NETWORK RECONSTRUCTION FROM TIME-COURSE PERTURBATION DATA USING MULTIVARIATE GAUSSIAN PROCESSES

Sara Al-Sayed[⋆] *and Heinz Koeppl*[⋆†]

[⋆]Department of Electrical Engineering and [†]Department of Biology,
Technische Universität Darmstadt, Germany

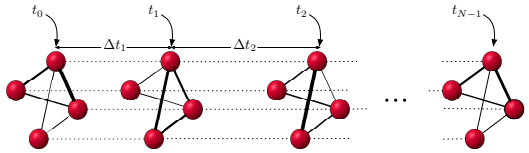

Fig. 1. Snapshots of a network at discrete, non-uniformly–spaced observation times.

ABSTRACT

In this work, we appropriate the popular tool of Gaussian processes to solve the problem of reconstructing networks from time-series perturbation data. To this end, we propose a construction for multivariate Gaussian processes to describe the continuous-time trajectories of the states of the network entities. We then show that this construction admits a state-space representation for the network dynamics. By exploiting Kalman filtering techniques, we are able to infer the underlying network in a computationally efficient manner.

Index Terms— Network reconstruction, multivariate Gaussian processes, state-space representation, time-course data.

1. INTRODUCTION

Time-course data observed under the perturbation of, for example, biological systems contain rich information about the salient structure of interconnectivity among the entities of the network underlying the system. Consider, for example, the undirected network depicted in Fig. 1. Temporal snapshots at a number of time points are shown. These correspond to observation times of the high-dimensional measurements at non-uniformly–spaced intervals. The edges linking the nodes of the network are seen to vary over time in intensity, as captured by their widths. The network connectivity pattern, however, remains unchanged. It is this connectivity pattern that a network reconstruction algorithm, like the one developed in this work, attempts to reconstruct given noisy time-course observations of the states of the nodes. After all, network reconstruction approaches that exploit the temporal dependency of the high-dimensional observational data tend to outperform those hinging on the assumption of temporal independence [1, 2, 3, 4, 5, 6, 7]. The challenges encountered include the availability of only few noisy high-dimensional measurements at non-uniformly–spaced intervals; missing data; and the computational complexity of inference, parameter estimation, and structure search.

This work was funded by the European Union's Horizon 2020 research and innovation programme under grant agreement 668858.

For the algorithm developed in this work, the discrete, noisy time-course observational data are modeled as temporal snapshots of realizations of a multivariate Gaussian process (GP). This amounts to specifying a multivariate GP prior over the temporal trajectories describing the evolution of the nodes in the network. GPs are a popular tool for regression analysis and prediction using time-series data [8]. Multivariate GPs have sparked interest recently, as they enable capturing the dependencies in high-dimensional data. The specification of the multivariate process, however, has largely been dictated by the target application. Though the framework of GP regression has been exploited for the prediction of regulator states from target observations in regulatory networks [9], to our knowledge it has not been used so far for network reconstruction. Moreover, devising efficient computational procedures for multivariate GP regression remains a challenge [10, 11].

Special choices of the covariance kernel of the multivariate GP give rise to processes that can be described by a system of coupled stochastic linear differential equations where the network structure and coupling weights are unknown. Exploiting the state-space representation of this system [12, 13], computationally efficient Kalman filtering techniques are used in this work to score candidate network structures given the observed data, in terms of the *a posteriori* candidate probabilities. The calculation of these scores, however, requires the estimation of the coupling weights. Those can be computed according to maximum *a posteriori* or maximum-likelihood criteria by a projected steepest-descent procedure that avails itself of the Kalman filtering recursions [14]. Hence, the proposed estimation approach is computationally efficient. Another advantage of the outlined representation is that it can naturally accommodate missing temporal measurements.

This paper is organized as follows. In Sec. 2, the basic data model is introduced, the network reconstruction problem is formulated, and the motivation for our approach is recapitulated. The approach is developed in Sec. 3 and its performance illustrated in Sec. 4. A discussion follows in Sec. 5

2. PRELIMINARIES

Consider an undirected network of P nodes described by a *candidate* adjacency matrix A of size $P \times P$, with real-valued trajectories $x_1(t), \ldots, x_P(t)$ that are realizations of a P-dimensional multivariate wide-sense stationary random process. The trajectories, indexed by $p = 1, \ldots, P$, are observed at N time points t_i, $i = 0, \ldots, N-1$, subject to measurement noise:

$$y_{p,i} = x_p(t_i) + \epsilon_{p,i}. \tag{1}$$

The noise processes $\{\epsilon_{p,i}\}$ are spatially and temporally independent and identically distributed (i.i.d.), with $\boldsymbol{\epsilon}_i \triangleq [\epsilon_{1,i}, \ldots, \epsilon_{P,i}]^\top \sim$

978-1-5386-5477-4/18/$31.00

$\mathcal{N}(0_P, \sigma_\epsilon^2 I_P)$, where 0_P and I_P are the all-zero vector of size $P \times 1$ and identity matrix of size $P \times P$, respectively. Let $\boldsymbol{x}(t) \triangleq [x_1(t), \ldots, x_P(t)]^\top$. In GP regression, a joint Gaussian prior is specified over $\{\boldsymbol{x}(t_i)\}$:

$$\boldsymbol{x}(t_0), \ldots, \boldsymbol{x}(t_{N-1}) \sim \mathcal{N}(0_{NP}, C_A) \tag{2}$$

where the covariance matrix C_A is an $N \times N$ block matrix with blocks $C_{A,ij} = \mathbb{E}\left[\boldsymbol{x}(t_i)\boldsymbol{x}^\top(t_j)\right]$ of size $P \times P$, $i, j = 0, \ldots, N-1$. The matrix C_A should ideally reflect the underlying candidate network structure as captured by the adjacency matrix A. Let $\boldsymbol{y}_i \triangleq [y_{1,i}, \ldots, y_{P,i}]^\top$. The *a posteriori* probability of the candidate adjacency matrix A given the measurements $\boldsymbol{y}_{0:N-1} \triangleq \{\boldsymbol{y}_i, i = 0, \ldots, N-1\}$, $p\big(A \mid \boldsymbol{y}_{0:N-1}\big)$, assuming all adjacency matrices to be equally probable, is proportional to the conditional likelihood of the measurements given the adjacency matrix, the *conditional data likelihood*:

$$p\big(\boldsymbol{y}_{0:N-1} \mid A\big) = \Pi_{i=0}^{N-1} \mathcal{N}\big(\boldsymbol{y}_i; \widehat{\boldsymbol{x}}_{A,i|i-1}, P_{A,i|i-1} + \sigma_\varepsilon^2 I_P\big) \tag{3}$$

where $\widehat{\boldsymbol{x}}_{A,i|i-1}$ and $P_{A,i|i-1}$ are filtering estimates, given the structure candidate A, of $\boldsymbol{x}(t_i)$ given measurements up to t_{i-1} and the corresponding estimator covariance, which coincide with the minimum mean-square-error estimate and mean-square-error estimator covariance, respectively. In Bayesian model inference, the conditional data likelihood (3) is used as a means to score candidate models, the adjacency matrices in this work. However, the calculation of the necessary quantities $\widehat{\boldsymbol{x}}_{A,i|i-1}$ and $P_{A,i|i-1}$ involves the inversion of matrices of sizes $(i-1)P \times (i-1)P$, $i = 2, \ldots, N-1$. Moreover, there is a lack of insight as to how best to design the covariance matrix C_A. In this work, we tackle the two problems of computational complexity and design ambiguity jointly, in that we devise a covariance kernel for the multivariate GP that allows for the description of the process by a system of coupled stochastic linear differential equations. The state-space representation of this system lends itself to the application of computationally efficient Kalman filtering techniques for the calculation of the quantities in question. The covariance kernel design proposed here is itself informed by the system description, as we demonstrate in the next section.

3. APPROACH

3.1. Linear Time-invariant (LTI) Stochastic Differential Equation Model

Now, assume the multivariate GP $\boldsymbol{x}(t)$ given a candidate adjacency matrix A to be described by the following nth-order stochastic differential equation:

$$\frac{d^n\boldsymbol{x}(t)}{dt^n} + B_{n-1}\frac{d^{n-1}\boldsymbol{x}(t)}{dt^{n-1}} + \ldots + B_1\frac{d\boldsymbol{x}(t)}{dt} + B_0\boldsymbol{x}(t) = \boldsymbol{z}(t) \tag{4}$$

where $\boldsymbol{z}(t) \triangleq [z_1(t), \ldots, z_P(t)]^\top$ is a vector of mutually independent zero-mean white Gaussian noise processes with powers $\sigma_{z,p}^2$, $p = 1, \ldots, P$; and $B_{n'} = [b_{n'}^{(k\ell)}]$, $n' = 0, 1, \ldots, n-1$, $k, \ell = 1, \ldots, P$, are coupling matrices. It is assumed for simplicity that the nonzero pattern of the matrices $B_{n'}$ for all n' is the same as that of the matrix $A + I_P$. The coupling coefficients $b_{n'}^{(k\ell)}$ are to be specified in Sec. 3.2, where it will be shown that an intuitively motivated specification of these coupling coefficients translates to a reasonable design for the covariance matrix C_A in (2) on the one hand, and an efficient computation of the filtering estimates towards the calculation of the structure candidate scores on the other hand by exploiting the model (4). In this manner, the aforementioned twofold problem of design ambiguity and computational complexity is adequately tackled.

The state-space representation of the model (4) is given by

$$\frac{d\widetilde{\boldsymbol{x}}(t)}{dt} = \widetilde{B}\widetilde{\boldsymbol{x}}(t) + L\boldsymbol{z}(t), \tag{5}$$

a first-order Markov process, where

$$\begin{aligned}
\widetilde{\boldsymbol{x}}(t) &\triangleq [\widetilde{\boldsymbol{x}}_1(t), \ldots, \widetilde{\boldsymbol{x}}_P(t)]^\top \\
\widetilde{\boldsymbol{x}}_p(t) &\triangleq \left[x_p(t), \frac{dx_p(t)}{dt}, \ldots, \frac{d^{n-1}x_p(t)}{dt^{n-1}}\right]^\top, \quad p = 1, \ldots, P \\
L &\triangleq \mathrm{col}\{L_1, \ldots, L_P\}
\end{aligned}$$

where the operator $\mathrm{col}\,\{\cdot\}$ stacks its arguments vertically, with

$$L_p = \begin{bmatrix} 0_{(n-1)\times P} \\ \boldsymbol{e}_{P,p} \end{bmatrix}$$

for $p = 1, \ldots, P$, where $0_{c\times d}$ is the all-zero matrix of size $c \times d$ and $\boldsymbol{e}_{P,p}$ is the pth canonical row basis vector of length P. The matrix $\widetilde{B}$ is a $P \times P$ block matrix with blocks $\widetilde{B}_{k\ell}$ of size $n \times n$ given by

$$\widetilde{B}_{k\ell} = \begin{cases} \begin{bmatrix} 0_{(n-1)\times n} \\ -\widetilde{\boldsymbol{b}}^{(k\ell)\top} \end{bmatrix}, & \text{if} \quad \ell \neq k \\ \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 1 \\ & & -\widetilde{\boldsymbol{b}}^{(kk)\top} & & \end{bmatrix}, & \text{if} \quad \ell = k \end{cases} \tag{6}$$

where $\widetilde{\boldsymbol{b}}^{(k\ell)} \triangleq [b_0^{(k\ell)}, b_1^{(k\ell)}, \ldots, b_{n-1}^{(k\ell)}]^\top$. The state-space representation will prove handy in the derivation of the discrete-time state-space model in Sec. 3.4. The latter model forms the basis for the application of computationally efficient discrete-time Kalman filtering techniques towards the calculation of the filtering estimates for the subsequent computation of the structure candidate scores.

3.2. Multivariate GP Specification

The LTI system represented by (4) is depicted in Fig. 2 in terms of frequency responses $\widehat{h}_{pq}(j\omega)$, $p, q = 1, \ldots, P$, that are given by the entries of the matrix

$$\widehat{H}(j\omega) = \left[(j\omega)^n I_P + (j\omega)^{n-1} B_{n-1} + \ldots + B_0\right]^{-1}. \tag{7}$$

Furthermore, it is assumed that the system is stable, i.e., the eigenvalues of $\widetilde{B}$ have strictly negative real parts. The auto- and cross-power spectral densities (PSDs) of the network nodes are given, respectively, by

$$s_{x_k}(j\omega) = \sum_{m=1}^{P} \sigma_{z,m}^2 \left|\widehat{h}_{km}(j\omega)\right|^2 \tag{8a}$$

$$s_{x_k x_\ell}(j\omega) = \sum_{m=1}^{P} \sigma_{z,m}^2 \widehat{h}_{km}(j\omega)\, \widehat{h}_{\ell m}^*(j\omega) \tag{8b}$$

for each node k and pair of nodes $k, \ell = 1, \ldots, P$. The corresponding auto- and cross-covariance functions are given by the Fourier transform of the auto- and cross-PSDs, respectively. It is then clear

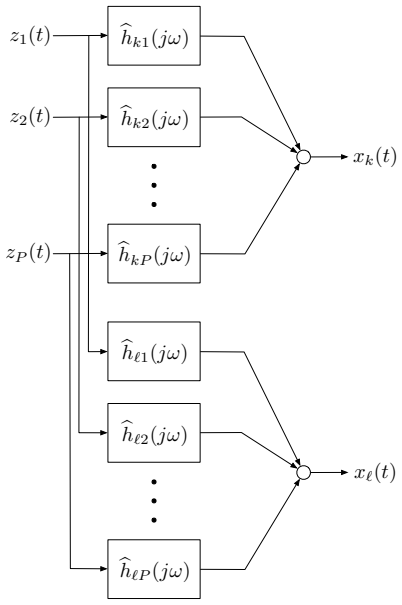

Fig. 2. LTI system represented by (4).

that the multivariate GP is completely specified by the coupling coefficients $b_{n'}^{(k\ell)}$. In the following, we propose a specification inspired by a univariate GP analog, where the nodes are decoupled. Note that the univariate GP framework [12] is recovered when $A = I_P$; and the matrices $B_{n'}$ and $\widehat{H}(j\omega)$ are diagonal, $n' = 0, 1, \ldots, n-1$, so that $\widehat{h}_{k\ell}(j\omega) = 0$, if $\ell \neq k$. For decoupled nodes, consider the following design for the frequency responses based on the Matérn covariance function [8, 12] ($b_n^{(kk)} \equiv 1 \; \forall k$):

$$\widehat{h}_{k\ell}(j\omega) = \begin{cases} \frac{1}{\sum_{n'=0}^{n} b_{n'}^{(kk)} (j\omega)^{n'}} \stackrel{!}{=} \frac{1}{(\lambda + j\omega)^n} & , \quad \ell = k \\ 0 & , \quad \text{o.w.} \end{cases} \tag{9}$$

where $\lambda \triangleq \frac{\sqrt{2\nu}}{l}$, with $\nu \triangleq n - \frac{1}{2}$ being the smoothness parameter and l, the length-scale parameter. Motivated by (9), consider the following design for the frequency-response matrix of the multivariate GP (7):

$$\widehat{H}(j\omega) \stackrel{!}{=} \left(\widehat{D}^o(j\omega) \odot (A + I_P) \right)^{-1} \tag{10}$$

where the entries of the $P \times P$ matrix $\widehat{D}^o(j\omega)$ are given by

$$\widehat{d}^o_{k\ell}(j\omega) = \begin{cases} (\lambda + j\omega)^n, & \text{if} \quad \ell = k \\ (\lambda + j\omega)^{n-1}, & \text{if} \quad \ell \neq k \end{cases} \tag{11}$$

and $\odot$ denotes the Hadamard (entry-wise) product. The coupling coefficients $b_{n'}^{(k\ell)}$ can be read off directly given the frequency-response matrix (7) and the design equations (10) and (11). Necessarily, $\sigma^2_{z,p} \equiv \sigma^2_z = \frac{2\sigma^2 \sqrt{\pi} \lambda^{2n-1} \Gamma(n)}{P \Gamma(\nu)}$ for all p, where σ^2 is the magnitude parameter of the Matérn covariance function. We illustrate the proposed design using an example:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{bmatrix}, \quad n = 2, \; \lambda = 5, \; \sigma^2 = 1. \tag{12}$$

The resulting covariance functions are plotted in Fig. 3, revealing that disconnected node pairs in the network described by the adjacency matrix in (12) have smaller cross-covariance than those that are connected. A similar pattern is observed with the resulting PSDs. This pattern is to be expected, since the zero pattern imposed on the frequency-response matrix in (10) induces conditional independence relations among the corresponding node pairs. Conditional independence relations, however, do not imply marginal independence relations, which the covariance functions are meant to capture [15].

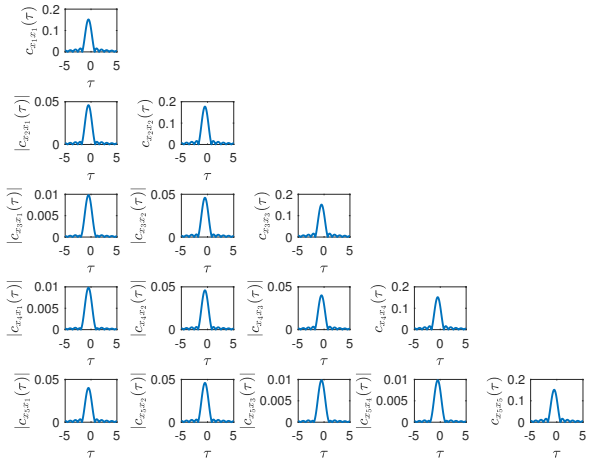

Fig. 3. Auto- and cross-covariance functions for the example in (12).

3.3. Weighted Interactions

Thus far, the coupling coefficients $b_{n'}^{(k\ell)}$, which essentially determine the strength of the interactions among the network nodes, are dictated by the design equations (10) and (11). In order to render the model at hand more flexible while maintaining a relatively low complexity overhead, we propose a simple modification to the model to account for weighted interactions. To this end, consider the $P \times P$ weights matrix $\Omega = [w_{k\ell}]$ that is positive definite, with entries $w_{k\ell} = 1$ if $\ell = k$, and $w_{k\ell} = 0$ if the nodes k and ℓ are disconnected. The corresponding trajectories can then be described by

$$\frac{d^n \boldsymbol{x}(t)}{dt^n} + \Omega \odot \sum_{r=0}^{n-1} B_r \frac{d^r \boldsymbol{x}(t)}{dt^r} = \boldsymbol{z}(t). \tag{13}$$

The corresponding modification to (5) follows by replacing $\widetilde{B}$ with $\widetilde{B}_\Omega \triangleq \widetilde{B} \odot (\Omega \otimes \mathbb{1}_{n \times n})$, where $\mathbb{1}_{c \times d}$ is the all-ones matrix of size $c \times d$. It is now assumed that the eigenvalues of $\widetilde{B}_\Omega$ have strictly negative real parts such that the resulting system is stable.

3.4. Discrete-time State-space Model

For measurement times t_i, $i = 0, \ldots, N-1$, $\boldsymbol{x}_i \equiv \widetilde{\boldsymbol{x}}(t_i)$, the discrete-time process equation is given by

$$\boldsymbol{x}_{i+1} = F_i \boldsymbol{x}_i + \boldsymbol{q}_i \tag{14}$$

where

$$F_i = \exp\Big(\widetilde{B}_\Omega \Delta t_{i+1}\Big), \quad \Delta t_{i+1} \triangleq t_{i+1} - t_i, \tag{15}$$

the process noise $\{\boldsymbol{q}_i\}$ is a white Gaussian process, with $\boldsymbol{q}_i \sim \mathcal{N}(0_{NP}, Q_i)$, where

$$Q_i = \int_0^{\Delta t_{i+1}} \exp\Big[\widetilde{B}_\Omega(\Delta t_{i+1} - \tau)\Big] L \Sigma_z L^\top \exp\Big[\widetilde{B}_\Omega(\Delta t_{i+1} - \tau)\Big]^\top d\tau \tag{16}$$

with $\Sigma_z = \sigma_z^2 I_{NP}$. Essentially, $\boldsymbol{x}_0 \sim \mathcal{N}(0_{NP}, \Pi_\infty)$, where Π_∞ solves the continuous Lyapunov equation of the continuous-time process model. The measurement equation is given by

$$\boldsymbol{y}_i = H\boldsymbol{x}_i + \boldsymbol{\varepsilon}_i \tag{17}$$

where H selects $\{x_p(t_i)\}$ from $\widetilde{\boldsymbol{x}}(t_i)$, and $\{\boldsymbol{\varepsilon}_i\}$ are i.i.d. measurement noises, independent of the process and process noise, with $\boldsymbol{\varepsilon}_i \sim \mathcal{N}(0_P, \sigma_\varepsilon^2 I_P)$.

3.5. Structure Scoring

Structure candidates are scored according to their *a posteriori* probability in terms of the conditional data likelihood:

$$p\big(\boldsymbol{y}_{0:N-1} \mid A, \widehat{\Theta}_A\big) = \Pi_{i=0}^{N-1} \mathcal{N}\big(\boldsymbol{y}_i; H\widehat{\boldsymbol{x}}_{A,i|i-1}^{(\widehat{\Theta}_A)}, HP_{A,i|i-1}^{(\widehat{\Theta}_A)} H^\top + \sigma_\varepsilon^2 I_P\big) \tag{18}$$

where $\widehat{\boldsymbol{x}}_{A,i|i-1}^{(\widehat{\Theta}_A)}$ and $P_{A,i|i-1}^{(\widehat{\Theta}_A)}$ are the discrete-time Kalman filter *a posteriori* ith state mean and covariance matrices, respectively, given the structure candidate A and the estimated (hyper)parameters $\widehat{\Theta}_A$, under the discrete-time state-space model in Sec. 3.4. Unknown (hyper)parameters Θ_A (interaction weights, noise variances, and covariance function parameters) can be estimated jointly with the Kalman filter procedure in either a maximum-likelihood (ML) or a maximum *a posteriori* (MAP) sense [14]. Although this approach marks a departure from a fully Bayesian inference framework, where unknown (hyper)parameters are averaged out (cf. [16]), it is preferred when a low computational complexity needs to be maintained. Moreover, by resorting to ML estimation of the (hyper)parameters, the need to specify priors is obviated. For this reason, we restrict our attention henceforth to ML estimation. Let the ML estimates be denoted by $\widehat{\Theta}_A^{\mathrm{ML}}$, and the corresponding conditional data log-likelihood score by

$$\mathrm{score}_{\mathrm{L}}(A) = \ln p\big(\boldsymbol{y}_{0:N-1} \mid A, \widehat{\Theta}_A^{\mathrm{ML}}\big) \tag{19}$$

where L in the subscript is for likelihood. Another note is in order: The likelihood score (19) tends to prefer dense networks to sparse ones [15]. It is therefore justified to add a penalty term. This leads to the following Bayesian information criterion (BIC) score [15]:

$$\mathrm{score}_{\mathrm{BIC}}(A) = \ln p\big(\boldsymbol{y}_{0:N-1} \mid A, \widehat{\Theta}_A^{\mathrm{ML}}\big) - \frac{N_{\Theta_A}}{2} \ln N \tag{20}$$

where N_{Θ_A} is the number of ML-estimated parameters.

3.6. Structure Search and Prior Network Information

Greedy (local hill-climbing) structure search algorithm variants can be used to find a locally optimal structure when an exhaustive search over all possible structure candidates proves computationally prohibitive [15]. In greedy search, starting from some initial structure, one edge at a time is modified by insertion or deletion as long as its modification leads to a structure with better score. However, greedy approaches often converge to structures that are only locally optimal. In order to circumvent this, several greedy search instances may be launched starting with different initial structures and the resulting locally optimal structures combined in some manner. The selection of the initial structures may benefit from some prior network knowledge, derived, in biological network reconstruction applications, for example, from biological databases [17].

4. PERFORMANCE ILLUSTRATION

First, in order to illustrate the average performance of our network reconstruction algorithm, we chose a small enough network size of $P = 4$ nodes $\in \{1, 2, 3, 4\}$ with two possible edge sets, $\mathcal{E}_1 = \{(1,2), (1,4), (2,3), (2,4)\}$ and $\mathcal{E}_2 = \{(1,2), (3,4)\}$, leading to the two network structures denoted as $\mathcal{N}_1$ and $\mathcal{N}_2$, and second-order ($n = 2$) stochastic differential equations such that an exhaustive search over all possible network structures is admissible by the system at hand. All our experiments were conducted using MATLAB 2016b running on a standard desktop computer with 2.7 GHz CPU and 8 GB RAM. The parameters λ and σ^2 were set to 3 and 1, respectively. The off-diagonal weights $w_{k\ell}$, $\ell \neq k$, were selected randomly and independently over $[-10, 10)$, then projected such that Ω, with ones along the diagonal, is positive definite. The measurements $\{\boldsymbol{y}_i\}$ were taken every time-unit, here hours, with measurement noise variance $\sigma_\varepsilon^2 = 10^{-1}$—see Figs. 4 and 5 for sample realizations of ten-hour trajectories arising from $\mathcal{N}_1$ and $\mathcal{N}_1$, respectively, with superimposed noisy measurements. We estimated the off-diagonal weights in an ML fashion, performing gradient-based optimization of the energy function by means of the Kalman filter sensitivity equations [14, *Appendix* 3]. The partial derivatives of F_i with respect to the individual weights were computed using the result by Wilcox for the differentiation of the matrix exponential function [18], while linearization was utilized to compute the partial derivatives of Q_i. For optimization we employed a projected steepest-descent method to ensure the estimated weights matrix Ω is positive definite [19]. The adaptive Barzilai–Borwein [20] step-size construction was used in the steepest-descent method; and the method was terminated when the Euclidean norm of the difference between successive iterates fell below 10^{-3}—it was observed that the number of iterations in this experiment never exceeded ten. CVX was called to solve the projection problem [21]. For each of the true underlying network structures $\mathcal{N}_1$ and $\mathcal{N}_2$, for a given number of measurements (10, 50, 100, 500, 1,000, 5,000, or 10,000), each of the 64 possible network structures were scored according to the likelihood and BIC scores in (19) and (20). The two scores of each possible structure for a given true underlying structure and a given number of measurements were then averaged each over 40 Monte Carlo runs. Then, the two sets of scores for all possible structures for a given true underlying structure and a given number of measurements were ordered descendingly and the respective rank of the true structure was computed. In Fig. 6, the resulting ranks of the true underlying structures arising from the likelihood and BIC scores are plotted with respect to the number of available measurements, a rank of one being the best achievable. The average computational run time was around 2 minutes for 10 measurements, and rose to around 13 minutes for 10,000 measurements.

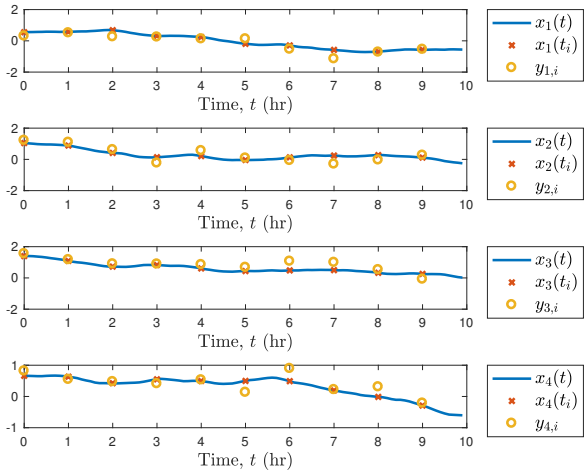

Fig. 4. Sample realization of ten-hour trajectories of the four nodes of the network corresponding to structure $\mathcal{N}_1$ with superimposed sequence of noisy measurements.

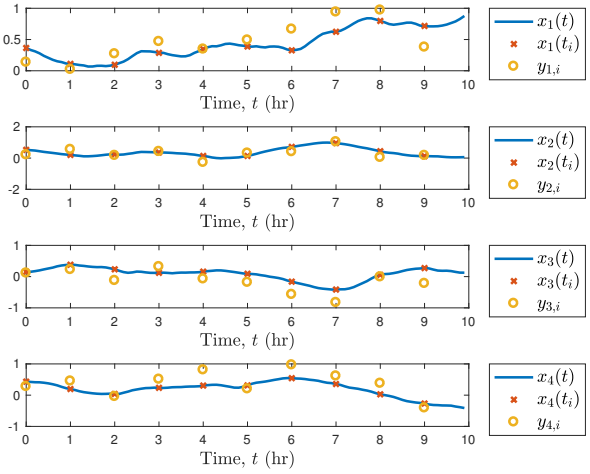

Fig. 5. Sample realization of ten-hour trajectories of the four nodes of the network corresponding to structure $\mathcal{N}_2$ with superimposed sequence of noisy measurements.

5. DISCUSSION

Hinging on computationally efficient Kalman filtering techniques, the proposed approach scales favorably with network size. Compared with classical inference employing matrix inversion of computational complexity $\mathcal{O}(N^3P^3)$, the Kalman filtering approach enjoys a computational complexity $\mathcal{O}(NP)$ for small stochastic differential equation order n relative to the number of time points N and network size P. An obvious bottleneck towards the application of the approach for the reconstruction of high-dimensional networks remains to be the graph enumeration procedure underlying all scoring-based methods.

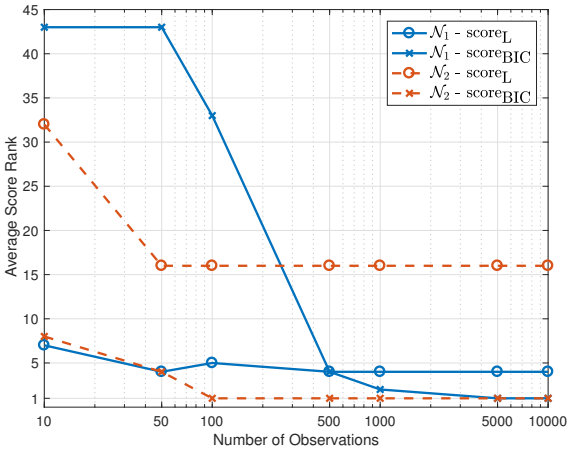

Fig. 6. Rank achieved by true network structures $\mathcal{N}_1$ (solid, blue) and $\mathcal{N}_2$ (dashed, red) according to conditional data log-likelihood score (circles) and BIC score (crosses) relative to the number of available measurements.

6. REFERENCES

[1] C. J. Oates and S. Mukherjee, "Network inference and biological dynamics," *The Annals of Applied Statistics*, vol. 6, no. 3, pp. 1209, 2012.

[2] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432–441, 2008.

[3] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "A sparse-group lasso," *Journal of Computational and Graphical Statistics*, vol. 22, no. 2, pp. 231–245, 2013.

[4] N. Friedman, K. Murphy, and S. Russell, "Learning the structure of dynamic probabilistic networks," in *Proc. Conference on Uncertainty in Artificial Intelligence*, Madison, WI, USA, July 1998, pp. 139–147.

[5] U. Nodelman, C. R. Shelton, and D. Koller, "Continuous time Bayesian networks," in *Proc. Conference on Uncertainty in Artificial Intelligence*, Alberta, Canada, Aug. 2002, pp. 378–387.

[6] U. Nodelman, C. R. Shelton, and D. Koller, "Learning continuous time Bayesian networks," in *Proc. Conference on Uncertainty in Artificial Intelligence*, Acapulco, Mexico, Aug. 2003, pp. 451–458.

[7] L. Studer, L. Paulevé, C. Zechner, M. Reumann, M. R. Martínez, and H. Koeppl, "Marginalized continuous time Bayesian networks for network reconstruction from incomplete observations," in *Proc. AAAI Conference on Artificial Intelligence*, Phoenix, Arizona, USA, Feb. 2016, pp. 2051–2057.

[8] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, Cambridge, Massachusetts, USA, 2006.

[9] N. D. Lawrence, G. Sanguinetti, and M. Rattray, "Modelling transcriptional regulation using Gaussian processes," in *Advances in Neural Information Processing Systems*, P. B. Schölkopf, J. C. Platt, and T. Hoffman, Eds., pp. 785–792. MIT Press, Cambridge, Massachusetts, USA, 2007.

[10] L. Csató and M. Opper, "Sparse on-line Gaussian processes," *Neural Computation*, vol. 14, no. 3, pp. 641–668, 2002.

[11] E. Gilboa, Y. Saatçi, and J. P. Cunningham, "Scaling multidimensional inference for structured Gaussian processes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 2, pp. 424–436, Feb. 2015.

[12] J. Hartikainen and S. Särkkä, "Kalman filtering and smoothing solutions to temporal Gaussian process regression models," in *IEEE International Workshop on Machine Learning for Signal Processing*, Aug. 2010, pp. 379–384.

[13] S. Särkkä, A. Solin, and J. Hartikainen, "Spatiotemporal learning via infinite-dimensional Bayesian filtering and smoothing: A look at Gaussian process regression through Kalman filtering," *IEEE Signal Processing Magazine*, vol. 30, no. 4, pp. 51–61, July 2013.

[14] S. Särkkä, *Bayesian Filtering and Smoothing*, Cambridge University Press, Cambridge, UK, 2013.

[15] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*, MIT Press, Cambridge, Massachusetts, USA, 2009.

[16] M. Girolami, "Bayesian inference for differential equations," *Theoretical Computer Science*, vol. 408, no. 1, pp. 4–16, 2008.

[17] D. Türei, T. Korcsmáros, and J. Saez-Rodriguez, "OmniPath: guidelines and gateway for literature-curated signaling pathway resources," *Nature Methods*, vol. 13, no. 12, pp. 966, 2016.

[18] R. M. Wilcox, "Exponential operators and parameter differentiation in quantum physics," *Journal of Mathematical Physics*, vol. 8, no. 4, pp. 962–982, 1967.

[19] S. Boyd and L. Vanderberghe, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.

[20] J. Barzilai and J. M. Borwein, "Two-point step size gradient methods," *IMA Journal of Numerical Analysis*, vol. 8, no. 1, pp. 141–148, 1988.

[21] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," `http://cvxr.com/cvx`, Mar. 2014.