ESTIMATION OF GENE EXPRESSION BY A BANK OF PARTICLE FILTERS

Mónica F. Bugallo, Çağla Taşdemir and Petar M. Djurić

Department of Electrical & Computer Engineering
Stony Brook University
Stony Brook, NY 11794, USA
e-mail: monica.bugallo@stonybrook.edu, petar.djuric@stonybrook.edu

ABSTRACT

This paper addresses the problem of joint estimation of time series of gene expressions and identification of the coefficients of gene interactions defining the network. The proposed method exploits a state-space structure describing the system so that a bank of particle filters can be used to efficiently track each of the time series separately. Since each gene interacts with some of the other genes, the individual filters need to exchange information about the states (genes) that they track. The analytical derivation of the posterior distribution of the states given the observed data allows for marginalization of the matrix describing the interactions in the network and for efficient implementation of the method. Computer simulations reveal a promising performance of the proposed approach when compared to the conventional particle filter that attempts to track the time series of all the genes and which, as a result, suffers from the curse-of-dimensionality.

***Index Terms*—** Gene regulatory network, particle filtering, dimensionality reduction.

1. INTRODUCTION

A gene regulatory network is a collection of genes interacting with each other and with other substances in order to govern cell functions. Inference of gene regulatory relationships is a very important problem in biology. The inference can be based on time series that represent genome expressions. Various frameworks have been proposed for identifying gene regulatory networks including Boolean networks [1, 2], neural networks [3], differential equations [4], factor graphs [5] and Bayesian networks [6, 7, 8]. Among these approaches, Bayesian dynamical networks have been particularly popular [9, 10, 11].

In a realistic scenario of a dynamic gene regulatory network, the evolution of the genes is modeled by a nonlinear function. Typically the number of genes is very high, and their evolution with time is tracked by using noisy measurements of gene expressions. A standard model for tracking is the state-space model where the states represent the various genes in the network. Not surprisingly, the state-space models have already been extensively studied and a wide range of estimation algorithms have been investigated [12, 13]. The most well-known method is the Kalman filter, which is employed when the models are linear and Gaussian [14, 15]. If the models are nonlinear, one option is the Extended Kalman Filter (EKF) [16]. While being computationally efficient, the EKF has the disadvantage that it is restricted for problems with Gaussian noise and that it has low accuracy in the case of severe nonlinearities. There are other variants of the Kalman filter. For example, in [17], a cubature Kalman filter is employed for the estimation of the gene expressions.

A successful methodology for estimation of nonlinear dynamic systems is particle filtering [18, 19]. With particle filtering, the nonlinearities are not approximated. Instead, one approximates the distributions of the system unknowns by discrete random measures. In [20], a particle filter with Markov Chain Monte Carlo move is employed to estimate the system parameters. In this work, the system dynamics are described by a deterministic model based on ordinary differential equations. In [21], the estimation is carried out by the unscented particle filter. A method that combines Kalman and particle filtering is proposed in [22]. Other approaches to the problem include the use of the Expectation Maximization algorithm [15]. A point-based Gaussian approximation filter that incorporates prior information about the gene regulatory network is presented in [23].

In this paper we address the estimation of gene expression and the interacting coefficients defining the system of evolving genes by particle filtering. Since in practice the addressed problem can be of very large dimensions, traditional particle filters (PFs) produce an approximation error that increases exponentially in the dimension of the state and suffer from what is referred to as the curse-of-dimensionality [24, 25]. We propose to apply a bank of PFs, each of them tracking one gene and its corresponding interacting coefficients. This concept has already been introduced as multiple particle filtering [26, 27]. The novelty in this paper is in the implementation of optimal particle filtering by each of the PFs. For correct functioning of the proposed approach, the filters need to communicate information related to the individual genes that they track. Each filter works on a minimal dimensional subspace and therefore the method is efficient while maintaining the performance accuracy.

The remaining of the paper is organized as follows. The next section introduces the problem formulation and the mathematical notation. In Section 3, the proposed multiple particle filtering method is explained with a detailed derivation of the optimal proposal function and specific description of the step-by-step implementation. Numerical results in Section 4 demonstrate the validity of the new approach and its improved performance over that of standard particle filtering. Finally, some concluding remarks complete the paper in Section 5.

2. PROBLEM STATEMENT

In our description of the problem, we follow [22]. Let the system of interest be comprised of N genes. The genes evolve with time and

This work was supported by NSF under Awards CCF-1320626 and CCF-0953316.

978-0-9928626-3-3/15/$31.00 ©2015 IEEE

the matrix corresponding to the gene expressions at time instant t is denoted by $X_t \in \mathbb{R}^{N\times t}$ and defined as

$$X_t = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,t} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,t} \\ \vdots & \vdots & \vdots & \vdots \\ x_{N,1} & x_{N,2} & \cdots & x_{N,t} \end{bmatrix}, \quad (1)$$

where $x_{n,j}$ is the expression of the nth gene at time instant j. For convenience, we denote the vector of all genes at time instant t by $x_t \in \mathbb{R}^{N\times 1}$, and it corresponds to the tth column vector of the matrix X_t), i.e., $x_t = [x_{1,t}\, x_{2,t}\, \cdots\, x_{N,t}]^\top$.

The evolution of the genes is described by

$$x_t = Ag_{t-1} + u_t, \quad (2)$$

where $A \in \mathbb{R}^{N\times N}$ is an unknown matrix whose rows are denoted by $a_n^\top$, i.e.,

$$A = \begin{bmatrix} a_1^\top \\ a_2^\top \\ \vdots \\ a_N^\top \end{bmatrix}, \quad (3)$$

$$a_n^\top = [a_{n,1}\; a_{n,2}\; \cdots\; a_{n,N}]. \quad (4)$$

Note that the coefficient $a_{n,m}$ indicates the regulatory relationship between gene n and gene m (positive vs. negative value signifies activation vs. repression activity, respectively). For the priors of a_n, $n = 1, 2, \cdots, N$, we assume

$$p(a_n) = \frac{1}{(2\pi)^{N/2}|\Sigma_n|^{1/2}} e^{-\frac{1}{2}a_n^\top \Sigma_n^{-1} a_n}, \quad (5)$$

where Σ_n is the $N \times N$ covariance matrix, i.e., $a_n \sim \mathcal{N}(0, \Sigma_n)$.

The symbol $g_{t-1} \in \mathbb{R}^{N\times 1}$ is a vector defined as

$$g_{t-1} = [g_{1,t-1}\; g_{2,t-1}\; \cdots\; g_{N,t-1}]^\top, \quad (6)$$

where

$$g_{n,t-1} = \frac{1}{1 + e^{-x_{n,t-1}}} \quad (7)$$

is a nonlinear function of the state of the nth gene at time instant $t-1$, and $u_t \in \mathbb{R}^{N\times 1}$ is a noise vector, which we assume to be zero mean Gaussian with a covariance matrix $\sigma_u^2 I$, i.e., $u_t \sim \mathcal{N}(0, \sigma_u^2 I)$, with I denoting the identity matrix. The vectors u_t are independent over time.

At each time instant t, we have measurements of all the genes. The observation model is given by [16]

$$y_t = x_t + v_t, \quad (8)$$

where $y_t \in \mathbb{R}^{N\times 1}$, is the measurement vector and $v_t \in \mathbb{R}^{N\times 1}$ is a measurement noise vector, which is assumed to be zero mean Gaussian with a covariance matrix $\sigma_v^2 I$.

Given a set of measurements $y_{1:T}$ (where the notation $y_{1:T}$ signifies $\{y_1, y_2, \cdots, y_T\}$), the primary objective is to jointly estimate the gene expressions, X_T or $x_{1:T}$, and the values of the interacting coefficients in matrix A. In solving the problem, for simplicity in the presentation, we assume that the variances σ_u^2 and σ_v^2 are known. If they are unknown, the proposed method can readily be replicated.

3. PROPOSED METHOD

We formulated the problem in a state-space form. The problem is nonlinear because the matrix A is unknown. Towards solving it, one can invoke one of the available methods for processing of nonlinear state-space models. Here, we adopt the particle filtering methodology [18, 19] that provides an approximation of the joint posterior distribution of the state given the measurements. This approximation is recursively obtained by means of a random measure composed of particles generated from a proposal distribution and weights assigned to the particles. With the particles and the corresponding weights, one can straightforwardly calculate point estimates of the state.

The main concern of traditional PFs is that their performance suffers with the increase of dimensionality of the system. The generation of significant particles in high-dimensional systems becomes extremely challenging to the point that the performance of the PF becomes unacceptable in such scenarios. The problem at hand can involve many genes and therefore the applicability of standard particle filtering becomes questionable.

We propose to use a bank of PFs, where each filter is assigned to track a separate gene and estimate one row of the matrix A. Splitting the overall state-space in subspaces and tackling each of them with a different PF results in a very efficient scheme since the dimensionality of the problem addressed by each individual PF becomes feasible [26, 27]. This is the main idea followed by the multiple particle filters (MPFs). It is important to note that since the genes interact, the filters need to cooperate and communicate some information for correct functioning of the overall scheme. Different strategies have been explored for communication but for simplicity here we adopt a scheme where estimates of the unknowns are shared [27].

More specifically, the problem is split into N identical subproblems. Before we proceed, we provide with some notation related to the state of the nth gene at time instant $t-1$. Let us define

$$\tilde{x}_{n,t-1} = [x_{n,2}\; x_{n,3}\; \cdots\; x_{n,t-1}]^\top, \quad (9)$$

$$\tilde{u}_{n,t-1} = [u_{n,2}\; u_{n,3}\; \cdots\; u_{n,t-1}]^\top, \quad (10)$$

$$G_{t-2}^\top = [g_1\; g_2\; \cdots\; g_{t-2}], \quad (11)$$

where $\tilde{x}_{n,t-1} \in \mathbb{R}^{(t-2)\times 1}$ represents the time series of the expressions of the nth gene from time instant 2 to $t-1$, $\tilde{u}_{n,t-1}$ the corresponding state noise sequence and $G_{t-2} \in \mathbb{R}^{(t-2)\times N}$ contains the necessary information about the function that drives the system. Then, we can write

$$\tilde{x}_{n,t-1} = G_{t-2}a_n + \tilde{u}_{n,t-1}. \quad (12)$$

The nth PF addresses the joint estimation of $x_{n,t}$ and a_n. For the expression of the nth gene at t, we have

$$x_{n,t} = g_{t-1}^\top a_n + u_{n,t}, \quad (13)$$

and we want to determine the posterior distribution of $x_{n,t}$ given the measurement $y_{n,t}$, and the past values of all the genes X_{t-1}, i.e., $p(x_{n,t}|X_{t-1}, y_{n,t})$. It is obtained from

$$p(x_{n,t}|X_{t-1}, y_{n,t}) \propto p(y_{n,t}|x_{n,t})p(x_{n,t}|X_{t-1})$$

$$= p(y_{n,t}|x_{n,t}) \int p(x_{n,t}|a_n, X_{t-1})p(a_n|X_{t-1})da_n. \quad (14)$$

First we obtain $p(a_n|X_{t-1})$, for which we write

$$p(a_n|X_{t-1}) \quad \propto \quad p(a_n)\prod_{k=2}^{t-1} p(x_{n,k}|a_n, x_{k-1}), \tag{15}$$

where $p(a_n)$ is given by (5) and $p(x_{n,k}|a_n, x_{k-1})$ is given by,

$$p(x_{n,k}|a_n, x_{k-1}) \quad = \quad \mathcal{N}\left(g_{k-1}^\top a_n, \sigma_u^2\right). \tag{16}$$

We readily deduce that

$$p(a_n|X_{t-1}) \quad = \quad \mathcal{N}\left(\eta_{n,t-1}, C_{n,t-1}\right), \tag{17}$$

where

$$\eta_{n,t-1} \quad = \quad \frac{1}{\sigma_u^2} C_{n,t-1} G_{t-2}^\top \tilde{x}_{n,t-1}, \tag{18}$$

$$C_{n,t-1} \quad = \quad \left(\Sigma_n^{-1} + \frac{1}{\sigma_u^2} G_{t-2}^\top G_{t-2}\right)^{-1}. \tag{19}$$

Next we integrate out a_n by using (14). Since

$$p(x_{n,t}|a_n, X_{t-1}) \quad = \quad \mathcal{N}\left(g_{t-1}^\top a_n, \sigma_u^2\right), \tag{20}$$

we immediately obtain

$$p(x_{n,t}|X_{t-1}) \quad = \quad \mathcal{N}\left(\nu_{n,t}, \xi_{n,t}\right), \tag{21}$$

where

$$\nu_{n,t} \quad = \quad g_{t-1}^\top \eta_{n,t-1}, \tag{22}$$

$$\xi_{n,t} \quad = \quad \sigma_u^2 + g_{t-1}^\top C_{n,t-1} g_{t-1}. \tag{23}$$

At last, we obtain $p(x_{n,t}|X_{t-1}, y_{n,t})$ from (14). Since

$$p(y_{n,t}|x_{n,t}) \quad = \quad \mathcal{N}(x_{n,t}, \sigma_v^2), \tag{24}$$

we have

$$p(x_{n,t}|X_{t-1}, y_{n,t}) \quad = \quad \mathcal{N}(\lambda_{n,t}, \psi_{n,t}), \tag{25}$$

where

$$\lambda_{n,t} \quad = \quad \frac{\xi_{n,t}}{\xi_{n,t} + \sigma_v^2} y_{n,t} + \frac{\sigma_v^2}{\xi_{n,t} + \sigma_v^2} \nu_{n,t}, \tag{26}$$

$$\psi_{n,t} \quad = \quad \frac{\sigma_v^2 \xi_{n,t}}{\sigma_v^2 + \xi_{n,t}}. \tag{27}$$

Now we are ready to apply the particle filtering algorithm. Suppose that at time $t-1$, the nth PF has a random measure

$$\chi_{n,t-1} = \{x_{n,t-1}^{(m)}, w_{n,t-1}^{(m)}\}_{m=1}^M, \tag{28}$$

where $x_{n,t-1}^{(m)}$ are the particles, $w_{n,t-1}^{(m)}$ are their respective weights, and M is the total number of particles. Furthermore, we also assume that this filter has an estimate of a_n given by $\eta_{n,t-1}$. The filter must also have the estimates of the values of the other genes for time instant $t-1$ obtained by the other PFs. We denote these estimates $\widehat{X}_{-n,t-1}$ (the subindex $-n$ indicates that the vector contains the estimates of all the other PFs except of that of the nth filter). The algorithm proceeds as follows:

1. Generation of particles using (25):

$$x_{n,t}^{(m)} \quad \sim \quad \mathcal{N}(\lambda_{n,t}^{(m)}, \psi_{n,t}^{(m)}), \tag{29}$$

where $\lambda_{n,t}^{(m)}$ and $\psi_{n,t}^{(m)}$ are computed via (26) and (27), respectively. We note that $\mathcal{N}(\lambda_{n,t}^{(m)}, \psi_{n,t}^{(m)})$ is the optimal importance function. We also reiterate that in order to obtain the parameters of the distribution one will need the estimates from the other filters, $\widehat{X}_{-n,t-1}$.

2. Computation of a prediction or estimate of the state of the gene, $\widehat{\widehat{x}}_{n,t}$, and transmission to the other filters.

3. Calculation of weights according to [18]:

$$\begin{aligned} \tilde{w}_{n,t}^{(m)} \quad &\propto \quad p(y_{n,t} \mid X_{n,t-1}^{(m)}, \widehat{\widehat{X}}_{-n,t-1}) \\ &= \quad \frac{1}{\sqrt{2\pi(\sigma_v^2 + \xi_{n,t}^{(m)})}} \exp\left(-\frac{\left(y_{n,t} - \nu_{n,t}^{(m)}\right)^2}{2(\sigma_v^2 + \xi_{n,t}^{(m)})}\right) \end{aligned} \tag{30}$$

4. Normalization of weights: $w_{n,t}^{(m)} = \frac{\tilde{w}_{n,t}^{(m)}}{\sum_{j=1}^M \tilde{w}_{n,t}^{(j)}}$

5. Estimation of the gene state, $\hat{x}_{n,t} = \sum_{m=1}^M w_{n,t}^{(m)} x_{n,t}^{(m)}$, and transmission to the other filters.

6. Resampling.

At the end of this process one obtains the random measure $\chi_{n,t},\ n = 1, \cdots, N$ that allows for approximation of the marginal posteriors

$$\begin{aligned} p(x_{n,t} \mid y_{n,1:t}) \quad &\approx \quad p_M(x_{n,t} \mid y_{n,1:t}) \\ &= \quad \textstyle\sum_{m=1}^M w_{n,t}^{(m)} \delta(x_{n,t} - x_{n,t}^{(m)}). \end{aligned} \tag{31}$$

4. SIMULATION RESULTS

We applied the proposed approach to the setup discussed in [22]. The gene network consisted of 8 genes whose expression was characterized by the model given in (2), where the matrix A was the one shown on the next page. The data were generated for 40 time instants and the variance of the noises were $\sigma_u^2 = 10^{-4}$ and $\sigma_v^2 = 10^{-4}$. The prior of each gene at each filter was Gaussian with zero mean and variance 10, and the prior of the a_n vectors was Gaussian with zero mean and covariance matrix $50\,I$.

We compared the proposed MPF algorithm that used 8 filters (one per gene) and $M = 75$ particles per filter with the standard PF (SPF) that employed only one filter tracking a state of dimension 8 (8 genes at once) with a total of 2000 particles. We note that the SPF was using almost three times as many particles as the MPF. Figure 1 shows the evolution of the expression of one of the genes at one particular run and the estimates obtained by both filters. It is clear that the SPF cannot track properly. However, the MPF shows a very good agreement with the true trajectory.

Figure 2 displays histograms corresponding to means of the marginal distributions of the coefficients (see (17)) obtained in the last step of the MPF for one of the genes. The results reveal a good

$$A = \begin{bmatrix} 0 & 0 & 0 & .6 & .7 & 0 & 1.9 & 2.9 \\ -.1 & 0 & 0 & 3.5 & 0 & -2.1 & 0 & 3.4 \\ -4.4 & .9 & -1.7 & -.3 & 3.4 & 0 & 1.7 & 0 \\ 0 & .5 & 2.8 & -3.7 & .9 & 0 & 0 & -3.1 \\ 0 & .2 & 0 & -2.6 & -3.2 & -.1 & -.5 & 4 \\ -.5 & -1.8 & 0 & 3.4 & 1.4 & 1.1 & 0 & -1.7 \\ -.8 & 0 & 0 & -3 & 1.1 & .4 & 0 & 0 \\ -.3 & 0 & -1 & 0 & .1 & 0 & 0 & 2.2 \end{bmatrix}.$$

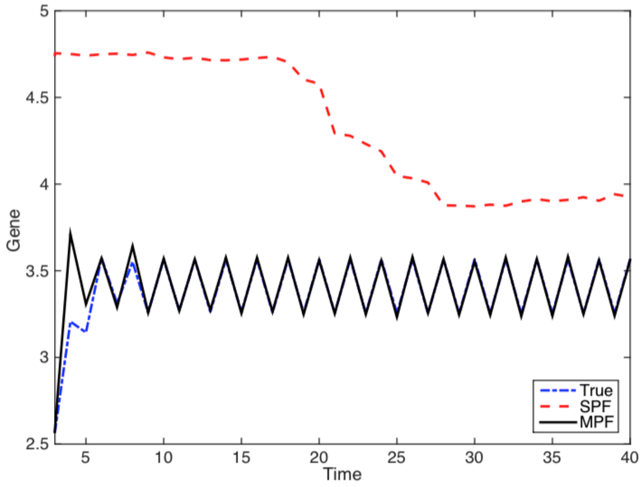

Fig. 1. State estimates of a particular gene in one specific run of the SPF and MPF methods. The blue dash-dot line shows the true evolution.

performance of the algorithm even though the number of particles is rather low.

Finally, we obtained the mean square error (MSE) of the methods for estimation of some of the gene expressions averaged over 50 runs. Figure 3 illustrates the poor performance of the SPF when dealing with all the genes at the same time. The performance of the MPF remains superior because of the strategy based on splitting the system into low-dimensional ones. We reiterate that the SPF uses more than three times the total number of particles used by the MPF. We also observe that for some of the genes the MPF needs more time steps to achieve good performance. In our experiments, the MSE became very small in less than 10 time instants.

5. CONCLUSION

In this paper we introduce a new efficient particle filter method for estimating the time series of gene expressions and identification of the matrix whose elements represent the interaction coefficients and that defines the gene network. The method uses as many filters as genes in the network and each of them operates using the marginal posterior distribution for optimal sampling of the particles. Under this strategy the coefficient matrix is marginalized and there is no need to produce particles for the coefficients. The numerical results show a clear advantage of the proposed approach when compared to the standard particle filter that uses only one filter to track all the genes.

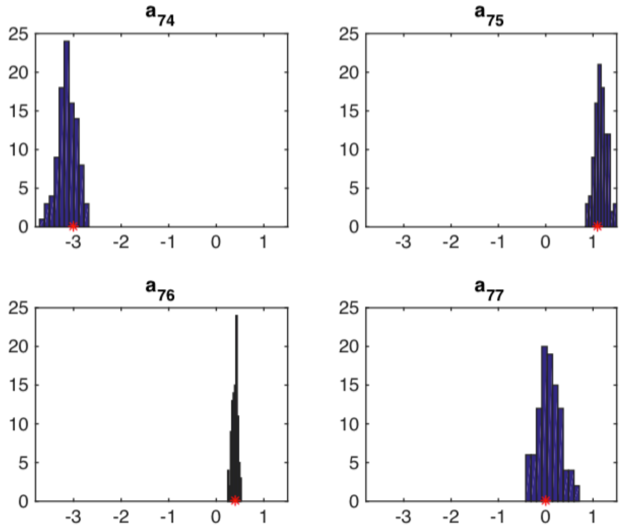

Fig. 2. Histograms for some of the coefficients of interactions corresponding to gene 7 in one specific run. The red stars at the bottom mark the true values of the interactions.

6. REFERENCES

[1] S. A. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *Journal of theoretical biology*, vol. 22, no. 3, pp. 437–467, 1969.

[2] I. Shmulevich, E. R. Dougherty, S. Kim, and W. Zhang, "Probabilistic boolean networks: a rule-based uncertainty model for gene regulatory networks," *Bioinformatics*, vol. 18, no. 2, pp. 261–274, 2002.

[3] T. Tian and K. Burrage, "Stochastic neural network models for gene regulatory networks," in *Evolutionary Computation, 2003. CEC'03. The 2003 Congress on*. IEEE, 2003, vol. 1, pp. 162–169.

[4] T. Chen, H. L. He, G. M. Church, et al., "Modeling gene expression with differential equations.," in *Pacific symposium on biocomputing*, 1999, vol. 4, p. 4.

[5] N. Friedman, "Inferring cellular networks using probabilistic graphical models," *Science*, vol. 303, no. 5659, pp. 799–805, 2004.

[6] N. Friedman, M. Linial, I. Nachman, and D. Pe'er, "Using bayesian networks to analyze expression data," *Journal of computational biology*, vol. 7, no. 3-4, pp. 601–620, 2000.

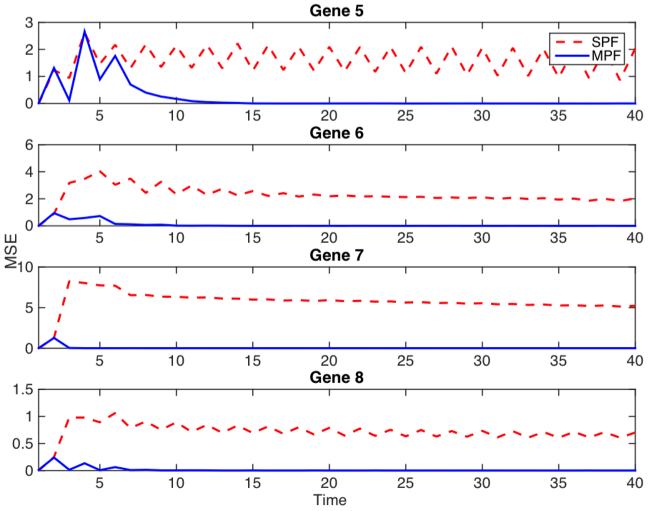

Fig. 3. MSE comparison of the MPF and SPF for four different genes.

[7] K. Murphy, S. Mian, et al., "Modelling gene expression data using dynamic bayesian networks," Tech. Rep., Technical report, Computer Science Division, University of California, Berkeley, CA, 1999.

[8] P. Li, C. Zhang, E. J. Perkins, P. Gong, and Y. Deng, "Comparison of probabilistic boolean network and dynamic bayesian network approaches for inferring gene regulatory networks," *BMC bioinformatics*, vol. 8, no. Suppl 7, pp. S13, 2007.

[9] M. Zou and S. D. Conzen, "A new dynamic Bayesian network (DBN) approach for identifying gene regulatory networks from time course microarray data," *Bioinformatics*, vol. 21, no. 1, pp. 71–79, 2005.

[10] J. Zhu, Y. Chen, A. S. Leonardson, K. Wang, J. R. Lamb, V. Emilsson, and E. E. Schadt, "Characterizing dynamic changes in the human blood transcriptional network," *PLoS computational biology*, vol. 6, no. 2, pp. e1000671, 2010.

[11] W. C. Young, A. E. Raftery, and K. Y. Yeung, "Fast Bayesian inference for gene regulatory networks using ScanBMA," *BMC systems biology*, vol. 8, no. 1, pp. 47, 2014.

[12] F.-X. Wu, W.-J. Zhang, and A. J. Kusalik, "Modeling gene expression from microarray expression data with state-space equations.," in *Pacific Symposium on Biocomputing*, 2004, vol. 9, pp. 581–592.

[13] Z. Wang, F. Yang, D. W. Ho, S. Swift, A. Tucker, and X. Liu, "Stochastic dynamic modeling of short gene expression time-series data," *NanoBioscience, IEEE Transactions on*, vol. 7, no. 1, pp. 44–55, 2008.

[14] Y. Huang, I. Tienda-Luna, and Y. Wang, "Reverse engineering gene regulatory networks," *Signal Processing Magazine, IEEE*, vol. 26, no. 1, pp. 76–97, 2009.

[15] B.-E. Perrin, L. Ralaivola, A. Mazurie, S. Bottani, J. Mallet, and F. dAlche Buc, "Gene networks inference using dynamic bayesian networks," *Bioinformatics*, vol. 19, no. suppl 2, pp. ii138–ii148, 2003.

[16] Z. Wang, X. Liu, Y. Liu, J. Liang, and V. Vinciotti, "An extended kalman filtering approach to modeling nonlinear dynamic gene regulatory networks via short gene expression time series," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 6, no. 3, pp. 410–419, 2009.

[17] A. Noor, E. Serpedin, M. Nounou, and H. Nounou, "Reverse engineering sparse gene regulatory networks using cubature kalman filter and compressed sensing," *Advances in bioinformatics*, vol. 2013, 2013.

[18] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*, Springer, 2001.

[19] P. M. Djurić, J. H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. F. Bugallo, and J. Míguez, "Particle filtering," *IEEE Signal Processing Magazine*, vol. 20, no. 5, pp. 19–38, 2003.

[20] X. Shen and H. Vikalo, "Inferring parameters of gene regulatory networks via particle filtering," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, pp. 5, 2010.

[21] Q. Bo and W. Zheng-Zhi, "Application of unscented particle filtering for estimating parameters and hidden variables in gene regulatory network," in *Bioinformatics and Biomedical Engineering (iCBBE), 2010 4th International Conference on*. IEEE, 2010, pp. 1–4.

[22] A. Noor, E. Serpedin, M. Nounou, and H. Nounou, "Inferring gene regulatory networks via nonlinear state-space models and exploiting sparsity," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 9, no. 4, pp. 1203–1211, 2012.

[23] B. Jia and X. Wang, "Gene regulatory network inference by point-based gaussian approximation filters incorporating the prior information," *EURASIP Journal on Bioinformatics and Systems Biology*, vol. 2013, no. 1, pp. 1–16, 2013.

[24] T. Bengtsson, P. Bickel, and B. Li, "Curse-of-dimensionality revisited: Collapse of the particle filter in very large scale systems," *Probability and Statistics: Essays in honor of David A. Freedman*, vol. 2, pp. 316–334, 2008.

[25] F. Daum and J. Huang, "Curse of dimensionality and particle filters," in *Proceedings of the IEEE Aerospace Conference*, 2003.

[26] M. F. Bugallo and P. M. Djurić, "Target tracking by multiple particle filtering," in *Proceedings of the IEEE Aerospace Conference*, 2007.

[27] P. M. Djurić, T. Lu, and M. F. Bugallo, "Multiple particle filtering," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2007.