s.d. = 1.12) from the Oregon Health & Science University, with 116 ROIs measured over 74 time points. Software for automated anatomical labeling was used to label macroscopic brain structures, which are used to categorize the brain into 116 ROIs (`http://neuro.imm.dtu.dk/wiki/Automated_Anatomical_Labeling|`). The patients were instructed to stay still, keep their eyes open, and focus on a standard fixation cross in the center of the display. Participants were scanned after a minimum washout of short-acting stimulant medications. The temporal-resolution of fMRI data is 2,500 ms.

We apply only the SPG and SPACE methods to these data, because the

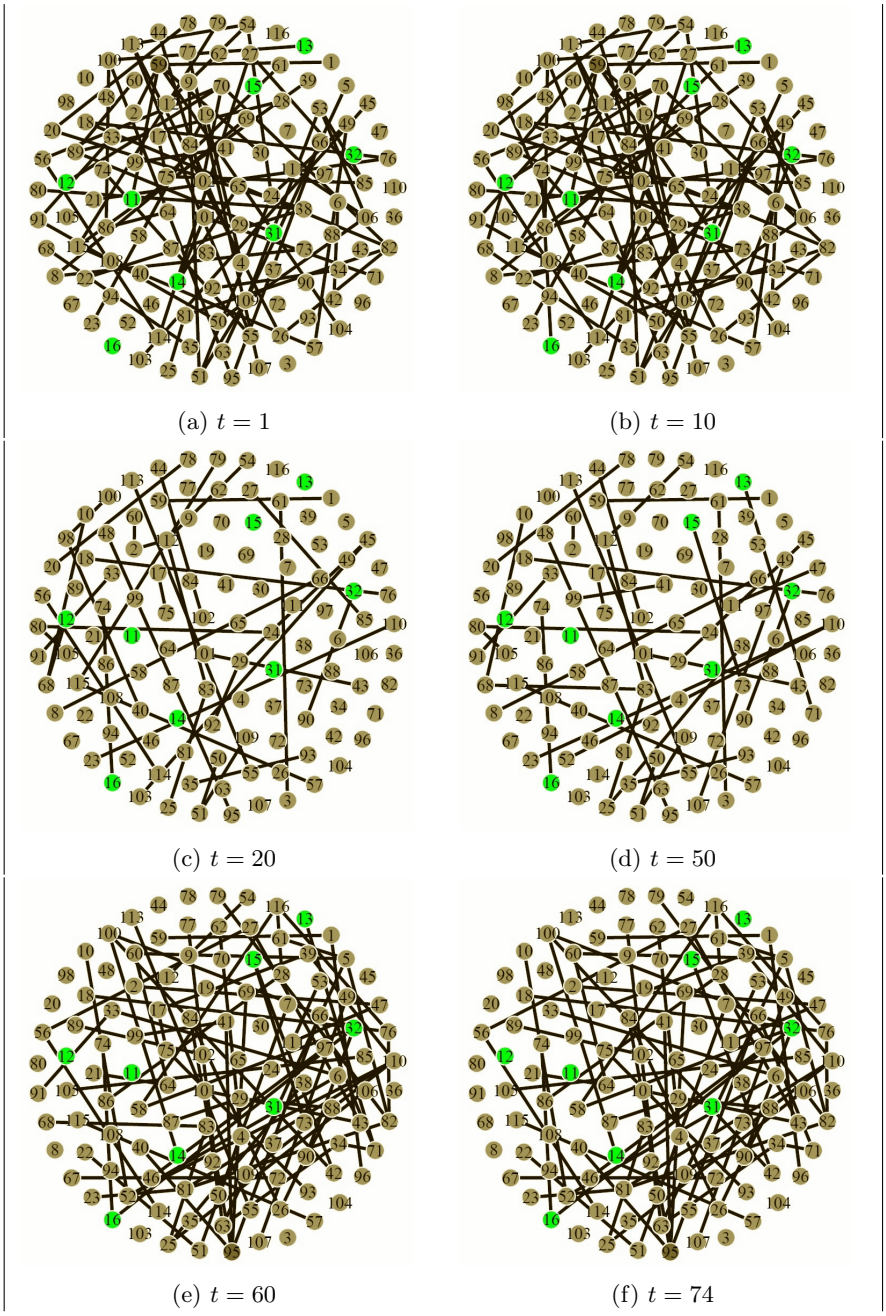
Table 3. Number of associations identified by SPG and SPACE from time-points 1 to 74.

Method	Number of associations from 1 to74
SPG	70 77 77 77 76 77 77 76 77 77 77 77 77 77 78 77 77 77 77
	35 36 35 35 35 35 35 35 35 35 35 35 35 35 36 35 35 35
	34 34 34 34 34 34 34 34 34 34 34 34 34 35 34 34 34 34
	76 76 76 75 76 76 76 76 76 77 76 76 76 76 76 76 76 76 66
SPACE	3,024 3,102 3,257 2,059 2,691 2,839 3,278 2,962 3,111 3,080 2,926
	2,946 2,833 3,079 3,171 3,156 3,067 2,932 3,129 2,955 2,934 3,025
	1,998 3,076 3,130 3,278 3,230 2,786 3,176 2,828 2,979 2,981 3,057
	3,045 2,695 3,070 2,665 3,120 3,090 2,916 3,054 2,982 2,670 3,038
	2,836 2,969 3,006 3,154 2,756 3,056 3,179 3,024 2,975 2,974 3,067
	3,273 1,956 3,157 2,707 3,132 3,115 2,948 2,799 2,967 3,028 3,059
	2,969 3,165 3,089 3,039 3,109 2,950 3,103 2,779

ADMM and KEN approaches are not able to handle a network size of 116. The numbers of connections between the ROIs at each time point are shown in Table 3. Note that the SPACE method identifies more than 2,000 connections at most of the time-points. In contrast, the SPG method identifies at most 78 connections at each time-point. The over-identifying problem of the SPACE method makes it difficult to select any useful connections. In the following, we provide a data analysis and a graphical illustration based on the SPG method only.

Figure 5 illustrates the associations and connections of the 116 ROIs formulated as a network at time-points $t = 1, 10, 20, 50, 60$, and 74. Note that each ROI in the brain is represented as a node or a vertex with either a green or a pink color. The associations between the nodes are shown as blue lines. A pink node represents five or more associations with other ROIs. A green node indicates fewer than five associations with other ROIs.

We are able to identify the dynamic changes of associations between the 116 ROIs over time. Specifically, the ADHD patients experience three distinct periods of brain activity during the test. The numbers of connections at each time-point are shown in Table 3. At the beginning of the test, the ADHD patients' brains are active. However, as the test proceeds, the patients' brains are mostly in a resting state, because there are few connections between the 116 ROIs, with most of the ROIs containing fewer than 36 connections. This is possibly because that patients are less disturbed in the middle of the experiment, because there is actually no stimulus imposed on the brain. In the latter part of the test, when $t > 57$, patients' brains again have more connections between ROIs, because patients might anticipate something happening by the end of the experiment.



(a) $t = 1$ (b) $t = 10$

(c) $t = 20$ (d) $t = 50$

(e) $t = 60$ (f) $t = 74$

Figure 5. Estimation of brain networks using ADHD-200 data at time-points $t = 1, 10, 20, 50, 60$, and 74.

Table 4. ROIs with five or more associations identified by SPG from time-points 1 to 74.

Time(t)	ROIs with 3 or more associations	Total
1-19	24 38 51 53 54 59 70 75 82 85 89 100 106 113 115	15
20-55	83 112	2
58-74	5 25 32 52 63 71 76 81 82 90 95 100 110 116	14

These phenomena are also indicated in Figure 5, showing that there are more associations between the ROIs for $t = 1$ and $t = 10$, and $t = 60$ and $t = 74$, but fewer brain activities for $t = 20$ and $t = 50$.

Table 4 confirms our findings and indicates that there are few associations between $t = 20$ and $t = 55$, with only two vertices having three or more connections during this period. However, between time-points $t = 1$ and 19, there are 15 vertices containing three or more connections between ROIs, and between $t = 56$ and 74, there are 14 vertices with three or more connections. The corresponding names of the ROIs with three or more connections and their gray levels are provided in Table 5 (gray level is defined as the percentage of gray matter in an ROI. Gray matter as distinguished from white matter, consists of cell bodies, neuropil, glial cells, and capillaries). These findings could be helpful in studying ADHD patients' brain function over time, even without any stimulation.

Compared with task-based fMRI experiments, results from resting-state fMRI studies can be more easily synthesized, because they investigate the differences between ADHD patients' ROIs connected in the absence of tasks. Fox and Greicius (2010) and Greicius (2008) studied the connections between any two ROIs, and used two-sample t-tests to infer whether the average strength of a connection between two ROIs is significantly different between ADHD and healthy patients. Dickstein et al. (2006) found that several ROIs consistently under-activated among patients with ADHD. These include portions of the frontal lobe: anterior cingulate cortex (ACC) (regions 31 and 32 in AAL), dorsolateral prefrontal cortex (DLPFC), and inferior prefrontal cortex (11-16, AAL), along with portions of the basal ganglia, thalamus, and parietal cortices. Hart et al. (2013) discovered that portions of the frontal lobe (the inferior frontal cortex, ACC, and supplemental motor area), basal ganglia, and thalamus are under-activated in response to inhibition tasks among ADHD patients. Furthermore, patients with ADHD showed under-activation in the DLPFC, parietal areas, basal ganglia, and thalamus in response to attention tasks. In Figure 5, we highlighted the nodes in our network graphs. Nodes 11–16 and 31, 32 are not active, except that node 32

Table 5. ROIs with three or more associations identified by SPG.

Number	Name	Gray level
5	Frontal_Sup_Orb_L	2,111
24	Frontal_Sup_Medial_R	2,602
25	Frontal_Mid_Orb_L	2,611
32	Cingulum_Ant_R	4,002
38	Hippocampus_R	4,102
51	Occipital_Mid_L	5,201
52	Occipital_Mid_R	5,202
54	Occipital_Inf_R	5,302
59	Parietal_Sup_L	6,101
63	SupraMarginal_L	6,211
70	Paracentral_Lobule_R	6,402
71	Caudate_L	7,001
75	Pallidum_L	7,021
76	Pallidum_R	7,022
82	Temporal_Sup_R	8,112
83	Temporal_Pole_Sup_L	8,121
85	Temporal_Mid_L	8,201
89	Temporal_Inf_L	8,301
90	Temporal_Inf_R	8,302
95	Cerebelum_3_L	9,021
100	Cerebelum_6_R	9,042
106	Cerebelum_9_R	9,072
110	Vermis_3	9,110
112	Vermis_6	9,130
113	Vermis_7	9,140
115	Vermis_9	9,160
116	Vermis_10	9,170

becomes active at the end of the test (with four connections). Thus, in general, our analysis results are consistent with the findings in the existing literature, as mentioned above.

Figure 6 describes the changes of associations between three ROIs (right-middle frontal gyrus, right gyrus rectus, and right angular gyrus) at $t = 1, 20$, and 60. The ROIs are indicated in as green if they are associated with each other at certain time-points. Figure 7(a) illustrates the locations of certain ROIs in the brain using an automated anatomical labeling (AAL) software package. Here, the ROIs are marked using different colors. Note that most of the ROIs have counterparts located on the opposite side of the brain, and are marked using the same color. For example, the cyan blue color is used for both Temporal_Mid_L and Temporal_Mid_R in Figure 7 (a). However, these counterpart ROIs are not

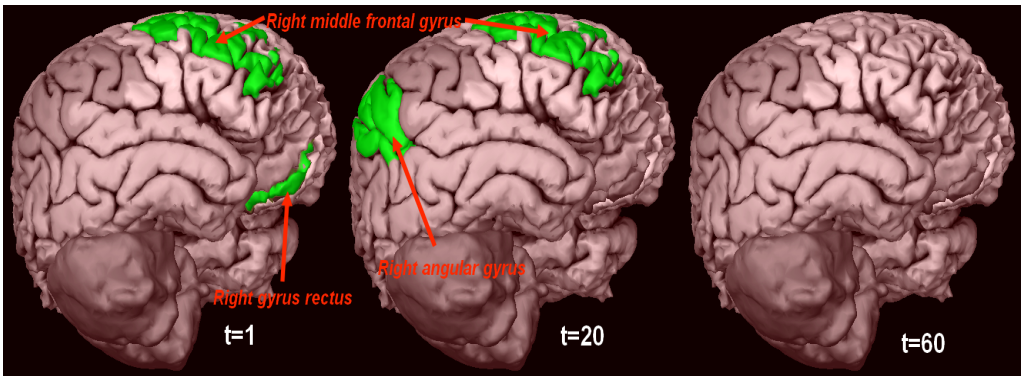


Figure 6. Changes of associations between the right-middle frontal gyrus, right gyrus rectus, and right angular gyrus over three time-points.

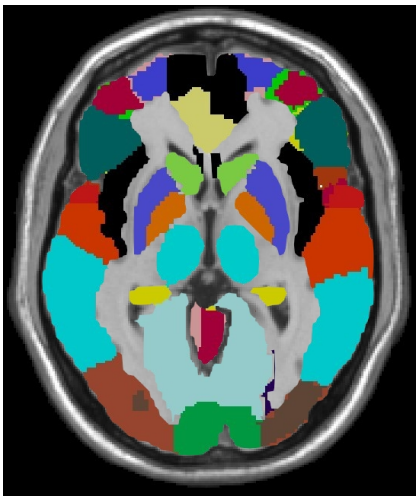
necessarily associated with each other. Figure 7(a) shows 50 of the 116 ROIs, and Figure 7(b) provides a partial network of the ROIs to illustrate the associations based on the 15 selected ROIs. The partial network is quite sparse. For a better visualization of the associated network, Figure 7(c) also provides the associated names of the 15 selected ROIs.

In addition, we provide an animated video ("ADHD.mp4") to illustrate the dynamic changes for the 116 ROIs of the brain over 74 time points. The colors of the nodes in the video range from red to purple, blue, and green, reflecting the level of connections with other ROIs over the entire period. The red nodes are the most active ROIs, with the number of connections ranging from 30 to 36; the purple nodes have 18-29 connections, and the blue and green nodes have moderate to few associations (8 to 17, and 0 to 7, respectively) with other ROIs of the brain.

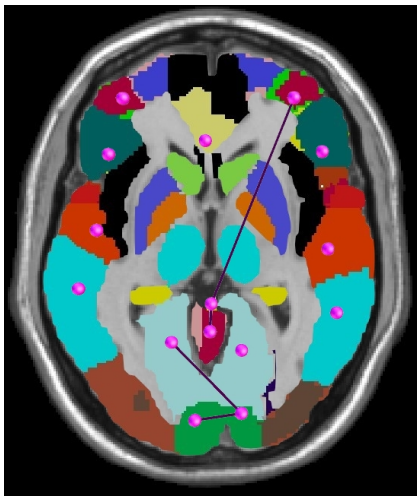
7. Discussion

The time-varying network model is powerful for identifying time-evolving associations for brain and biological functions, gene networks, social networks, and environmental networks over time. In this study, we develop a local varying-coefficient model to effectively quantify and detect dynamic changes in network associations and interactions. A distinctive feature of the proposed approach is that we are able to incorporate local features of a varying-coefficient function. Furthermore, we provide local signal detection and estimation simultaneously for time-varying network data.

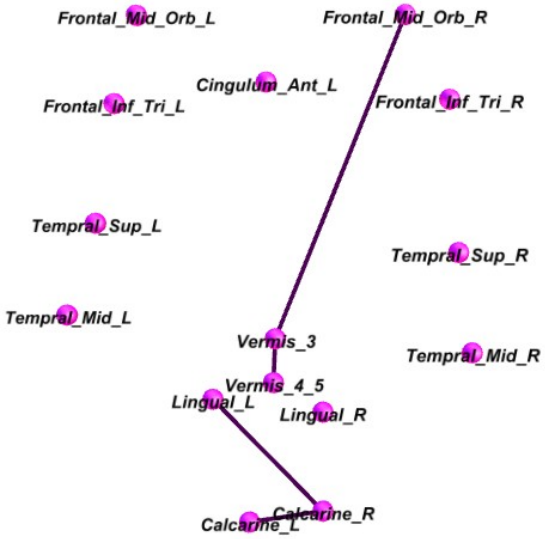
We propose a piecewise penalized loss function, such that the coefficients



(a) Locations of AAL ROIs in the brain



(b) The partial network of ROIs



(c) ROIs in the partial network

Figure 7. Illustration of AAL ROIs in the brain and its networks.

associated with the varying-coefficient model at the local region shrink to zero if the magnitude of the grouped coefficients is sufficiently small. This has significant advantages over the traditional varying-coefficient model selection approach that does not incorporate local features, especially for time-varying network data. This is because the network associations can be quite volatile over time, and

local-region estimations and signal detection are of greater scientific interest than global feature selection. Our simulations and data application to the ADHD study indicate that the proposed method is quite effective at capturing the local features of the time-varying network data.

However, it is challenging to develop computationally intensive algorithms that achieve sparsity properties in estimations and signal detection at local time intervals. The group penalization strategy employs parameters that overlap between groups, which makes the optimization process extremely challenging when the network size is large. To overcome these difficulties, we develop a smoothing proximal-gradient method, which does not require inverting a large-dimensional matrix. The proposed algorithm has significant computational advantages in terms of increased computational speed and efficiency. Most importantly, the proposed algorithm is able to analyze a relatively large quantity of network data within a reasonable time frame. We also compare out method with the ADMM and kernel-based algorithms, which require inverting a large-dimensional matrix, and therefore cannot feasibly estimate large amounts of network data.

From a theoretical viewpoint, we show that the proposed method achieves model selection consistency in local regions and provides a uniform rate of convergence for local-signal coefficient estimators. Scientifically, it is important to detect dynamic changes in networks, because identifying the associations between biological functions over time can help us to better understand the mechanisms underlying network changes.

The proposed method is developed for networks with a fixed dimension. For a high-dimensional network, we suggest first using screen methods to reduce the dimensionality. For example, we can use a global selection method similar to those in Xue (2009) and Xue and Qu (2012) to delete pairs of variables that are not associated/connected in the entire region. Then, for the remaining pairs, we can apply the proposed method to locate the time region where this association might change.

We do not consider the longitudinal dependence structure over time in our estimation, although this can be incorporated using either the generalized estimation equation (Liang and Zeger (1986)) or the quadratic inference function approach, as in Xue, Qu and Zhou (2010) and Wang et al. (2014). However, incorporating the dependence structure does not effect the convergence rate, as in Section 4, but does affect the estimation efficiency. This is left for future research.

Supplementary Material

The Supplementary Material includes detailed proofs of the main theorems and necessary lemmas.

Acknowledgments

Xue's research was supported by the Simons Foundation (F0782A) and National Science Foundation (DMS-1812258). Qu's research was supported by the National Science Foundation (DMS-1308227, DMS-1415308, and DMS-1613190).

References

Boyd, S., Parikh, N., Chu, E., Peleato, B. and Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning* **3**, 1–122.

Chen, Z. and Leng, C. (2016). Dynamic covariance models. *Journal of the American Statistical Association* **111**, 1196–1207.

Chen, X., Lin, Q., Kim, S., Carbonell, J. G. and Xing, E. P. (2012). Smoothing proximal gradient method for general structured sparse regression. *The Annals of Applied Statistics* **6**, 719–752.

Cheng, M.-Y., Honda, T. and Zhang, J.-T. (2016). Forward variable selection for sparse ultra-high dimensional varying coefficient models. *Journal of the American Statistical Association* **111**, 1209–1221.

Chung, F. R. K. (1997). Spectral graph theory. *CBMS Regional Conference Series in Mathematics*, No. 92.

Dickstein, S. G., Bannon, K., Castellanos, F. X. and Milham, M. P. (2006). The neural correlates of attention deficit hyperactivity disorder: An ALE meta-analysis. *Journal of Child Psychology and Psychiatry* **47**, 1051–1062.

Fan, J. and Gijbels, I. (1996). *Local Polynomial Modelling and Its Applications*. Chapman and Hall, London.

Fox, M. D. and Greicius, M. (2010). Clinical applications of resting state functional connectivity. *Frontiers in Systems Neuroscience* **4**, 19.

Friedman, J. H., Hastie, T. and Tibshirani, R. (2007). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* **9**, 432–441.

Friston, K. J., Harrison, L. and Penny, W. (2003). Dynamic causal modelling. *Neuroimage* **19**, 1273–1302.

Girvan, M. and Newman, M. E. J. (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences* **99**, 7821–7826.

Greicius, M. (2008). Resting-state functional connectivity in neuropsychiatric disorders. *Current Opinion in Neurology* **21**, 424–430.

Guo, J., Levina, L., Michailidis, G. and Zhu, J. (2011). Joint estimation of multiple graphical models. *Biometrika* **98**, 1–15.

Hart, H., Radua, J., Nakao, T., Mataix-Cols, D. and Rubia, K. (2013). Meta-analysis of

functional magnetic resonance imaging studies of inhibition and attention in attention deficit/hyperactivity disorder: Exploring task-specific, stimulant medication, and age effects. *JAMA Psychiatry* **70**, 185–198.

Hastie, T. and Tibshirani, R. (1993). Varying-coefficient models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **55**, 757–796.

Huang, J. Z. (1998). Projection estimation in multiple regression with application to functional ANOVA models. *The Annals of Statistics* **26**, 242–272.

Huang, J. Z., Wu, C. O. and Zhou, L. (2002). Varying-coefficent models and basis function approximations for the analysis of repeated measurements. *Biometrika* **89**, 111–128.

Huang, J. Z., Wu, C. O. and Zhou, L. (2004). Polynomial spline estimation and inference for varying coefficient models with longitudinal data. *Statistica Sinica* **14**, 763–788.

Huang, J. Z., Zhang, L. and Zhou, L. (2007). Efficient estimation in marginal partially linear models for longitudinal/clustered data using splines. *Scandinavian Journal of Statistics* **34**, 451–477.

Jacob, L., Obozinski, G. and Vert, J. P. (2009). Group lasso with overlap and graph lasso. In *Proceedings of the International Conference on Machine Learning*, 433–440.

Jenatton, R., Audibert, J.-Y. and Bach, F. (2011). Structured variable selection with sparsity inducing norms. *Journal of Machine Learning Research* **12**, 2777–2824.

Kim, W. H., Adluru, N., Chung, M. K., Charchut, S., GadElkarim, J. J., Altshuler, L., Moody, T., Kumar, A., Singh, V. and Leow, A. D. (2013). Multi-resolutional brain network filtering and analysis via wavelets on non-Euclidean space. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 643–651.

Kolaczyk, E. D. (2009). *Statistical Analysis of Network Data: Methods and Models.* New York, Springer.

Kolar, M., Parikh, A. and Xing, E. P. (2010). On sparse nonparametric conditional covariance selection. *The 27th International Conference on Machine Learning.*

Kolar, M., Song, L. and Xing, E. P. (2009). Sparsistent learning of varying-coefficient models with structural changes. *Advances in Neural Information Processing Systems* **23**, 1006–1014.

Kolar, M. and Xing, E. P. (2009). Sparsistent estimation of time-varying discrete Markov random fields. *arXiv Preprint ArXiv:0907.2337.*

Kolar, M. and Xing, E. P. (2011). On time varying undirected graphs. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, 407–415.

Kolar, M. and Xing, E. P. (2012). Estimating networks with jumps. *Electronic Journal of Statistics* **6**, 2069–2106.

Lebre, S., Becq, J., Devaux, F., Stumpf, M. P. and Lelandais, G. (2010). Statistical inference of the time-varying structure of gene-regulation networks. *BMC Systems Biology* **4**, 130–145.

Liang, K. Y. and Zeger, S. L. (1986). Longitudinal data analysis using generalized linear models. *Biometrika* **73**, 13–22.

Lee, H., Lee, D. S., Kang, H., Kim, B. N. and Chung, M. K. (2011). Sparse brain network recovery under compressed sensing. *IEEE Transactions on Medical Imaging* **30**, 1154–1165.

Leonardi, N. and Van De Ville, D. (2011). Wavelet frames on graphs defined by fMRI functional connectivity. In *2011 IEEE International Symposium*, 2136–2139.

Meinshausen, N. and Bühlmann, P. (2006). High-dimensional graphs and variable selection with

the LASSO. *The Annals of Statistics* **34**, 1436–1462.

Obozinski, G., Jacob, L. and Vert, G. (2011). Group lasso with overlaps: The latent group lasso approach. *arXiv Preprint ArXiv:1110.0413.*

Peng, J., Wang, P., Zhou, N. and Zhu, J. (2009). Partial correlation estimation by joint sparse regression models. *Journal of the American Statistical Association* **104**, 735–746.

Qu, A. and Li, R. (2006). Quadratic inference functions for varying coefficient models with longitudinal data. *Biometrics* **62**, 379–391.

Shen, X., Huang, H. and Pan, W. (2012). Simultaneous supervised clustering and feature selection over a graph. *Biometrika* **99**, 899–914.

Shojaie, A. and Michailidis, G. (2010). Network enrichment analysis in complex experiments. *Statistical Applications in Genetics and Molecular Biology* **9**, Art. 22.

Song, L., Kolar, M. and Xing, E. P. (2009). KELLER: Estimating time-evolving interactions between genes. *Bioinformatics* **25**, 128–136.

Stephan, K. E., Kasper, L., Harrison, L. M., Daunizeau, J., den Ouden, H. E., Breakspear, M. and Friston, K. J. (2008). Nonlinear dynamic causal models for fMRI. *Neuroimage* **42**, 649–662.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **58**, 267–288.

Tibshirani, R., Sauders, M., Zhu, J. and Knight, K. (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **67**, 91–108.

Valencia, M., Pastor, M. A., Fernadez-Seara, M. A., Artieda, J., Martinerie, J. and Chavez, M. (2009). Complex modular structure of large-scale brain networks. *Chaos* **19**, 023–119.

Wang, H. and Banerjee, A. (2014). Bregman alternating direction method of multipliers. In *Advances in Neural Information Processing System*, 2816–2824.

Wang, H., Li, R. and Tsai, C.-L. (2007). Tuning parameter selectors for the smoothly clipped absolute deviation method. *Biometrika* **94**, 553–568.

Wang, L., Xue, L., Qu, A. and Liang, H. (2014). Estimation and model selection in generalized additive partial linear models for correlated data with diverging number of covariates. *The Annals of Statistics.* **42**, 592–624.

Wang, R., Lin, P. and Wu, Y. (2015). Exploring dynamic temporal-topological structure of brain network within ADHD. In *Advances in Cognitive Neurodynamics (IV)*, 643–651.

Wee, C. Y., Yang, S., Yap, P. T., Shen, D. and Alzheimer's Disease Neuroimaging Initiative. (2016). Sparse temporally dynamic resting-state functional connectivity networks for early MCI identification. *Brain Imaging and Behavior* **10**, 342–356.

Wei, F., Huang, J. and Li, H. (2011). Variable selection and estimation in high-dimensional varying coefficient models. *Statistica Sinica.* **21**, 1515–1540.

Wee, C.-Y., Yap, P.-T., Zhang, D., Wang, L. and Shen, D. (2012). Constrained sparse functional connectivity networks for MCI classification. *Medical Image Computing and Computer-assisted Intervention - MICCAI* **15**, 212–219.

Xue, L. (2009). Variable selection in additive models. *Statistica Sinica.* **19**, 1281–1296.

Xue, L. and Qu, A. (2012). Variable selection in high-dimensional varying-coefficient models with global optimality. *Journal of Machine Learning Research* **13**, 1973–1998.

Xue, L., Qu, A. and Zhou, J. (2010). Consistent model selection for marginal generalized additive

model for correlated data. *Journal of the American Statistical Association* **105**, 1518–1530.

Xue, L. and Yang, L. (2006). Additive coefficient modeling via polynomial spline. *Statistica Sinica*, 1423–1446.

Zhou, J., Wang, N. Y. and Wang, N. (2013). Functional linear model with zero-value coefficient function at sub-region. *Statistica Sinica.* **23**, 25–50.

Zhou, S., Lafferty, J. and Wasserman, L. (2010). Time-varying undirected graphs. *Machine Learning* **80**, 295–319.

Zhu, Y., Shen, X. and Pan, W. (2013). Simultaneous grouping pursuit and feature selection in regression over an undirected graph. *Journal of the American Statistical Association* **108**, 713–725.

Zhu, Y., Shen, X. and Pan, W. (2014). Structural pursuit over multiple undirected graphs. *Journal of the American Statistical Association* **109**, 1683–1696.

Zou, H. (2006). The adaptive LASSO and its oracle properties. *Journal of the American Statistical Association* **101**, 1418–1429.

Department of Statistics, Oregon State University, Corvallis, OR 97331-4606, USA.

E-mail: xuel@science.oregonstate.edu

Merck, Kenilworth, NJ 07033, USA.

E-mail: xinxin.shu@merck.com

Department of Statistics, University of Illinois at Urbana-Champaign, Champaign, IL 61820, USA.

E-mail: anniequ@illinois.edu

(Received May 2017; accepted March 2018)