Inference of Time-Varying Gene Networks using Constrained and Smoothed Kalman Filtering

Ghulam Rasool and Nidhal Bouaynaya

Abstract—**This paper tackles the problem of recovering time-varying gene networks from a series of undersampled and noisy observations. Gene regulatory networks evolve over time in response to functional requirements in the cell and environmental conditions. Collected genetic profiles from dynamic biological processes, such as cell development, cancer progression and treatment recovery, underlie genetic interactions that rewire over the course of time. We formulate the problem of estimating time-varying networks in a state-space framework. We show that, due to the small number of measurements, the system is unobservable; thus making the application of the standard Kalman filter ineffective. We remedy the problem by performing simultaneous compression and state estimation. The sparsity property of gene regulatory networks is incorporated as a constraint in the Kalman filter, leading to a compressed Kalman estimate and reducing the number of required observations for effective tracking of the network. Moreover, we improve the estimation accuracy by taking into account the entire sample set for each time instant estimate of the network through a forward-backward smoothing procedure. The proposed constrained and smoothed Kalman filter is shown to yield good tracking results for varying small and medium-size networks.**

I. Introduction

Deciphering the complex dynamic nature of genetic regulatory networks (GRNs) is crucial for understanding cellular system dynamics, which can be fostered into educated control mechanisms and design principles for therapeutic targeting and drug design. The literature about reverse-engineering of genetic networks from high-throughput data is replete with various mathematical, statistical and graphical methods. These methods, however, infer a time-invariant network, i.e., a network with fixed genetic interactions and structure over the course of time. Biological processes, however, are dynamic and evolve over time in response to various intrinsic and extrinsic factors, such as cellular development, disease progression, targeted therapy and environmental conditions.

A major difficulty in inferring time-varying GRNs is the limited number of available measurements compared to the number of nodes (here genes) , i.e., “small n, large p” problem. This problem, which already exists in the inference of time-invariant networks, is even more severe in the time-varying case: At each time point, usually only one measurement is available. Thus, a naive formulation of the time-varying inference problem, which requires the estimation of a network given one measurement, is ill-posed. A plausible assumption on the network is that it evolves in phases or regimes, or equivalently that the underlying biological process is piecewise stationary. In this case, the time-series data is segmented into regimes and a time-invariant network is inferred in each regime segment [1]. This approach still suffers from a high variance in the estimators due to the limited number of data points in each segment. Alternatively, Ahmed and Xing [2] presented a regularized logistic regression method to capture the temporal rewiring of time-varying networks. Their model-based approach, however, reveals only the network skeleton and does not render the nature (inhibitive or stimulative) or strength of the interactions between the nodes, which are of crucial importance for biologists. Other graphical models, such as dynamic Bayesian networks (DBNs), have been extended to the time varying case [3]. In time-varying DBNs, the time-varying structure and parameters of the network are treated as additional hidden nodes in the graph model, and prior knowledge on their time evolution is required to update the model.

The two main approaches discussed above (segmentation into time-invariant regimes and extending existing graphical models to the time-varying case) do not take advantage of the sparse nature of gene regulatory networks. In this paper, we propose a different and new perspective to the inference of time-varying networks, which has full temporal resolution (i.e., uses the entire data set to estimate the network at each time instant) and takes into account the sparse nature of the network in order to overcome the undersampling (scarcity of measurements) problem. Specifically, we propose a constrained and smoothed Kalman filter to track the time-varying interactions between the nodes. The Kalman filter provides the optimum mean-square error of a time-varying signal with linear dynamics in Gaussian noise. The sparsity constraint on the filter overcomes the undersampling problem by reducing the number of required observations for a statistically meaningful estimation. The smoothing uses all available data points in the inference; thus improving the estimation by reducing its variance.

II. The State-Space Model

We model the concentrations of mRNAs, proteins, and other molecules using a time-varying ordinary differential equation (ODE). More specifically, the concentration of each molecule is modeled as a linear function of the concentrations of the other components in the system. The linearity of the ODE model can be justified if the system is operating near its steady-state. The time-dependent coefficients of the linear ODE capture the rewiring structure of the network. We have

$$\dot{x}_i(t) = -\lambda_i(t)x_i(t) + \sum_{j=1}^{p} w_{ij}(t)x_j(t) + b_i u(t) + v_i(t), \quad (1)$$

where $i = 1, \cdots, p$, p being the number of genes, $x_i(t)$ is the expression level of gene i at time t, $\dot{x}_i(t)$ is the rate of change of expression of gene i at time t, λ_i is the self degradation rate, $w_{ij}(t)$ represents the time-varying influence of gene j on gene i, b_i is the effect of the external perturbation $u(t)$ on gene i and $v_i(t)$ models the measurement and biological noise. The goal is to infer the time-varying gene interactions $\lambda_i(t), \{w_{ij}(t)\}_{i,j=1}^{p}$, given a limited number of measurements $n < p$.

To simplify the notation, we absorb the self degradation rate $\lambda_i(t)$ into the interaction parameters by letting $a_{ij}(t) = w_{ij}(t) - \lambda_i(t)\delta_{ij}$, where δ_{ij} is the Kronecker delta function. The external perturbation is assumed to be known. The model in (1) can be simplified by introducing a new variable

$$z_i(t) = \dot{x}_i(t) - b_i u(t). \tag{2}$$

The discrete-time equivalent of (1) can, therefore, be expressed as

$$z_i(k) = \sum_{j=1}^{p} a_{ij}(k) x_j(k) + v_i(k),\ i = 1, \cdots, p,\ k = 1, \ldots, n. \tag{3}$$

Writing (3) in matrix form, we obtain

$$\mathbf{z}(k) = A(k)\mathbf{x}(k) + \mathbf{v}(k), \tag{4}$$

where $\mathbf{z}(k) = [z_1(k), \ldots, z_p(k)]^T$, $A(k) = \{a_{ij}(k)\}$ is the matrix of time-dependent interactions, $\mathbf{x}(k) = [x_1(k), \ldots x_p(k)]^T$ and $\mathbf{v}(k) = [v_1(k), \ldots, v_p(k)]^T$. Let $\mathbf{a}(k) \in \mathbb{R}^{p^2}$ be the vectorized form of the matrix $A(k)$, i.e.,

$$\begin{aligned} \mathbf{a}(k) &= \text{vec}[A(k)^T] \\ &= [a_{11}(k), \ldots, a_{1p}(k), \ldots, a_{p1}(k), \ldots, a_{pp}(k)]^T \end{aligned} \tag{5}$$

where vec(.) is the vectorization operator. Using this notation, we can write

$$\begin{aligned} A(k)\mathbf{x}(k) &= \begin{bmatrix} x_1(k) \ldots x_p(k) & \mathbf{0} \\ \vdots & \vdots \\ \mathbf{0} & x_1(k) \ldots x_p(k) \end{bmatrix} \mathbf{a}(k) \\ &= [\mathbf{I}_p \otimes \mathbf{x}(k)^T]\mathbf{a}(k) = \Lambda(k)\mathbf{a}(k), \end{aligned} \tag{6}$$

where $\Lambda(k) = \mathbf{I}_p \otimes \mathbf{x}(k)^T$ is a $p \times p^2$ block diagonal matrix and $\otimes$ represents the Kronecker product. Therefore, the observation equation (4) becomes

$$\mathbf{z}(k) = [\mathbf{I}_p \otimes \mathbf{x}(k)^T]\ \mathbf{a}(k) + \mathbf{v}(k). \tag{7}$$

The state equation models the dynamics of the state vector $\mathbf{a}(k)$ given *a priori* knowledge of the system. In this work, we assume a random walk model of the network parameters. The random walk model is chosen for two reasons. First, it reflects a flat prior or a lack of *a priori* knowledge. Second, it leads to a smooth evolution of the state vector over time (if the variance of the random walk is not too high). The state space model of the time-varying network is, therefore, given by

$$\mathbf{a}(k+1) = \mathbf{a}(k) + \mathbf{w}(k), \tag{8}$$

$$\mathbf{z}(k) = [\mathbf{I}_p \otimes \mathbf{x}(k)^T]\mathbf{a}(k) + \mathbf{v}(k), \tag{9}$$

where $\mathbf{w}(k)$ and $\mathbf{v}(k)$ are, respectively, the process noise and the observation noise, assumed to be zero mean Gaussian noise processes with known covariance matrices. In addition, the process and observation noise are assumed to be uncorrelated with each other and with the state vector $\mathbf{a}(k)$. If the linear system is observable, then the minimum mean-square error (MSE) solution can be obtained using the Kalman filter. On the other hand, if the system is unobservable, then the regular Kalman filter cannot recover the optimal solution. In the sequel, we will show that the observability problem may be circumvented by taking into account the sparsity of the network.

III. Observability of the System

The time-varying linear system in (8, 9) is observable if the $p^2 \times p^2$ matrix $\mathcal{Q}(k)$

$$\mathcal{Q}(k) = \sum_{k=0}^{n} \Phi^T(k,0) H^T(k) H(k) \Phi(k,0). \tag{10}$$

is positive definite (equivalently does not have a zero as an eigenvalue or has a non-zero determinant). $\Phi(k,j)$ is the state transition matrix and $H(k) = \mathbf{I}_p \otimes \mathbf{x}(k)^T$ is the observation matrix. From the random walk state transition model, we have $\Phi(k,j) = \mathbf{I}_{p^2}^{(k-j)} = \mathbf{I}_{p^2}$. Equation (10) can, therefore, be simplified to

$$\mathcal{Q}(k) = \sum_{k=0}^{n} H^T(k) H(k). \tag{11}$$

Replacing the matrix $H(k)$ by its expression and using the properties of the Kronecker product, we have

$$\begin{aligned} \mathcal{Q}(k) &= \sum_{k=0}^{n} \left[\mathbf{I}_p^T \otimes \mathbf{x}(k)\right] \left[\mathbf{I}_p \otimes \mathbf{x}^T(k)\right], \\ &= \sum_{k=0}^{n} \left[\mathbf{I}_p \otimes \{\mathbf{x}(k)\mathbf{x}^T(k)\}\right] \\ &= \mathbf{I}_p \otimes \sum_{k=0}^{n} \mathbf{x}(k)\mathbf{x}^T(k). \end{aligned} \tag{12}$$

The matrix summation on the right side of the Kronecker product in (12) is the sum of n rank 1 matrices. Its rank is, therefore, upper bounded by the sum of the ranks of matrices, i.e.,

$$\text{rank}\left[\sum_{k=0}^{n} \mathbf{x}(k)\mathbf{x}^T(k)\right] \leq \sum_{k=0}^{n} \text{rank}\left[\mathbf{x}(k)\mathbf{x}^T(k)\right] = n. \tag{13}$$

Finally, using the property of the Kronecker product for the rank of matrices, we obtain from (12)

$$\begin{aligned} \text{rank}[\mathcal{Q}(k)] &= \text{rank}\left[\mathbf{I}_p\right] \times \text{rank}\left[\sum_{k=0}^{n} \mathbf{x}(k)\mathbf{x}^T(k)\right] \\ &\leq p \times n < p^2, \end{aligned} \tag{14}$$

where the last inequality follows from the fact that the system is undersampled or $n < p$. Therefore, $\mathcal{Q}(k)$ is rank-deficit for all k whenever the number of measurements n is less than the number of genes p. Thus, the system is unobservable.

IV. The Constrained and smoothed Kalman Filter

A. Constraining the filter

Gene regulatory networks are known to be sparse: each gene is governed by only a small number of the genes in the network. The recovery of sparse networks is an NP-hard combinatorial problem. Instead, the problem can be relaxed by resorting to a convex l_1 minimization. We, subsequently, constrain the estimated state in (8, 9) to be sparse by bounding its l_1-norm. Intuitively, the sparsity constraint compresses the state vector by introducing zero entries; thus reducing the amount of required observations and making the system observable. Without such a constraint, the system is unobservable and any effort to reconstruct the time-varying network from undersampled measurements will be ineffective. We derive the constrained state estimate using the projection method. We compute the standard unconstrained estimate $\hat{a}$ and project it onto the constraint space. This can be formulated as

$$\tilde{\mathbf{a}} = \underset{\tilde{\mathbf{a}}}{\operatorname{argmin}}(\tilde{\mathbf{a}} - \hat{\mathbf{a}})^T W(\tilde{\mathbf{a}} - \hat{\mathbf{a}}) \text{ such that } \|\tilde{\mathbf{a}}\|_1 \leq \gamma, \quad (15)$$

where W is a positive definite weighting matrix, and γ is the sparsity inducing parameter. The optimization problem in (15) may be conveniently expressed in the following form

$$\tilde{\mathbf{a}} = \underset{\tilde{\mathbf{a}}}{\operatorname{argmin}}(\tilde{\mathbf{a}} - \hat{\mathbf{a}})^T W(\tilde{\mathbf{a}} - \hat{\mathbf{a}}) + \lambda\|\tilde{\mathbf{a}}\|_1. \quad (16)$$

The problem in (16) is a Second Order Cone Programming (SOCP) problem, and can be efficiently solved using a myriad of existing convex optimization methods. The parameter $\lambda \geq 0$ controls the amount of compression that is applied to the estimate and can be estimated by the cross-validation or the generalized cross-validation methods.

B. Forward-backward smoothing

The Kalman filter is a causal filter. It takes into account only the past and current observations in order to obtain an estimate of the state vector at the current time. The Kalman filter equations for the state-space model in (8, 9) are given by

$$\begin{aligned}
&\text{[Prediction]}\\
&\mathbf{a}_{k|k-1} = \mathbf{a}_{k-1|k-1},\\
&V_{k|k-1} = V_{k-1|k-1} + Q_k. && (17)\\
&\text{[Filtering]}\\
&K_k = V_{k|k-1}H_k^T(H_k V_{k|k-1} H_k^T + R_k)^{-1},\\
&\mathbf{a}_{k|k} = \mathbf{a}_{k|k-1} + K_k(\mathbf{y}_k - H_k \mathbf{a}_{k|k-1}),\\
&V_{k|k} = (I - K_k H_k)V_{k|k-1}. && (18)
\end{aligned}$$

Here, K_k is the Kalman gain and $V_{k|.}$ is the state estimation error covariance matrix.

We would like to use all of the available data for each state estimate, thus improving the estimation accuracy. We propose to smooth the Kalman filter by working backwards from $k = n$ to obtain the optimal estimate in the light of the whole sample. Specifically, we obtain two estimates for the forward and backward runs of the filter. The first estimate is based on the standard Kalman filter that operates from $i = 1$ to $i = k$ as described in Eqs. (17)-(18). The second estimate is based on a filter than runs backward in time from $i = n$ back to $i = k$. The forward-backward approach to smoothing combines the two estimates to form an optimal smoothed estimate. Smoothing will provide $\mathbf{a}_{k|n}$, $k = n-1, \ldots, 1$, according to the following equations

$$\begin{aligned}
\Phi_k &= V_{k|k}V_{k+1|k}^{-1},\\
\mathbf{a}_{k|n} &= \mathbf{a}_{k|k} + \Phi_k(\mathbf{a}_{k+1|n} - \mathbf{a}_{k+1|k}),\\
V_{k|n} &= V_{k|k} + \Phi_k(V_{k+1|n} - V_{k+1|k})\Phi_k^T. \quad (19)
\end{aligned}$$

C. The constrained and smoothed Kalman filter

The constrained and smoothed Kalman filter algorithm is summarized below.

1) Initialize the state vector $\mathbf{a}_{\mathbf{0}|\mathbf{0}} = \hat{\mathbf{a}}$ and state estimation error covariance $V_{0|0} = 0$.
2) For $k = 1, \cdots, n$
 - Update the matrix H_k using the vector $\mathbf{x}_k$.
 - Compute the state estimate at time k using the standard Kalman filter equations as described in (17) and (18).
 - Project the estimated state onto a sparse space by solving the convex optimization problem in (16) using, for instance, the cvx package [4].
3) Smooth the state estimate $\mathbf{a}_{k|n}$ using (19).

V. Simulation Results

We generate synthetic GRNs with different sizes (number of genes), varying degrees of sparsity and noise level. We simulate a smooth evolution of the network over time by including elimination of edges, birth of new edges and changes of the strength of interactions between the nodes. We perform Monte Carlo simulations in order to obtain statistically reliable results. To assess the efficiency of the algorithm, we use the performance measure in [5]. We present results for four distinct cases: (i) unconstrained Kalman estimation; (ii) constrained causal Kalman estimation; (iii) constrained and smoothed Kalman estimation; and (iv) constrained and smoothed Kalman estimation with an initial estimate of the state vector given by the time-invariant network estimation algorithm described in [5].

Figure 1 shows the error plots as a function of the number of measurements for the four cases. The underlying network topology and parameters are changing at each time instant. The constrained Kalman filter outperforms the unconstrained filter and converges to the optimal estimate when the number of measurements exceeds the number of nodes. The constrained and smoothed Kalman filter leads to the smallest error, particularly when the number of measurements is much smaller than the number of nodes. The estimated time-varying network, with a number of measurements smaller than the number of

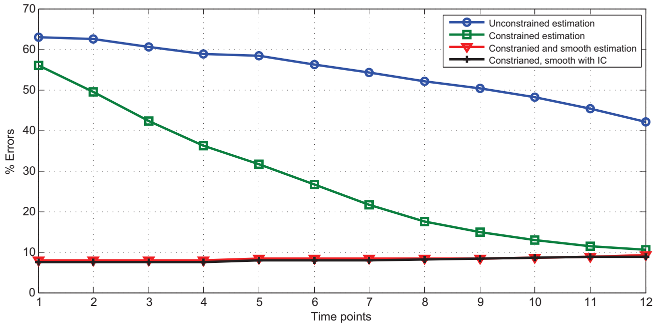


Fig. 1. Percentage error for a 10-gene network as a function of the number of measurements: unconstrained Kalman filtering in blue, constrained Kalman filtering in green, constrained and smoothed Kalman with a random initial estimate in red, and constrained and smoothed Kalman filtering with an initial state estimate given by [5] in black.

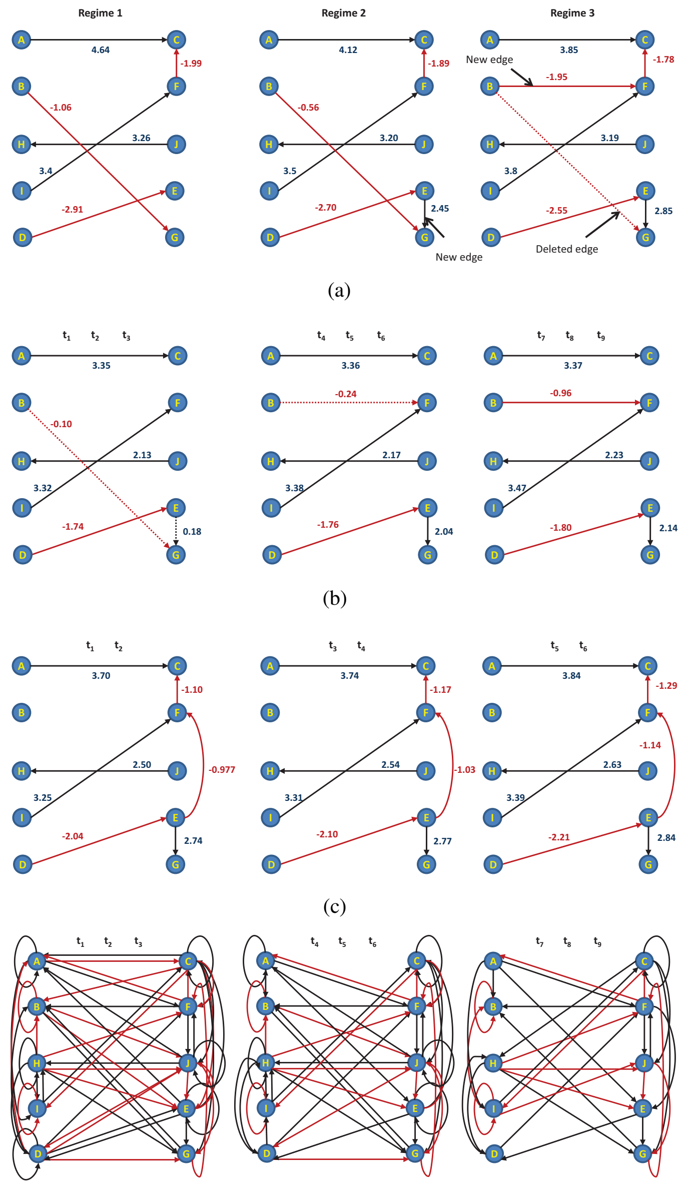
genes, is shown in Fig. 2. Figure 2(a) shows the initial time-varying network, which has three "regimes" over time. In the second regime of the network, a new stimulative edge (E→G) appears. In the third regime, the inhibitory edge (B→G) disappears and a new inhibitory edge (B→F) appears. Figures 2(b) (2(c)) shows the constrained and smoothed Kalman estimate of the network with 9 (resp., 6) measurements available, each phase of the network having 3 (resp., 2) measurements. Figure 2(d) is the estimation result using the standard (causal and unconstrained) Kalman filter with 9 measurements.

VI. Conclusion

We proposed a constrained and smoothed Kalman filter algorithm to estimate time-varying networks from undersampled data, when the number of measurements is smaller than the number of nodes. We showed that the under-determined system of time-varying gene profiles is unobservable. Thus the time-varying network parameters cannot be tracked using the standard Kalman filter. We showed that this problem can be circumvented if compression and tracking are carried out simultaneously, thereby reducing the amount of required observations. We have also adjusted the constrained (sparse) Kalman estimate to obtain an optimal estimate in the light of the whole data sample by smoothing. The proposed constrained and smoothed Kalman framework for time-varying network inference can be easily extended to the nonlinear case by considering a non-linear ODE in the state equation. In the non-linear case, the constrained and smoothed extended Kalman filter (EKF) or unscented Kalman filter (UKF) can be used. Our simulation results show that the constrained and smoothed Kalman filter yields good tracking of small and medium-size time-varying networks with undersampled data, compared to the unconstrained or unsmoothed versions of the filter.

References

[1] A. Rao, A. O. Hero, D. J. States, and J. D. Engel, "Inferring time-varying network topologies from gene expression data," *EURASIP Journal on Bioinformatics and Systems Biology*, vol. 2007, pp. 7–19, 2007.



(a)

(b)

(c)

(d)

Fig. 2. Kalman inference of the time-varying network: (a) the original 10-gene time-varying network having three regimes; (b) the constrained and smoothed Kalman estimate for 9 measurements, each phase of the network having 3 measurements (dotted lines show very weak interactions); (c) the constrained and smoothed Kalman estimate for 6 measurements, each phase of the network having 2 measurements; (d) standard Kalman filtering with 9 measurements, each phase of the network having 3 measurements.

[2] M. Kolar, L. Song, A. Ahmed, and E. P. Xing, "Estimating time-varying networks," *The Annals of Applied Statistics*, vol. 4, no. 1, pp. 94–123, 2010.

[3] E. E. Kuruoğlu, X. Yang, Y. Xu, and T. S. Huang, "Time varying dynamic Bayesian network for nonstationary events modeling and online inference," *IEEE Transactions on Signal Processing*, vol. 59, no. 4, pp. 1553 – 1568, April 2011.

[4] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 1.21," Apr. 2011.

[5] G. Rasool, N. Bouaynaya, H. Fathallah-Shaykh, and D. Schonfeld, "Inference of genetic regulatory networks using regularized likelihood with covariance estimation," in *IEEE Statistical Signal Processing Workshop*, Ann Arbor, Michigan, USA, 2012, pp. 560–563.