Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2019) 27: 2648 – 2665
© TÜBİTAK
doi:10.3906/elk-1809-187

A comparative study of nonlinear Bayesian filtering algorithms for estimation of gene expression time series data

Nesrine AMOR[*], **Asma MEDDEB**, **Sahbi MARROUCHI**, **Souad CHEBBI**
National Superior School of Engineers of Tunis (ENSIT), University of Tunis, Tunisia

Received: 26.09.2018 • **Accepted/Published Online:** 19.02.2019 • **Final Version:** 26.07.2019

Abstract: This paper addresses the problem of estimating the time series of a gene expression using nonlinear Bayesian filtering algorithms. The response of gene regulatory networks (GRNs) to functional requirements in the cell and environmental conditions evolves over time. Dynamic biological processes such as cancer progression and treatment recovery depend on the collected genetic profiles. These processes are behind genetic interactions that rewire over the course of time. The GRN was formulated as a nonlinear and non-Gaussian dynamic system defined by the gene measurement model and the unknown state is an evolution of the gene model. However, the GRN has a high dimensional space where most of nonlinear Bayesian filtering algorithms are ineffective in high dimensional spaces. Therefore, many authors have introduced various techniques to overcome what has become known as the curse of dimensionality. This paper presents a comparative study between extended Kalman filter, unscented Kalman filter and derivatives of particle filters, in tracking the evolution of gene expression over time. Application of the nonlinear Bayesian filtering algorithms to estimate the evolution of gene expression from synthetic and real data, shows that the unscented particle filter (UKF-PF) provides promising and robust results compared to other filters. Furthermore, UKF-PF provides an alternative solution to the problem of modeling gene regulatory networks.

Key words: Extended Kalman filter, unscented Kalman filter, particle filter, gene regulatory network

1. Introduction

The biological mechanisms that govern our development are complex and crucial to understand the cellular system. The gene regulatory network (GRN) controls the expression of thousands of proteins and genes in any specific cellular function. However, the biological processes are dynamic and evolve over time in response to various extrinsic and intrinsic factors, such as cellular development, targeted therapy disease progression and environmental conditions [1]. Understanding these gene regulatory networks can help us significantly enrich our knowledge of health and disease.

A gene regulatory network is a group of genes that interact with each other and with other substances to control the functions of the cell. Inference of gene regulatory relationships is a very important problem in biology [2, 3]. The unknown state is estimated based on time series data that represent the evolution of the genomic expressions [1, 4]. However, the estimation of gene expression is formulated as a nonlinear problem, and the unknown state has a high dimensional state space model. Various techniques have been introduced to estimate the gene expression time series including extended Kalman filter (EKF) [1] and multiparticle filtering [2].

[*]Correspondence: nisrine.amor@hotmail.fr

The EKF is the most widely utilized estimation algorithm for nonlinear state-space models. However, the great amount of experience of more than 40 years in the signal processing community has revealed many shortcomings of EKF that make it difficult to implement, i.e difficulties in determining the Jacobians, errors introduced by linearization, or the inability to handle with systems with asymmetric or multimodal probability density functions. Therefore, EKF is only robust for systems that are almost linear during the time scale of the updates. To overcome this limitation, the unscented kalman filter (UKF) was introduced as a method based on unscented transform (UT) [5, 6], where its performance is always better than the EKF.

Particle filters (PF) are widely used for latent state estimation/tracking in dynamic systems where systems dynamics or observation models are nonlinear and non-Gaussian [7]. The technique of PF is based on powerful sampling that aims to find an optimal estimate by exploiting a set of random weighted samples called 'particles'. These particles are used to approximate the posterior density of the state [7]. Due to the complex nature of computations in PF, it is not straightforward to handle with a high dimensional state space.

Particle Markov chain Monte Carlo (PF-MCMC) algorithms have been proposed to improve particle filtering performance and robustness in the high-dimensional state-spaces [8]. The main idea of the PF-MCMC method is to use the Metropolis–Hastings acception/rejection step as a correction to generate the best proposal distribution. On the other hand, EKF and UKF were used to generate importance density functions [6, 9]. Specifically, at every step, EKF or UKF is used to generate the mean and covariance of the proposal distribution per particle. Then, particles are drawn from the newly found distributions. The obvious advantage is that the EKF and UKF take into account the most recent measurement while estimating mean and covariance [10].

The main contribution of this paper is using the nonlinear Bayesian filtering algorithms from Extended Kalman Filter to Unscented Particle filter for the estimation of gene expression where the state estimation is a nonlinear and has a high dimensional state space model. Simulation results on synthetic and real data will support the comparison between EKF, UKF, PF, PF-MCMC, EKF-PF, and UKF-PF for estimation of gene expression. We will show that UKF-PF provides effective results and will be an alternative solution for estimating the regulatory gene network.

The paper is organized as follows: In section 2, we review the nonlinear Bayesian filtering algorithms. In section 3, we introduce the problem formulation of tracking gene expressions. In section 4, we present and discuss the simulation results. Finally, section 5 summarizes the main conclusions followed by references.

2. Nonlinear Bayesian filtering algorithms

2.1. Bayesian recursion

We consider the general state-space model defined by the state transition and measurement models in a discrete form given by:

$$x_{n+1} = f_n(x_n) + u_n, \tag{1}$$

$$y_n = h_n(x_n) + v_n, \tag{2}$$

where $x_n \in \mathbb{R}^{n_x}$ and $y_n \in \mathbb{R}^{n_y}$ are, respectively, the hidden state vector with transition probability density functions (PDFs) $p(x_n|x_{n-1})$, and the observation vector with conditional PDFs $p(y_n|x_n)$ at time instant n. f_n and h_n are possibly nonlinear state transition and observation functions, respectively. n_x and n_y are state and observation dimensions. u_n and v_n are zero-mean state and observation white noise sequences with known

PDFs, respectively, $p(u)$ and $p(v)$. Both noise sequences are supposed to be uncorrelated with each other and the initial condition of the state x_0 given by $p(x_0)$.

State estimation is designed to find the state x_n using the available measurements up to time n, $y_{1:n} = [y_1, ..., y_n]$. The solution to this problem is the density of the system state conditional on the measurements; either joint PDF $p(x_1, ..., x_n|y_{1:n})$ or the marginal PDF $p(x_n|y_{1:n})$.

In the Bayesian framework, a recursion can be defined to estimate *a posteriori* conditional PDFs $p(x_n|y_{1:n})$ using the *a priori* and *likelihood* PDFs, are respectively, $p(x_n|x_{n-1})$ and $p(y_n|x_n)$ [7]. For instance, the optimal minimum mean-square-error estimate of x_n is given by the mean of the posterior density, i.e., $E(x_n|y_{1:n})$. Using Bayes rule and Chapman–Kolmogorov equation, the posterior distribution can be computed recursively using the following two-step formulas:

- **Prediction step**

$$p(x_n|y_{1:n-1}) = \int p(x_{n-1}|y_{1:n-1})\ p(x_n|x_{n-1})\ dx_{n-1}, \tag{3}$$

- **Update step**

$$\begin{aligned} p(x_n|y_{1:n}) &= \frac{p(y_{1:n}|x_n)\ p(x_n)}{p(y_{1:n})}, &(4)\\ &= \frac{p(y_n|y_{1:n}, x_n)\ p(y_{1:n-1}|x_n)\ p(x_n)}{p(y_n|y_{1:n-1})p(y_{1:n-1})}, \\ &= \frac{p(y_n|x_n)\ p(x_n|y_{1:n-1})}{p(y_n|y_{1:n-1})}, \\ &= \frac{p(y_n|x_n)\ p(x_n|y_{1:n-1})}{\int p(y_n|x_n)\ p(x_n|y_{1:n-1})\ dx_n}. &(5)\end{aligned}$$

Unfortunately, in the nonlinear case, Eqs. (3)–(5) are only a conceptual solution because the defined integrals are generally intractable. However, analytical (closed-form) solutions in some special cases may exist, e.g., the Kalman filter for linear dynamics system, linear observation models and Gaussian densities for the noise sequences.

2.2. Extended Kalman filter

For the nonlinear model, a linearization of nonlinear functions $f_n(x_n)$ and $h_n(x_n)$, using the Taylor series expansion is used to formulate the extended Kalman filter (EKF). The state error covariance is propagated in time using the linearized functions, whereas the means are propagated using nonlinear functions. In the EKF method, the unknown state is estimated by employing first-order Taylor series approximations to the nonlinear functions as follows:

- **Prediction step**

$$\hat{x}_{n|n-1} = f_n(\hat{x}_{n-1|n-1}), \tag{6}$$

$$P_{n|n-1} = \hat{F}_n P_{n-1|n-1} \hat{F}_n^T + Q_n, \tag{7}$$

- **Update step**

$$
\begin{aligned}
S_n &= \hat{H}_n P_{n|n-1} \hat{H}_n^T + R_n, && (8)\\
K_n &= P_{n|n-1} \hat{H}_n^T S_n^{-1}, && (9)\\
\hat{x}_{n|n} &= \hat{x}_{n|n-1} + K_n(y_n - h_n(\hat{x}_{n|n-1})), && (10)\\
P_{n|n} &= P_{n|n-1} - K_n S_n K_n^T.
\end{aligned}
$$

where F_x and H_x are the Jacobian matrices of $f(x)$ and $h(x)$, respectively,

$$\hat{F}_n = \left.\frac{df_n(x)}{dx}\right|_{x=\hat{x}_{n-1|n-1}},$$

$$\hat{H}_n = \left.\frac{dh_n(x)}{dx}\right|_{x=\hat{x}_{n|n-1}}.$$

It is important to realize that in the EKF, the covariance is propagated using a linearized form of the nonlinear functions. However, the linearization of nonlinear system dynamics and observation models may induce errors in the estimation of the state, and in the worst-case, the filter may diverge especially for highly nonlinear function [6].

2.3. Unscented Kalman filter

The Unscented Kalman Filter (UKF) was proposed as a method based on a mathematical approach called the 'Unscented Transform' (UT) [11]. The UKF approximates the probability distribution to propagate the mean and covariance, based on UT which uses a deterministic set of samples called sigma points [11]. The calculated sigma points are propagated through the nonlinear function. The statistics of transformed points can be calculated to form an estimate of the nonlinearly transformed mean and covariance [5, 6, 11]. The sigma points $\mathcal{X}_n^j \in \mathbb{R}^{n_x}$, $j = 0, 1, \cdots, 2L$ (where L is dimensionality) are chosen deterministically as apposite to the particle filters. Consider x_n with mean $\hat{x}_n$ and covariance P_n. Let the matrix of all the sigma points be $\mathcal{X} := [\mathcal{X}_n^j, \cdots, \mathcal{X}_n^{2L}]$. In the following, we summarize the main steps of the UKF algorithm.

- **Prediction step**

Compute the sigma points according to the following:

$$
\begin{aligned}
\mathcal{X}_{n-1}^0 &= \hat{x}_{n-1|n-1},\\
\mathcal{X}_{n-1}^j &= \hat{x}_{n-1|n-1} + (\sqrt{(L+\gamma)P_{n-1|n-1}})_j, j = 1, \cdots, L,\\
\mathcal{X}_{n-1}^j &= \hat{x}_{n-1|n-1} - (\sqrt{(L+\gamma)P_{n-1|n-1}})_{j-L}, j = 1, \cdots, 2L,
\end{aligned}
$$

where $(\sqrt{(L+\gamma)P_{n-1|n-1}})_j$ is the row of the matrix square root of $((L+\gamma)P_{n-1|n-1})$.

Compute the weights of the sigma points by:

$$\begin{aligned}
\mathcal{W}_0^m &= \gamma/(L+\gamma), \\
\mathcal{W}_0^c &= \gamma/(L+\gamma) + (1-\alpha^2+\beta), \\
\mathcal{W}_j^m &= \mathcal{W}_j^c = 1/\{2(L+\gamma)\}, j = 1, \cdots, 2L,
\end{aligned}$$

where $\gamma = \alpha^2(L+\kappa) - L$ is a scaling parameter. α indicates the spread of sigma point around $\hat{x}$. κ represents a secondary scaling parameter and β is used to incorporate the prior distribution of x. In $\mathcal{W}_i^m$, m refers to computing the mean and in $\mathcal{W}_i^c$, c refers to computing the covariance.

Then, these sigma points should be propagated through a nonlinear function of the state model by:

$$\hat{\mathcal{X}}_n^j = f(\mathcal{X}_n^j), j = 1, \cdots, 2L, \tag{11}$$

Furthermore, the mean $\hat{x}_{n|n-1}$ and the covariance $P_{n|n-1}$ can be calculated as follows:

$$\hat{x}_{n|n-1} = \sum_{j=1}^{2L} \mathcal{W}_j^m \hat{\mathcal{X}}_n^j, \tag{12}$$

$$P_{n|n-1} = \sum_{j=1}^{2L} \mathcal{W}_j^c (\hat{\mathcal{X}}_n^j - \hat{x}_{n|n-1})(\hat{\mathcal{X}}_n^j - \hat{x}_{n|n-1})^T + +Q_n. \tag{13}$$

- **Update step**

Compute the updated sigma points by:

$$\begin{aligned}
\mathcal{X}_n^0 &= \hat{x}_{n|n-1}, \\
\mathcal{X}_n^j &= \hat{x}_{n|n-1} + (\sqrt{(L+\gamma)P_{n|n-1}})_j, j = 1, \cdots, L, \\
\mathcal{X}_n^j &= \hat{x}_{n|n-1} - (\sqrt{(L+\gamma)P_{n|n-1}})_{j-L}, j = 1, \cdots, 2L.
\end{aligned}$$

Therefore, these obtained sigma points should be propagated through a nonlinear function of the measurement model as follow:

$$\hat{\mathcal{Y}}_n^j = h(\mathcal{X}_n^j), j = 1, \cdots, 2L. \tag{14}$$

Then, calculate the estimated mean $\hat{y}_n$ and the covariance of the measurement Py_n:

$$\hat{y}_n = \sum_{j=1}^{2L} \mathcal{W}_j^m \hat{\mathcal{Y}}_n^j, \tag{15}$$

$$Py_n = \sum_{j=1}^{2L} \mathcal{W}_j^c (\hat{\mathcal{Y}}_n^j - \hat{y}_n)(\hat{\mathcal{Y}}_n^j - \hat{y}_n)^T + R_n. \tag{16}$$

In addition, the cross-covariance C_n of the state transition and the measurement are computed by:

$$C_n = \sum_{j=1}^{2L} \mathcal{W}_j^c(\mathcal{X}_n^j - \hat{x}_{n|n-1})(\hat{\mathcal{Y}}_n^j - \hat{y}_n)^T, \tag{17}$$

Finally, the mean $\hat{x}_n$ and the covariance P_n are computed by the following equations:

$$\begin{aligned} \hat{x}_n &= \hat{x}_{n|n-1} + K_n(\hat{y}_n), & (18)\\ P_n &= P_{n|n-1} - K_n P y_n K_n^T, & (19)\\ K_n &= C_n P y_n^{-1}. & (20) \end{aligned}$$

The approximations obtained with at least $2L+1$ sampling points are accurate to the third-order of Gaussian inputs for all nonlinearities and at least to the second for non-Gaussian inputs [12]. Furthermore, the UKF algorithm does not work well with nearly singular covariances due to the Cholesky decomposition failure. Also, Cholesky decomposition at each time-step may be computationally demanding. The claimed advantages with UKF are that it is more accurate and easier to implement than EKF by avoiding the requirement of using the Jacobians in the algorithm.

The above methods are not robust when the problem is highly non-Gaussian and/or nonlinear. The particle filters (PF) are able to proceed better in these situations. The PF are flexible and simple simulation-based numerical approaches applied for estimating the state in a sequential manner.

2.4. Particle filtering

Particle filters solve the optimal estimation problem in nonlinear and non-Gaussian dynamic systems by incorporating sequential Monte Carlo sampling with a Bayesian filtering framework [7, 13, 14]. The PF approximates the posterior density using a group of weighted samples called also particles. This approximation converges, in the mean-square error and under mild conditions, to the true posterior density of the state [7].

2.4.1. Sequential Monte Carlo approximation in Bayesian inference

In Bayesian statistics, an optimal state estimation is given by computing the conditional mean of the posterior density as:

$$E[g(x_n)|y_{1:n}] = \int g(x_n)\ p(x_n|y_{1:n})\ dx_n, \tag{21}$$

where g represents an arbitrary (linear or nonlinear) function. In general, many numerical methods are used to compute this integral in order to evaluate it in closed form. Monte Carlo methods have a powerful numerical method for computing integrals of Eq. (21) in closed form by generating N random samples, and the solution is then estimated by averaging these samples as:

$$\hat{E}[g(x_n)|y_{1:n}] \approx \frac{1}{N}\sum_{i=1}^{N} g(x_n^i). \tag{22}$$

It is important to highlight that the convergence of the Monte Carlo approximation is almost surely, i.e $\hat{E}[g(x_n)|y_{1:n}] \to E[g(x_n)|y_{1:n}]$ when the number of particles $N \to \infty$ [14].

2.4.2. Importance sampling (IS)

Generally, it is impossible to sample from the true posterior because integral is intractable. Therefore, a proposal distribution or an importance distribution $q(x_n|y_{1:n})$ is defined [7, 15]. The posterior density is presented by the Bayes theorem as:

$$p(x_n|y_{1:n}) = \frac{p(y_{1:n}|x_n)p(x_n)}{p(y_{1:n})}, \tag{23}$$

By substituting Eq. (23) into Eq. (21), we get

$$\int g(x_n)\ p(x_n|y_{1:n})\ dx_n = \int g(x_n)\ w_n\ q(x_n|y_{1:n})\ dx_n, \tag{24}$$

where w_n represents the (unnormalized) importance weights computed by:

$$w_n \quad = \quad \frac{p(y_{1:n}|x_n)p(x_n)}{q(x_n|y_{1:n})}. \tag{25}$$

Using the technique of importance sampling, we can generate N samples from the proposal distribution as:

$$x_n^i \sim q(x_n|y_{1:n}),\ where\ i = 1, ...N. \tag{26}$$

Recall that the Eq. (4) provided by the Bayes rule can be rewritten as:

$$p(x_n|y_{1:n}) = \frac{p(y_{1:n}|x_n)\ p(x_n)}{\int p(y_{1:n}|x_n)p(x_n)\ dx_n}.. \tag{27}$$

By incorporating Eq. (27) into Eq. (21), we obtain the conditional mean estimate by:

$$E[g(x_n)|y_{1:n}] = \sum_{i=1}^{N} \tilde{w}_n^i\ g(x_n^i), \tag{28}$$

where $\tilde{w}_n^i$ represents the normalized importance weight for particle i given by:

$$\tilde{w}_n^i \quad = \quad \frac{w_n^i}{\sum_{i=1}^{N} w_n^i}, \tag{29}$$

The approximation of the posterior probability distribution can be written as:

$$p(x_n|y_{1:n}) \approx \sum_{i=1}^{N} w_n^i \delta(x_n - x_n^i), \tag{30}$$

where δ is the dirac delta function.

2.4.3. Sequential importance sampling (SIS)

Sequential importance sampling technique represents a sequential version of importance sampling. Since dynamic systems evolve over time n, the SIS algorithm is used to estimate the posterior distribution $p(x_n|y_{1:n})$ sequentially to track the evolution of dynamic systems over time. The SIS approximates the posterior PDF of the state, using a set of N particles and their associated weights $\{x_n^{(i)}, w_n^{(i)}\}_{i=1}^N$ as:

$$p(x_n|y_{1:n} \approx \sum_{i=1}^{N} w_n^i \delta(x_n - x_n^i), \tag{31}$$

The SIS algorithm is started by generating N samples or particles using the importance distribution $q(x)$. These particles are then weighted using w^i in order to approximate to posterior density $p(x_n|y_{1:n})$. Therefore, it is important to derive the expression of w^i in the form that allows evolving over time. Using the Bayes theorem and the Markov properties of the state-space model, we get the following expression:

$$\begin{aligned} p(x_{1:n}|y_{1:n}) &\approx p(y_n|x_{1:n}, y_{1:n-1})p(x_{1:n}|y_{1:n-1}) \\ &= p(y_n|x_n)p(x_n|x_{1:n-1})p(x_{1:n-1}|y_{1:n-1}), \end{aligned} \tag{32}$$

By using the similar rationale as in the previous part of the importance sampling, we can calculate the importance weights by:

$$w_n^i \propto \frac{p(y_n|x_n^i)p(x_n^i|x_{1:n-1}^i)p(x_{1:n-1}^i|y_{1:n-1})}{q(x_{1:n}^i|y_{1:n})}. \tag{33}$$

The proposal distribution can be rewritten as follows:

$$q(x_{1:n}|y_{1:n}) = q(x_n|x_{1:n-1}, y_{1:n})q(x_{1:n-1}|y_{1:n-1}). \tag{34}$$

Therefore, the unnormalized weights w_n^i can be computed using the following expression:

$$w_n^i \quad \propto \quad \frac{p(y_n|x_n^i)p(x_n^i|x_{1:n-1}^i)}{q(x_n^n|x_{1:n-1}^i, y_{1:n})} w_{n-1}^i. \tag{35}$$

2.4.4. Sequential importance resampling (SIR)

One issue with the particle filter that the weights of particles may be zero or close to zero, which is known as degeneracy problem. To address this issue, resampling techniques have become an essential step of the particle filter algorithm to avoid the degeneracy problem due to zero or near zero weights of a large set of particles. The most popular resampling algorithm is the one that selects the particles according to their weights by removing particles with very small weights and duplicating particles with large weights [16]. As a result, equal weights ($\frac{1}{N}$) are assigned to all selected N particles.

2.4.5. Particle filtering

The particle filters is a sequential Monte Carlo method to estimate the posterior density of the state. The PFs approximates the posterior PDF of the state, using a set of N particles and their associated weights

$\{x_n^{(i)}, w_n^{(i)}\}_{i=1}^N$ [7, 17]. The conditional mean estimate of the state is then given by:

$$\widehat{x}_n = E[x_n|y_{1:n}] = \int p(x_n|y_{1:n})\ dx_n \approx \sum_{i=1}^{N} w_n^{(i)} x_n^{(i)}. \tag{36}$$

Ideally, the particles are required to be sampled from the true posterior, $p(x_n|y_{1:n})$, which is not available. Therefore, another distribution, referred to as the importance distribution or the proposal distribution $q(x_n|x_{n-1}, y_n)$, is used. Theoretically, the only condition on the importance distribution is that its support includes the support of the posterior distribution [17]. In practice, the number of particles is finite and the importance distribution should be chosen to approximate the posterior density. The importance weights are given:

$$\tilde{w}_n^i = \frac{p(y_n|x_n^i)p(x_n^i|x_{1:n-1}^i)}{q(x_n^n|x_{1:n-1}^i, y_{1:n})} w_{n-1}^i. \tag{37}$$

The normalized weight of particle i at time n in Eq. (37) is given by $w_n^{(i)} = \tilde{w}_n^{(i)} / \sum_{j=1}^{N} w_n^{(j)}$. It can be shown that the particle filter converges asymptotically, as $N \rightarrow \infty$, towards the optimal filter in the mean square error sense [14].

2.4.6. Derivatives of particle filters

Despite the powerfulness and the robustness of PF, it has a major problem in determining the proposal distribution or called also the importance distribution of the particles. In addition, PF is ineffective in high dimensional spaces where the number of particles required increases superexponentially with the dimension of the state [7]. The durability of PF is mainly based on generating of the proposal distributions especially in high dimensional state spaces. Therefore, several methods have emerged to ameliorate the performance of particle filters as follows: the Particle Markov chain Monte Carlo (PF-MCMC) [8], the extended particle filter (EKF-PF) and the UKF-PF [6, 9].

- Particle-MCMC filter: The PF-MCMC method is an incorporation of Markov chain Monte Carlo (MCMC) with particle filter methods to create an efficient and powerful high dimensional proposal distributions design.
- Extended particle filter: The EKF is applied to generate the importance distribution. However, the linearization process in EKF-PF offers modeling errors, which can lead to large estimation errors if the system is severely nonlinear.
- Unscented particle filter: The UKF-PF was proposed using UKF to generate the proposal distribution and it has been shown that UKF is able to provide a better and a more accurate performance compared to the EKF in the generation of the proposal distributions.

3. Estimation of gene expression

The state-space system of dynamical gene expression is represented by a state transition and observation model given by:

$$x_n = f_n(x_{n-1}) + u_n, \tag{38}$$

$$y_n = h_n(x_n) + v_n, \tag{39}$$

where f_n is a nonlinear function which represents the regulatory relationship between different genes. x_n represents the gene expression at a time instant n. u_n is an additional gaussian noise. y_n represents the micro-array data with additional gaussian noise v_n.

The genes expressions evolve over time and its corresponding matrix at time instant n is defined as:

$$X_n = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\ \vdots & \vdots & \vdots & \vdots \\ x_{N,1} & x_{N,2} & \cdots & x_{N,n} \end{bmatrix}, \tag{40}$$

where N is the number of genes and $x_{n,j}$ represents the evolution of the gene j at time n. Therefore, we can rewrite the estimated state at time n by $x_n = [x_{1,n}, x_{1,n}, \cdots, x_{1,n}]^T$. Furthermore, the evolution of genes expressions is introduced by:

$$x_n = Ag_{n-1} + u_n, \tag{41}$$

where A represents the matrix coefficient of the regulatory relationship between genes and g_{n-1} represents a nonlinear function (a sigmoid squash function) of the transition state given by:

$$g_{j,n-1} = \frac{1}{1+e^{-x_{j,n-1}}}. \tag{42}$$

Given a collection of observations $y_{1:T}$, the main objective is to estimate the gene expressions $x_{1:T}$, and its matrix A.

We estimated the gene expression problem using various nonlinear Bayesian filtering algorithms, including the EKF, UKF, PF, EKF-PF, UKF-PF, and PF-MCMC. We computed the coefficients of regulatory relationship matrix A in the same method presented in [2, 3], where the priors of these coefficients were calculated using a normal distribution.

4. Simulation results and discussion

The proposed algorithm for estimating the evolution of gene expression using nonlinear Bayesian algorithms are presented in Figure 1. The input is synthetic or real-world biological data, and the output is the estimated gene expression over time. We have applied the nonlinear Bayesian filtering algorithms, i.e. EKF, UKF, PF, PF-MCMC, EKF-PF, and UKF-PF to estimate the gene expression from synthetic and real time series data.

4.1. Simulation results on synthetic data

We applied nonlinear Bayesian filtering algorithms to estimate the evolution of the gene expression, using the setting described in [3]. The regulatory gene network was described by the model presented in Eq. (41), the coefficients of matrix A are calculated using the normal distribution and it was presented on the next page. The data were generated for 60 time-steps. We used the EKF, UKF, PF, PF-MCMC, EKF-PF, and UKF-PF for estimate gene expression over time. The GRN consists of 8 genes, i.e. the state vector $x = [x_n^1, ..., x_n^8]^t$. The prior of every gene is a Gaussian with a zero mean and a variance 10^{-1}. We used 5000 particles for PF,

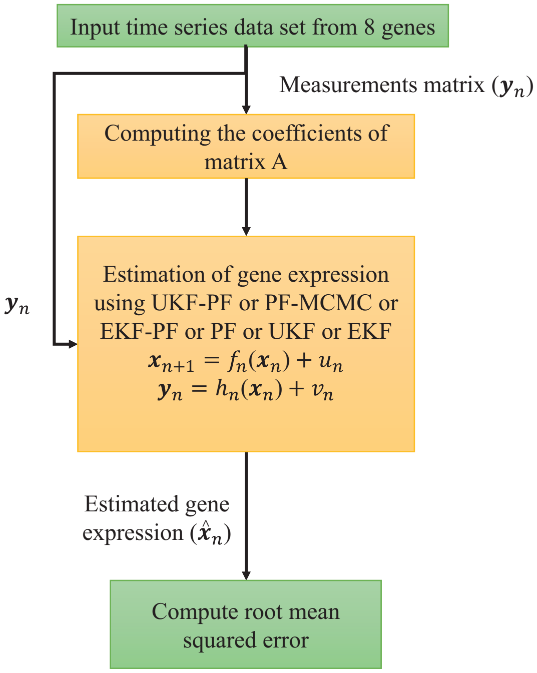
Input time series data set from 8 genes
Measurements matrix ($\boldsymbol{y}_n$)
Computing the coefficients of matrix A
$\boldsymbol{y}_n$
Estimation of gene expression using UKF-PF or PF-MCMC or EKF-PF or PF or UKF or EKF
$\boldsymbol{x}_{n+1} = f_n(\boldsymbol{x}_n) + u_n$
$\boldsymbol{y}_n = h_n(\boldsymbol{x}_n) + v_n$
Estimated gene expression ($\hat{\boldsymbol{x}}_n$)
Compute root mean squared error

Figure 1. A schematic layout of the proposed algorithm for estimating the evolution of gene expression using nonlinear Bayesian algorithms. The input is synthetic or real-world biological data, and the output is the estimated gene expression over time.

PF-MCMC, EKF-PF, and UKF-PF. The system noise and the measurement noise are $u \sim \mathcal{N}(0, 10^{-4})$ and $v \sim \mathcal{N}(0, 10^{-4})$, respectively. All the simulation results were performed 100 Monte Carlo runs.

$$A = \begin{bmatrix} 0 & 0 & 0 & 0.6 & 0.7 & 0 & 1.9 & 2.9 \\ -.1 & 0 & 0 & 3.5 & 0 & -2.1 & 0 & 3.4 \\ -4.4 & 0.9 & -1.7 & -0.3 & 3.4 & 0 & 1.7 & 0 \\ 0 & 0.5 & 2.8 & -3.7 & 0.9 & 0 & 0 & -3.1 \\ 0 & 0.2 & 0 & -2.6 & -3.2 & -0.1 & -0.5 & 4 \\ -0.5 & -1.8 & 0 & 3.4 & 1.4 & 1.1 & 0 & -1.7 \\ -0.8 & 0 & 0 & -3 & 1.1 & 0.4 & 0 & 0 \\ -0.3 & 0 & -1 & 0 & 0.1 & 0 & 0 & 2.2 \end{bmatrix}.$$

Figure 2 shows the calculated root-mean-square error (RMSE) between the estimated and true states for all filters. We noticed that UKF-PF and UKF (red and blue lines respectively in Figure 1) are able to properly track the evolution of gene expression over time compared to others filters, in terms of RMSE. We also noted that PF and PF-MCMC produced a large estimation error and thus failed to track all genes, due to the fact that GRN is modulated and formulated as a high-dimensional state-space. Moreover, we have confirmed that the PF has a problem dealing with a high-dimensional state-spaces. In addition, we averaged RMSE for eight genes over 60 time-steps as follows: EKF $= 0.0192$, UKF $= 0.0128$, PF $= 0.0442$, PF-MCMC $= 0.0649$, EKF-PF $= 0.0945$ and UKF-PF $= 0.0076$. Furthermore, we observed that UKF-PF provides the best solution with less RMSE compared to others filters and it has been able to track the evolution of expression gene successfully over time.

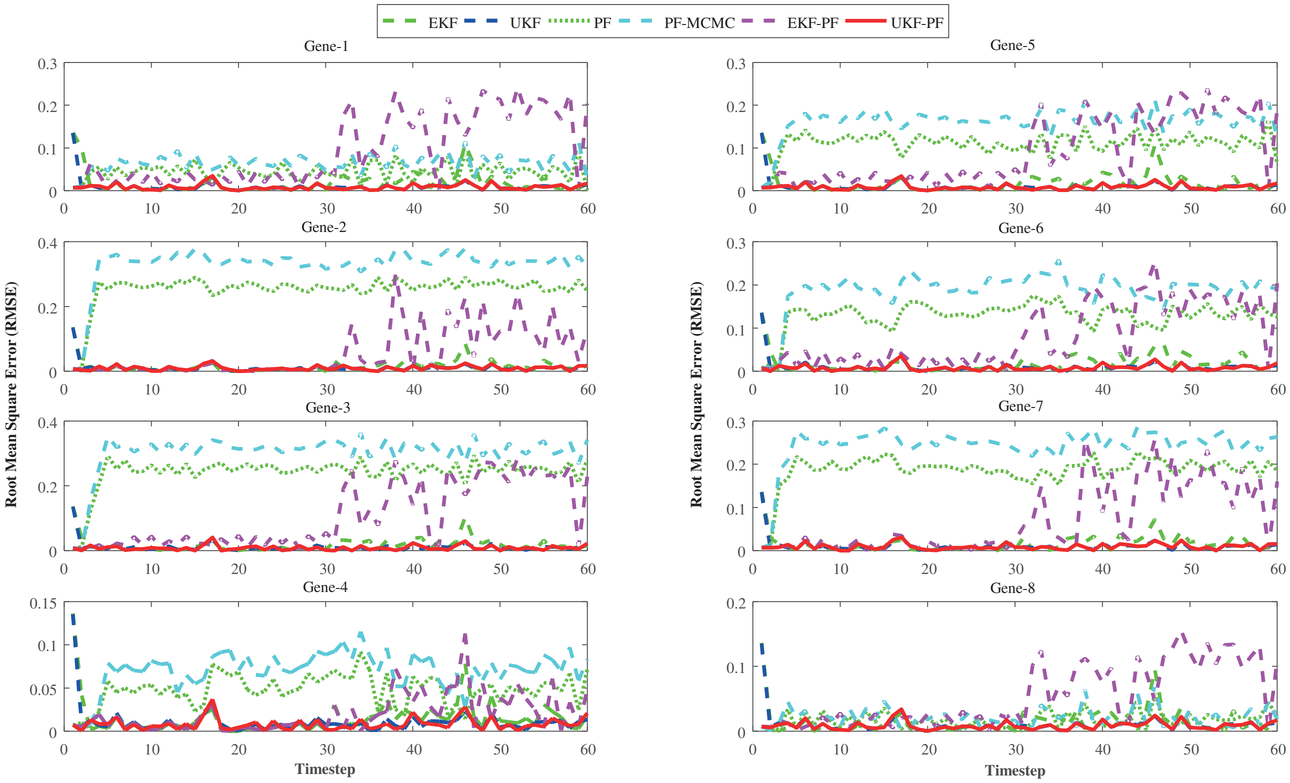

Figure 2. RMSE for estimated gene expression for synthetic data using nonlinear Bayesian filtering algorithms. The averaged RMSE values for eight genes are: EKF$= 0.0192$, UKF$= 0.0128$, PF$= 0.0442$, PF-MCMC$= 0.0649$, EKF-PF$= 0.0945$, and UKF-PF$= 0.0076$.

4.2. Application on real biological data

We have applied the nonlinear Bayesian filtering algorithms to estimate the gene expression from real data. The real data was used for worm time series and Drosophila melanogaster.

4.2.1. Application on the worm time series data

We estimated the evolution of gene expression for the worm time series data using nonlinear Bayesian filtering algorithms. The real data of the worm time series consisting of 98 time points for 123 genes was presented in [18]. We selected the first 8 genes expression for this simulation. The system noise is $u \sim \mathcal{N}(0, 10^{-4})$ and the measurement noise is $v \sim \mathcal{N}(0, 10^{-3})$. Figure 2 shows the evolution of all genes expression. We estimated the evolution of eight genes using 5000 particles for PF, EKF-PF, PF-MCMC, and UKF-PF, and we performed 100 Monte Carlo runs. The coefficients of regulatory relationship matrix A are computed using the normal distribution. We observed that UKF-PF and UKF are able to predict the evolution of 8 genes expression over time better than the other filters as shown in Figure 3. Another observation is that the estimation of the evolution of the eight genes expression are different.

The computed RMSE between the estimated and true state for worm time series data is presented in Figure 4. We averaged RMSE for eight genes: EKF $= 0.2348$, UKF $= 0.0117$, PF $= 0.4412$, PF-MCMC $= 0.4710$, EKF-PF $= 0.3020$, and UKF-PF $= 0.0026$. Moreover, UKF-PF provides better results in terms of RMSE where PF, PF-MCMC, and EKF-PF lead to a large estimation error.

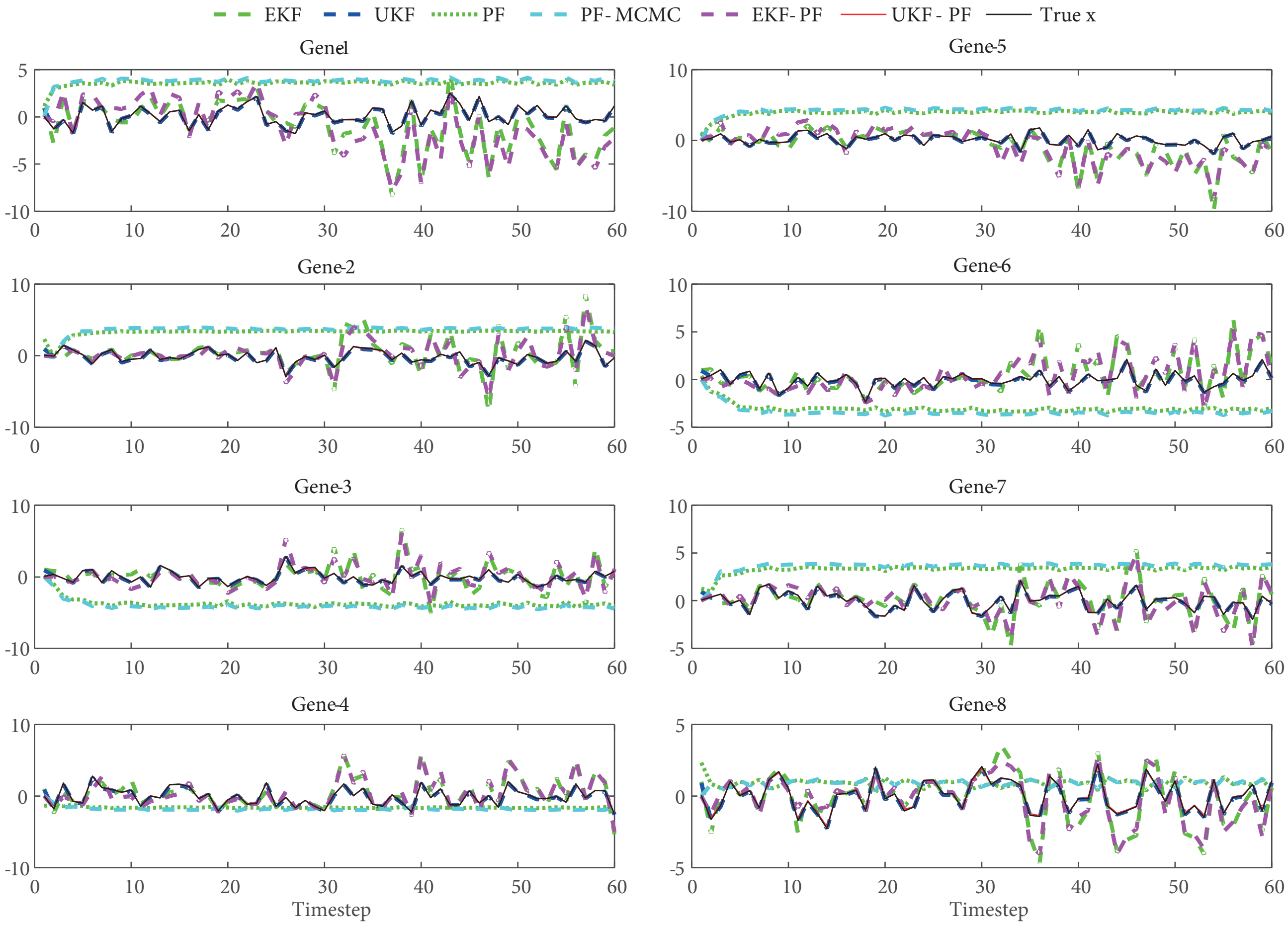

Figure 3. Estimated gene expression for worm time series data using nonlinear Bayesian filtering algorithms.

4.2.2. Application on the Drosophila melanogaster time series data

We tracked the evolution of gene expression of the Drosophila melanogaster dataset using nonlinear Bayesian filtering algorithms. The Drosophila melanogaster time series dataset consisting of 63 time points for 13955 genes was provided in [19]. The dataset is for temporal changes in gene expression during the adult aging process of Drosophila melanogaster. This object is chosen for the purpose of simulation because of its well-explained and described genomes as well as the large amount of the experimental data available. We selected the first 8 genes expression. We used the same settings for the algorithms that were provided in the Subsection 4.2.1. Figure 5 shows the evolution of 8 estimated gene expression for Drosophila melanogaster. We noted that UKF-PF and UKF are able to track the evolution of the gene expression compared with the other filters.

Figure 6 illustrates the calculated RMSE between the estimated and true state for the Drosophila melanogaster dataset. The averaged RMSE values for eight genes are: EKF $= 0.1733$, UKF $= 0.0348$, PF $= 0.3093$, PF-MCMC $= 0.3444$, EKF-PF $= 0.1710$, and UKF-PF $= 0.0036$. Furthermore, UKF-PF provides better results in terms of RMSE, where PF, PF-MCMC, and EKF-PF yields an erroneous estimation for all 8 estimated gene expression. Therefore, the UKF-PF algorithm introduces a powerful solution to the problem of modeling gene regulatory networks.

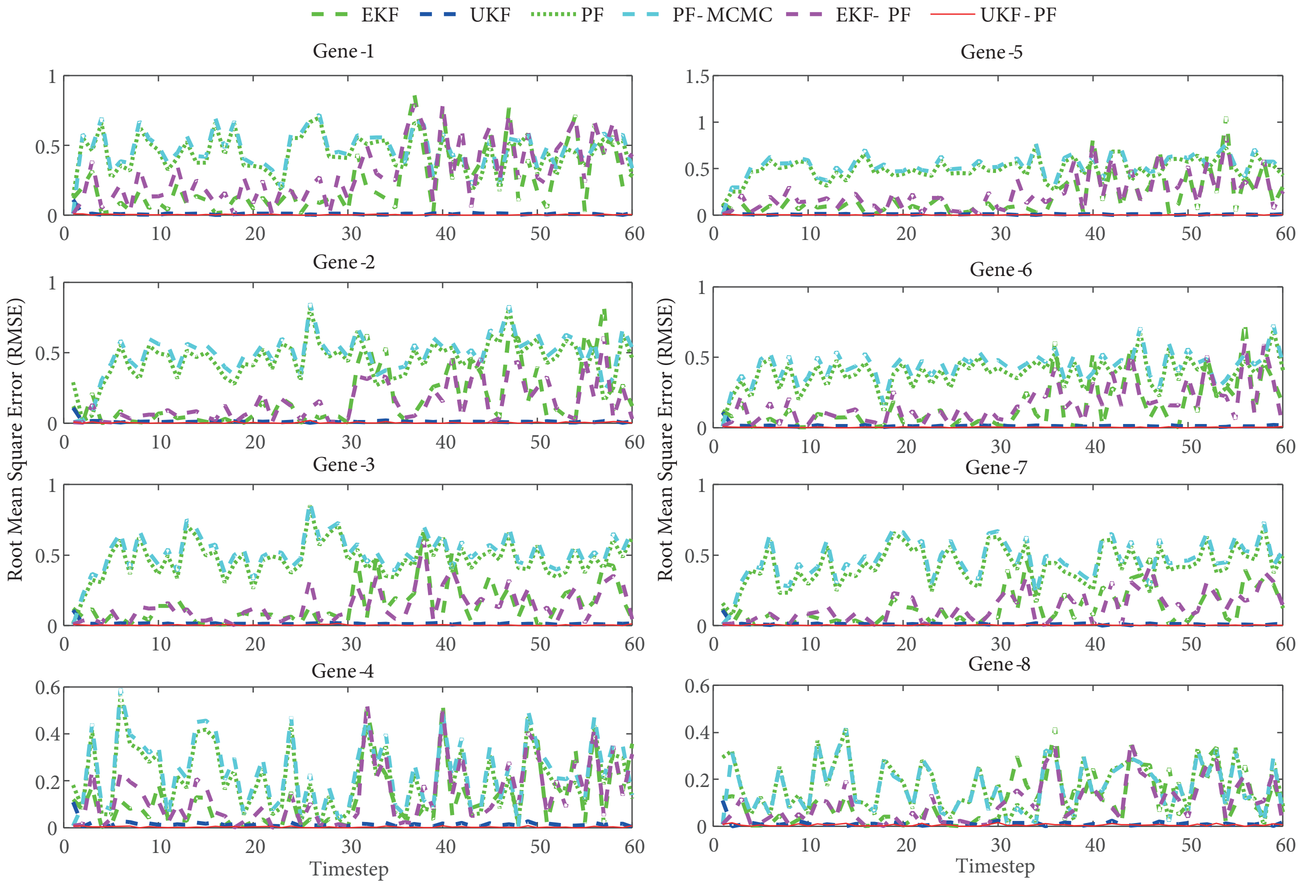

Figure 4. RMSE for estimated gene expression for worm time series data using nonlinear Bayesian filtering algorithms. The averaged RMSE values for eight genes are: EKF= 0.2348, UKF= 0.0117, PF= 0.4412, PF-MCMC= 0.4710, EKF-PF= 0.3020, and UKF-PF= 0.0026.

4.3. Statistical analysis

The ANOVA is a technique used to evaluate whether there is any statistically significant difference between two or more independent groups [20]. A repeated measures ANOVA test followed by a post hoc Bonferroni multiple comparisons test (with an alpha level significance value of 0.05) was performed for all applications. These tests were used to compare the differences in the estimation errors between the results obtained by all algorithms and to determine which algorithm was different from the others.

Table 1 shows the analysis report of the repeated measures ANOVA of the estimation errors for the predicted gene expression of the synthetic data, worm time series data, and Drosophila melanogaster dataset. The repeated measures ANOVA test uses the F-statistic ratio to define whether a significant difference exists among mean responses for interactions between factors. A P-value less than 0.05 means that differences between column means are significant. We noted that P = 0.000 is less than 0.05; thus, the null hypothesis is rejected, and the differences between the estimation errors of EKF, UKF, PF, PF-MCMC, EKF-PF, and UKF-PF, were statistically significant for the three applications.

Paired sample t-tests with Bonferroni correction shows that the UKF-PF was statistically significantly different from EKF, PF, PF-MCMC, and EKF-PF for the synthetic data, worm data, and Drosophila melanogaster

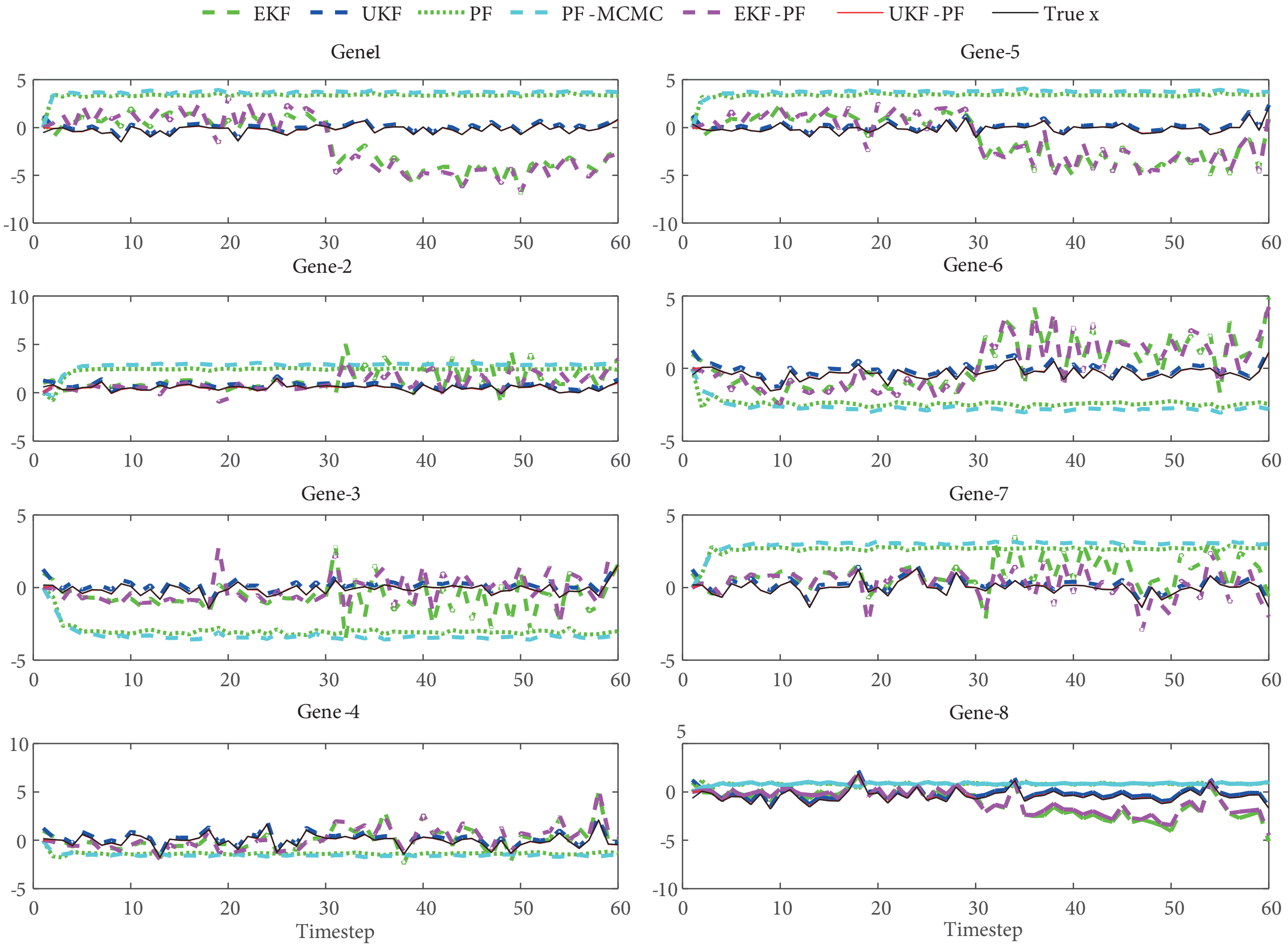

Figure 5. Estimated gene expression for Drosophila melanogaster using nonlinear Bayesian filtering algorithms.

Table 1. Repeated measures ANOVA: Tests of between-algorithms effects for the three applications.

Application	F	P-value
Synthetic data	315.576	0.000
Worm data	300.807	0.000
Drosophila melanogaster data	889.772	0.000

data as shown in Tables 2–4, respectively. We also noted that the UKF-PF and UKF were statistically similar (P = 1). However, we observed that there is a minor difference in the mean error between UKF-PF and UKF for the three applications as shown in Tables 2–4, which confirms that the UKF-PF provides the best results for estimating the evolution of gene expression time series data.

5. Conclusion

This paper addressed the problem of accurate estimation of gene expression using nonlinear Bayesian filtering algorithms. We introduced a widely comparative study of nonlinear Bayesian filters for nonlinear and non-

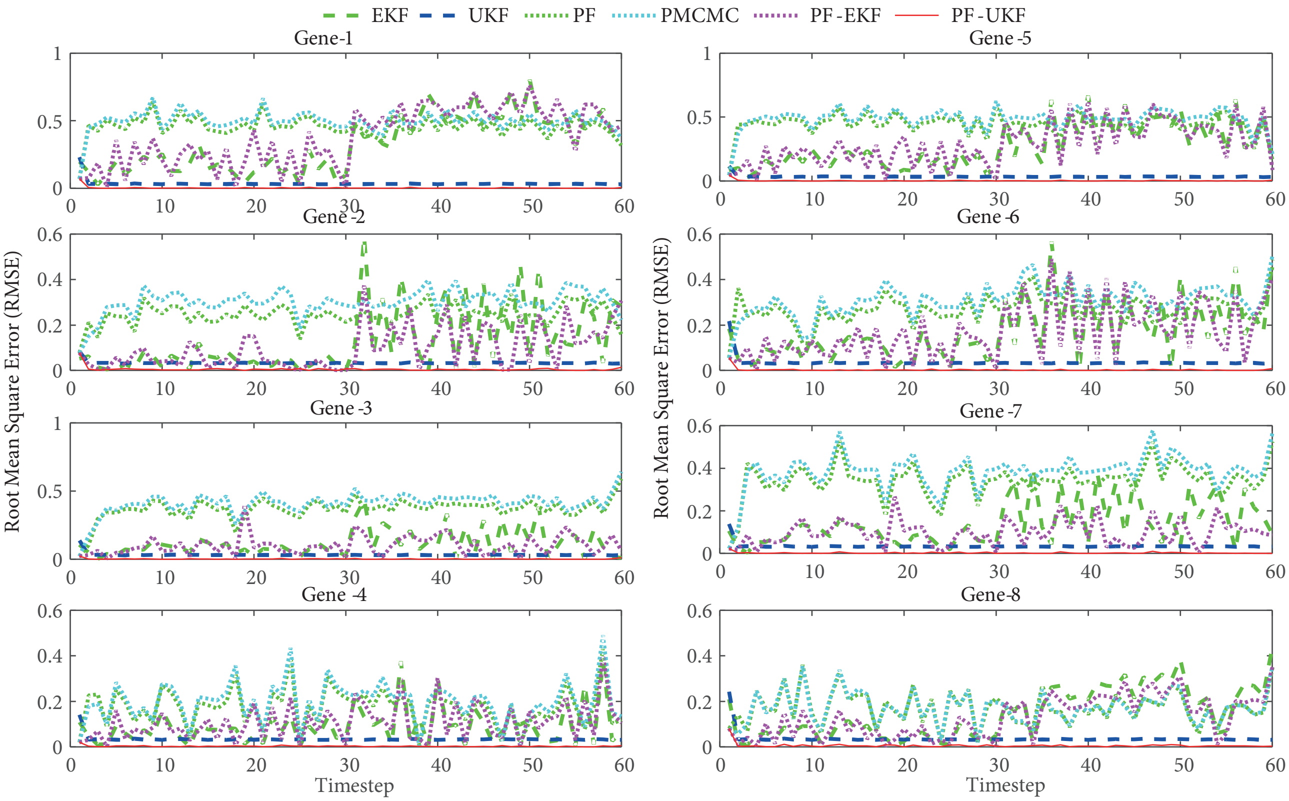

Figure 6. RMSE for estimated gene expression for Drosophila melanogaster using nonlinear Bayesian filtering algorithms. The averaged RMSE values for eight genes are: EKF= 0.1733, UKF= 0.0348, PF= 0.3093, PF-MCMC= 0.3444, EKF-PF= 0.1710 and UKF-PF= 0.0036.

Table 2. Pairwise comparisons of all algorithms estimation error for the synthetic application.

Algorithms	Mean difference	Std. error	P-value^b	Lower bound	Upper bound
UKF-PF & EKF	−0.013*	0.003	0.006	−0.023	−0.002
UKF-PF & UKF	−0.003	0.002	1.000	−0.010	0.004
UKF-PF & PF	−0.101*	0.004	0.000	−0.113	−0.089
UKF-PF & PF-MCMC	−0.151*	0.005	0.000	−0.165	−0.136
UKF-PF & EKF-PF	−0.073*	0.010	0.000	−0.104	−0.042

* The mean difference is significant at the 0.05 level.
^{b} Adjustment for multiple comparisons: Bonferroni.

Gaussian state-space systems. The simulation results on synthetic and real data showed that the UKF-PF was able to estimate the evolution of gene expression from time series data with minimum RMSE compared to other filters, i.e. EKF, UKF, PF, PF-MCMC, and EKF-PF. We showed that the UKF-PF outperformed ($P < 0.05$) for all applications and had a higher tracking accuracy as compared to others nonlinear Bayesian filtering algorithms. In conclusion, the UKF-PF presented a robust and effective algorithm for estimating of time series of gene expression, thus providing an alternative solution to the problem of modeling gene regulatory networks.

Table 3. Pairwise comparisons of all algorithms estimation error for the worm application.

Algorithms	Mean difference	Std. error	P-value^b	Lower bound	Upper bound
UKF-PF & EKF	-0.233^*	0.029	0.000	-0.322	-0.145
UKF-PF & UKF	-0.016^*	0.002	0.000	-0.022	-0.011
UKF-PF & PF	-0.416^*	0.019	0.000	-0.475	-0.357
UKF-PF & PF-MCMC	-0.450^*	0.019	0.000	-0.508	-0.392
UKF-PF & EKF-PF	-0.286^*	0.029	0.000	-0.374	-0.199

* The mean difference is significant at the .05 level. ^{b}Adjustment for multiple comparisons: Bonferroni.

Table 4. Pairwise comparisons of all algorithms estimation error for the Drosophila melanogaster application.

Algorithms	Mean difference	Std. error	P-value^b	Lower bound	Upper bound
UKF-PF & EKF	-0.325^*	0.026	0.000	-0.405	-0.245
UKF-PF & UKF	-0.003	0.002	1.000	-0.0009	0.004
UKF-PF & PF	-0.455^*	0.011	0.000	-0.488	-0.422
UKF-PF & PF-MCMC	-0.493^*	0.011	0.000	-0.527	-0.458
UKF-PF & EKF-PF	-0.367^*	0.028	0.000	-0.452	-0.282

*The mean difference is significant at the .05 level. ^{b}Adjustment for multiple comparisons: Bonferroni.

References

[1] Wang Z, Liu X, Liu Y, Liang J, Vinciotti V. 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 2009; 6: 410-419.

[2] Bugallo MF, Tasdemir C, Djuric PM. Estimation of gene expression by a bank of particle filters. In: IEEE 2015 23rd European Signal Processing Conference; Nice, France; 2015. pp. 494-498.

[3] Noor A, Serpedin E, Nounou M, Nounou H. Inferring gene regulatory networks via nonlinear state-space models and exploiting sparsity. IEEE/ACM Transactions on Computational Biology and Bioinformatics 2012; 9: 1203-1211.

[4] Fan Y, Wang X, Peng Q. Inference of gene regulatory networks using bayesian nonparametric regression and topology information. Computational and Mathematical Methods in Medicine 2017: 1-8. doi: 10.1155/2017/8307530

[5] Julier SJ, Uhlmann JK, Durrant-Whyte H F. A new method for the nonlinear transformation of means and covariances in filters and estimators. IEEE Transactions on Automatic Control 2000; 45: 477-482.

[6] Julier SJ, Uhlmann JK. Unscented filtering and nonlinear estimation. Proceedings of the IEEE 2004; 92: 401-422.

[7] Doucet A, Johansen A. A tutorial on particle filtering and smoothing: fifteen years later. In: Crisan D, Rozovsky B (editors). Handbook of Nonlinear Filtering, Oxford: Oxford University Press, 2011.

[8] Andrieu C, Doucet A, Holenstein R. Particle markov chain monte carlo methods. Journal of the Royal Statistical Society 2010; 72: 269-342.

[9] Van Der Merwe R, Doucet A, De Freitas N, Wan E. The unscented particle filter. In: Proceedings of the 13th International Conference on Neural Information Processing Systems; Denver, CO, USA; 2000. pp. 563-569.

[10] Amor N, Kahlaoui S, Chebbi S. Unscented particle filter using student-t distribution with non-Gaussian measurement noise. In: IEEE IC-ASET 2018- International Conference on Advanced Systems and Electric Technologies; Hammamet, Tunisia; 2018. pp. 34-38.

[11] Julier SJ, Uhlmann JK. New extension of the kalman filter to nonlinear systems. In: Signal processing, sensor fusion, and target recognition VI, International Society for Optics and Photonics; Orlando, FL, USA; 1997. pp. 182-194.

[12] Terejanu GA. Unscented kalman filter tutorial. In: Workshop on Large-Scale Quantification of Uncertainty; Sandia National Laboratories; Livermore, CA, USA; 2009. pp. 1-6.

[13] Gordon NJ, Salmond DJ, Smith AFM. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEEE Proceedings in Radar and Signal Processing 1993; 140: 107-113.

[14] Crisan D, Doucet A. A survery of convergence results on particle filtering methods for practitioners. Transaction on Signal Processing 2002; 50: 736-746.

[15] Särkkä S. Bayesian Filtering and Smoonthing. Cambridge, England, UK: Cambridge University Press, 2013.

[16] Arulampalam M S, Maskell S, Gordon N, Clapp T. A tutorial on particle filters for online nonlinear/non-Gaussian bayesian tracking. IEEE Transactions on Signal Processing 2002; 50: 174-188.

[17] Doucet A, Godsill S, Andrieu C. On sequential monte carlo sampling methods for bayesian filtering. Statistics and Computing 2000; 10: 197-208.

[18] Maduro MF, Rothman JH. Making worm guts: the gene regulatory network of the caenorhabditis elegans endoderm. Developmental Biology Journal 2002; 246 : 68-85.

[19] Carlson KA, Zhang C, Harshman LG. A dataset for assessing temporal changes in gene expression during the aging process of adult Drosophila melanogaster. Data in Brief 2016; 7: 1652-1657.

[20] Stevens JP. Applied Multivariate Statistics for the Social Sciences. 5th ed. New York, NY, USA: Routledge Taylor & Francis Group, 2009.