Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Identification of Nonlinear Processes with known Model Structure Under Missing Observations

R.B. Gopaluni *

* *Department of Chemical and Biological Engineering, University of British Columbia, Vancouver, Canada V6T 1Z3 (Tel: 604 827 5668; e-mail: gopaluni@chml.ubc.ca).*

Abstract: A novel maximum likelihood solution to the problem of identifying parameters of a nonlinear model under missing observations is presented. An expectation maximization (EM) algorithm, which uses the expected value of the complete log-likelihood function including the missing observations, is developed. The expected value of the complete log-likelihood (E-step) in the EM algorithm is approximated using particle filters and smoothers. New expressions for particle filters and smoothers under missing observations are derived. The maximization step (M-step) in the EM algorithm is performed using standard optimization routines. The above nonlinear identification approach is illustrated through numerical examples.

1. INTRODUCTION

Many chemical processes can be modeled using nonlinear stochastic differential equations arising from fundamental physical laws. These differential equations are usually continuous in time, and various approaches exist for their discretization (Yuz and Goodwin [2005]). For instance, the dynamics of polymerization and other chemical reactors, which are highly nonlinear, can be discretized and represented by the following general stochastic state space model: $x_{t+1} = f(x_t, u_t, \theta) + w_t$, $y_t = g(x_t, u_t, \theta) + v_t$, where $x_t \in \mathbf{R}^n$ is the n-dimensional state vector, $u_t \in \mathbf{R}^s$ is the s-dimensional input vector, $y_t \in \mathbf{R}^m$ is the m-dimensional output or measurement vector, and w_t, v_t are independent and identically distributed Gaussian noise sequences of appropriate dimension, $\theta \in \mathbf{R}^p$ is a p-dimensional parameter vector and $f(.)$, $g(.)$ are known nonlinear functions. A good model - in other words a good estimate of θ in the above model structure - is required for state estimation, control, performance monitoring and assessment, fault detection and diagnosis of such processes. This article focusses on parameter estimation of models of the form presented above under missing observations by combining expectation maximization algorithm with particle filters.

Expectation maximization (EM) is a standard algorithm for parameter estimation in state space models (Shumway and Stoffer [2000]). It involves two steps, where one estimates the joint probability density of the states and the observations based on an initial estimate of parameters in the first step, and maximizes the expected value of the joint density in the second step to obtain a new estimate of the parameter vector (Dempster et al. [1977]). These two steps are repeated until the change in parameters after each iteration is within a specified tolerance level. For linear systems with Gaussian noise, the expectation and maximization steps in the algorithm can be solved analytically, and explicit recursive equations for parameter estimation can be developed (Shumway and Stoffer [2000]). On the other hand, for most nonlinear state space models with Gaussian or non-Gaussian noise, the expectation and maximization steps do not have explicit solutions.

A number of approximations of the expectation and maximization steps have been proposed in the literature for nonlinear processes. In Roweis and Ghahramani [2001] an extended Kalman filter is used to approximate the filtered states and the expected value of the complete likelihood function. In Goodwin and Agüero [2005] a similar linearized approximation of the process around a maximum *a posteriori* estimate of the state vector is used. In Doucet et al. [2001], Poyiadjis et al. [2005], Schön et al. [2006] a particle approximation of the expectation step is used. While linearization methods fail to perform well if the nonlinearities are strong, particle filter approaches require large number of particles for good approximation of the expected likelihood function (Andrieu et al. [2004]). Most particle based approaches use a state-path based density function to approximate the likelihood function *i.e.,* a density function of the form $p(x_1, x_2, \cdots, x_T)$ (Andrieu et al. [2004]). It is known that the variance of path-based density functions increases rapidly with the data length, T (Andrieu et al. [2004], Poyiadjis et al. [2005]). An approximation based on point-wise state density functions for affine parameter models is presented in Schön et al. [2006]. In this paper, a new approximation, based on point-wise state density functions of the form $p(x_t)$, for non-affine parametric models is proposed and extended to handle missing data in the observations.

The second step in the EM algorithm involves maximization of the expected complete likelihood function with respect to the parameter vector. For most nonlinear processes, this maximization step does not have an explicit solution. However, in some special cases, such as bilinear models (Gibson et al. [2005]) or processes defined by radial basis functions (Roweis and Ghahramani [2001]), it is possible to find an explicit solution. Depending on the structure of the nonlinear process, any of the above mentioned approaches or any standard optimization approach,

can be used with the method proposed in this article.

Missing observations are commonplace in the chemical industry. For linear systems, EM algorithm has been adapted to handle missing observations (Shumway and Stoffer [2000], Isaksson [1993]), and also applied in practice (Raghavan et al. [2006]). Other approaches for linear systems based on lifting techniques (Li et al. [2003]) and continuous time identification (Wang and Gawthrop [2001]) have also been reported. While the importance of estimating nonlinear processes under missing observations has long been recognized (Gudi et al. [1995], Tatiraju et al. [1999]), to the best knowledge of the author, no work has been reported for nonlinear stochastic processes. The published work on parameter estimation for nonlinear systems treats only states as missing data. In this paper the EM algorithm is adapted to also handle nonlinear processes with missing observations.

2. EXPECTATION MAXIMIZATION ALGORITHM

Expectation Maximization is an elegant optimization algorithm that constructs a complete likelihood function including the hidden states and missing observations, and maximizes the likelihood function of observed data through iterations. A brief description of the EM algorithm is presented in this section to facilitate the development of the proposed algorithm in later sections.

For the state-space model described in this article, let $p(y_{1:T}|\theta)$ [1] denote the joint likelihood function of the observed data. The maximum likelihood estimate of the parameter vector is obtained by maximizing this observed data likelihood function. For certain classes of state-space models, such as linear systems, it is possible to derive an explicit expression for this joint density. However, for the model considered in this paper, it is difficult to develop such an expression. Instead, using the Markov property of the model it is straightforward to develop an expression for the complete (including states and observations) likelihood function, $p(x_{1:T}, y_{1:T}|\theta)$. In light of this feature of the Markovian state-space model, the joint probability density function of the states and observations is iteratively maximized to obtain a maximizing θ for $p(y_{1:T}|\theta)$. This maximization approach is called EM algorithm and can be summarized in four steps: (1). Choose an initial guess of the parameter vector $\theta_0 \in \Omega$. (2). Estimate the states given the parameter vector and the observations and evaluate $Q(\theta_i, \theta) = \int \log[p(x_{1:T}, y_{1:T}|y_{1:T}, \theta)]p(x_{1:T}|y_{1:T}, \theta_i)dx_{1:T}$. (3). Maximize $Q(\theta_i, \theta)$ with respect to θ. Call the maximizing value θ_{i+1}. (4). Repeat the above two steps until the change in parameter vector is within a specified tolerance level. The second step in the above algorithm is called E-step and the third step is called M-step. The likelihood function increases monotonically through these iterations.

3. THE Q FUNCTION

In this section an approximation of the Q function that is free from the dimensionality problems explained earlier, and that can handle missing observations, is developed using the Markovian property of the state-space model.

[1] $y_{1:T}$ denotes the set $\{y_1, \cdots, y_T\}$.

3.1 Full Data Case

In the rest of this article, it is assumed that the inputs are known and all the density functions of the form $p(.|., ., u_{1:T})$ are denoted by $p(.|., .)$ without explicitly showing the input dependence. Consider the case where all the observations $\{y_1, \cdots, y_T\}$ and the inputs $\{u_1, \cdots, u_T\}$ are available. Then, using the Markov property of the state space model, the joint density function of states and outputs can be written as

$$p(x_{1:T}, y_{1:T}|y_{1:T}, \theta) = \\ = p(x_1|y_{1:T}, \theta) \prod_{t=2}^{T} p(x_t|x_{t-1}, \theta) \prod_{t=1}^{T} p(y_t|x_t, \theta)$$

Performing the integrations in the expression for Q, the following form of Q function can be obtained

$$\begin{aligned} Q(\theta_i, \theta) = &\int \log[p(x_1|y_{1:T}, \theta)]p(x_1|y_{1:T}, \theta_i)dx_1 \\ &+ \sum_{t=2}^{T} \int \log[p(x_t|x_{t-1}, \theta)]p(x_{t-1:t}|y_{1:T}, \theta_i)dx_{t-1:t} \\ &+ \sum_{t=1}^{T} \int \log[p(y_t|x_t, \theta)]p(x_t|y_{1:T}, \theta_i)dx_t. \end{aligned} \quad (1)$$

From the above expression, notice that approximations of the density functions $p(x_1|y_{1:T}, \theta_i)$, $p(x_{t-1:t}|y_{1:T}, \theta_i)$, $p(x_t|y_{1:T}, \theta_i)$ would allow one to approximate the Q function.

3.2 Missing Data in Output

Suppose that only a portion of the output measurements at time instants $\{t_1, \cdots, t_\gamma\}$ are available and that they are not available at time instants $\{s_1, \cdots, s_\beta\}$. In other words only $\{y_{t_1}, \cdots, y_{t_\gamma}\}$ out of $\{y_1, \cdots, y_T\}$ are available. For notational simplicity, it is also assumed that y_1 and y_T are available. Then the Q function can be written as

$$\begin{aligned} Q(\theta_i, \theta) = &\int \log[p(x_1|y_{t_1:t_\gamma}, \theta)]p(x_1|y_{t_1:t_\gamma}, \theta_i)dx_1 \\ &+ \sum_{t=2}^{T} \int \log[p(x_t|x_{t-1}, \theta)]p(x_{t-1:t}|y_{t_1:t_\gamma}, \theta_i)dx_{t-1:t} \\ &+ \sum_{t=t_1}^{t_\gamma} \int \log[p(y_t|x_t, \theta)]p(x_t|y_{t_1:t_\gamma}, \theta_i)dx_t \\ &+ \sum_{t=s_1}^{s_\beta} \int \log[p(y_t|x_t, \theta)]p(x_t, y_t|y_{t_1:t_\gamma}, \theta_i)dx_t dy_t \end{aligned} \quad (2)$$

In order to approximate the Q functions, approximations of the following density functions are required: **Full data case** - $p(x_t|y_{1:T}, \theta)$, $p(x_{t-1}, x_t|y_{1:T}, \theta)$. **Missing data case** - $p(x_t|y_{t_1:t_\gamma}, \theta)$, $p(x_{t-1}, x_t|y_{t_1:t_\gamma}, \theta)$, $p(x_t, y_t|y_{t_1:t_\gamma}, \theta)$ for $t \notin \{t_1, \cdots, t_\gamma\}$. Notice that the maximum dimensionality of the above density functions is $max(2n, n+m)$, and hence the accuracy of these density functions does not deteriorate with increase in the size of available measurements as is the case with the method suggested in Andrieu et al. [2004].

4. BAYESIAN FILTERING AND SMOOTHING

In this section, Bayesian algorithms to generate approximations of the above density functions are presented.

4.1 Filtering

Full Data The density function of the states given the past and current outputs, $p(x_t|y_{1:t},\theta)$ is called a filter. Applying Bayes' rule in a straightforward manner, one can derive recursive expressions for the density function of the filter. The following predictor density function can be derived using Bayes' rule,

$$p(x_t|y_{1:t-1},\theta) = \int p(x_t|x_{t-1},\theta)p(x_{t-1}|y_{1:t-1},\theta)dx_{t-1} \quad (3)$$

Now using the predictor, one can write the following expression for the filter,

$$p(x_t|y_{1:t},\theta) = \frac{p(y_t|x_t,\theta)p(x_t|y_{1:t-1},\theta)}{\int p(y_t|x_t,\theta)p(x_t|y_{1:t-1},\theta)dx_t} \quad (4)$$

The filter density can be evaluated recursively by substituting (3) in (4). The above integrals needed to estimate the filter density are often intractable, and need to be approximated. Although numerous approximations are available, in this paper a particle filter approach is used. The basic idea behind particle filters is to approximate a density function using dirac-delta functions. The filter density at $t-1$, could be approximated as

$$p(x_{t-1}|y_{1:t-1},\theta) = \sum_{i=1}^{N} w_{t-1|t-1}^{(i)}\delta(x_{t-1} - x_{t-1}^{(i)}) \quad (5)$$

where $w_{t-1|t-1}^{(i)}$ are weights proportional to the filter density at $x_{t-1}^{(i)}$ and $\delta(.)$ is a dirac-delta function. Substituting (5) in (3), an approximation of the predictor can be obtained as follows,

$$p(x_t|y_{1:t-1},\theta) = \sum_{j=1}^{N} p(x_t|x_{t-1}^{(j)},\theta)w_{t-1|t-1}^{(j)} \quad (6)$$

Similarly, substituting (6) in (4), one can approximate the filter density function (Poyiadjis et al. [2005])

$$p(x_t|y_{1:t},\theta) = \sum_{i=1}^{N} w_{t|t}^{(i)}\delta(x_t - x_t^{(i)}) \quad (7)$$

where $x_t^{(i)}$ are chosen from an importance sampling function $p(x_t|y_{1:t-1},\theta)$, and therefore weights are given by

$$w_{t|t}^{(i)} = \frac{p(y_t|x_t^{(i)},\theta)}{\sum_{j=1}^{N} p(y_t|x_t^{(j)},\theta)} \quad (8)$$

Particle Filter Algorithm - Full Data: (1). ***Initialization***: Generate N samples of the initial state x_1 from an initial distribution, $p(x_1)$. Set $w_{1|1}^{(i)} = \frac{1}{N}$ for $i \in \{1,\cdots,N\}$. Set t = 2. (2). ***Prediction***: Sample N values of x_t from the distributions $p(x_t|x_{t-1}^{(i)},\theta)$ for each i. (3). ***Update***: Using (8), find the weights of filter density, $w_{t|t}^{(i)}$. (4). ***Resampling***: Resample N particles from the set $\{x_t^{(1)},\cdots,x_t^{(N)}\}$ with the probability of picking $x_t^{(i)}$ being $w_{t|t}^{(i)}$. Assign $w_{t|t}^{(i)} = \frac{1}{N}$ for all i. (5). Set $t = t+1$. Repeat the above steps (2), (3), and (4) for $t \le T$.

Missing Data For the missing data case, the prediction equation is used recursively until an observation is available. Once an observation is available, the update equation is also used. If an observation is not available at time t, then the following filter equation (ideally it should be called a predictor since no observation is available at time t. However, to be consistent with the full data case, it is called a filter) is used

$$p(x_t|y_{t_1:t_\alpha},\theta) = \int p(x_t|x_{t-1},\theta)\cdots p(x_{t_\alpha+1}|x_{t_\alpha},\theta)p(x_{t_\alpha}|y_{t_1:t_\alpha},\theta)dx_{t_\alpha:t-1} \quad (9)$$

where t_α is the last observation available up to time t. Now assuming that the following approximation of the filter at time t_α is available, $p(x_{t_\alpha}|y_{t_1:t_\alpha},\theta) = \sum_{i=1}^{N} \bar{w}_{t_\alpha|t_\alpha}\delta(x_{t_\alpha} - x_{t_\alpha}^{(i)})$ one can write an approximation of (9) at time t, as $p(x_t|y_{t_1:t_\alpha},\theta) = \sum_{i=1}^{N} p(x_t|x_{t-1}^{(i)},\theta)\cdots p(x_{t_{\alpha+1}}^{(i)}|x_{t_\alpha}^{(i)},\theta)\bar{w}_{t_\alpha|t_\alpha}$ which can be represented using particles as follows

$$p(x_t|y_{t_1:t_\alpha},\theta) = \sum_{i=1}^{N} \bar{w}_{t|t}^{(i)}\delta(x_t - x_t^{(i)}) \quad (10)$$

where $\bar{w}_{t|t}^{(i)} = \frac{1}{N}$, and if an observation is available at time t, then the filtered density is

$$p(x_t|y_{1:t},\theta) = \sum_{i=1}^{N} \bar{w}_{t|t}^{(i)}\delta(x_t - x_t^{(i)}) \quad (11)$$

where $\bar{w}_{t|t}^{(i)}$ are given by (8) and $x_t^{(i)}$ are drawn from the density function $p(x_t|y_{1:t_\alpha},\theta)$.

Particle Filter Algorithm - Missing Data: (1). ***Initialization***: Generate N samples of the initial state x_1 from an initial distribution, $p(x_1)$. Set $\bar{w}_{1|1}^{(i)} = \frac{1}{N}$ for $i \in \{1,\cdots,N\}$. (2). ***Prediction***: Sample N values of x_t from the distributions $p(x_t|x_{t-1}^{(i)},\theta)$ for each i. (3). ***Update***: If y_t is available, using (8), find the weights of filter density, $\bar{w}_{t|t}^{(i)}$. If not, use the density in (10) as the filtered density. (4). ***Resampling***: Resample N particles from the set $\{x_t^{(1)},\cdots,x_t^{(N)}\}$ with the probability of picking $x_t^{(i)}$ being $\bar{w}_{t|t}^{(i)}$. Assign $\bar{w}_{t|t}^{(i)} = \frac{1}{N}$ for all i. (5).Set $t = t+1$. Repeat the above steps (2), (3), and (4) for $t \le T$.

4.2 Smoothing

Full Data The density function of a state given the past and future observations is called a smoother. In the above expressions for the Q-function, $p(x_t|y_{1:T},\theta)$ is the smoothed density functions of the states. There are many approaches to estimate the density function of the

smoothed states (Klaas et al. [2006]). A forward-backward smoother algorithm, which is explained below, is used in this paper. The smoothed density can be factored as

$$p(x_t|y_{1:T},\theta) = \int p(x_{t+1}|y_{1:T},\theta)p(x_t|x_{t+1},y_{1:t},\theta)dx_{t+1}$$
$$= p(x_t|y_{1:t},\theta)\int \frac{p(x_{t+1}|y_{1:T},\theta)p(x_{t+1}|x_t,\theta)}{\int p(x_{t+1}|x_t,\theta)p(x_t|y_{1:t},\theta)dx_t}dx_{t+1} \quad (12)$$

Hence the smoothed density function can be obtained as a function of filtered state density at time t, smoothed density at $t+1$ and the state prediction density $p(x_{t+1}|x_t,\theta)$. Clearly, this approach to smoothing involves a forward filtering step and a backward smoothing step. Assuming that the smoothed density function at time t, can be approximated using the following particle approximation $p(x_t|y_{1:T},\theta) = \sum_{i=1}^{N} w_{t|T}^{(i)}\delta(x_t - x_t^{(i)})$ one can derive the following recursive particle approximation of the smoothed density function Klaas et al. [2006],

$$w_{t|T}^{(i)} = w_{t|t}^{(i)}\left[\sum_{j=1}^{N} w_{t+1|T}^{(j)}\frac{p(x_{t+1}^{(j)}|x_t^{(i)},\theta)}{\sum_{k=1}^{N} w_{t|t}^{(k)}p(x_{t+1}^{(j)}|x_t^{(k)},\theta)}\right] \quad (13)$$

Particle Smoother Algorithm - Full Data: (1). ***Filtering***: For $t = 1$ to $t = T$ perform filtering according to the algorithm in the previous section and obtain weights $w_{t|t}^{(i)}$ for all i. (2). ***Initialization***: Initialize the smoother weights at $t = T$ to $w_{T|T}^{(i)} = \frac{1}{N}$ for all i. (3). ***Smoothing***: Find the smoothed weights recursively using (13).

Missing Data It is easy to see that the backward recursion of the smoothing algorithm does not depend on the observations while the forward recursion depends on the observations. Therefore, the only modification needed in the smoothing algorithm is usage of missing data weights of the filtering density. The algorithm is summarized for the sake of completeness.

Particle Smoother Algorithm - Missing Data: (1). ***Filtering***: For $t = 1$ to $t = T$ perform filtering according to the algorithm in the previous section and obtain weights $\bar{w}_{t|t}^{(i)}$ for all i. (2). ***Initialization***: Initialize the smoother weights at $t = T$ to $\bar{w}_{T|T}^{(i)} = \frac{1}{N}$. (3). ***Smoothing***: Find the smoothed weights using (13) by replacing $w_{t|t}^{(i)}$ by $\bar{w}_{t|t}^{(i)}$.

4.3 Joint Distribution of x_t, x_{t+1}

The joint density function between x_t and x_{t-1} can be obtained by using (12)

$$p(x_t, x_{t+1}|y_{1:T},\theta) =$$
$$= p(x_t|y_{1:t},\theta)\frac{p(x_{t+1}|y_{1:T},\theta)p(x_{t+1}|x_t,\theta)}{\int p(x_{t+1}|x_t,\theta)p(x_t|y_{1:t},\theta)dx_t} \quad (14)$$

Substituting particle approximations of $p(x_t|y_{1:t},\theta)$ and $p(x_{t+1}|y_{1:T},\theta)$ in (14), the following approximation of the joint distribution can be obtained, $p(x_t, x_{t+1}|y_{1:T},\theta) = \sum_{j=1}^{N}\sum_{i=1}^{N} w_{t,t+1}^{(ij)}\delta(x_t - x_t^{(i)})\delta(x_{t+1} - x_{t+1}^{(j)})$ where $w_{t,t+1}^{(ij)} = w_{t|t}^{(i)}w_{t+1|T}^{(j)}\frac{p(x_{t+1}^{(j)}|x_t^{(i)},\theta)}{\sum_{k=1}^{N} w_{t|t}^{(k)}p(x_{t+1}^{(j)}|x_t^{(k)},\theta)}$. The above approximation is found to be computationally very expensive, and hence it is replaced with the following approximation that has fewer particles, $p(x_t, x_{t+1}|y_{1:T},\theta) = \sum_{i=1}^{N} w_{t,t+1}^{(i)}\delta(x_t - x_t^{(i)})\delta(x_{t+1} - x_{t+1}^{(i)})$. where $w_{t,t+1}^{(i)} = \frac{\eta_t^{(i)}}{\sum_{i=1}^{N}\eta_t^{(i)}}$, and $\eta_t^{(i)} = w_{t|t}^{(i)}w_{t+1|T}^{(i)}\frac{p(x_{t+1}^{(i)}|x_t^{(i)},\theta)}{\sum_{k=1}^{N} w_{t|t}^{(k)}p(x_{t+1}^{(i)}|x_t^{(k)},\theta)}$.

Similarly, for the missing data case simply replace the weights of the filter in the above equation by those corresponding to missing data.

4.4 Joint Distribution of x_t, y_t

The joint distribution, $p(x_t, y_t|y_{t_1:t_\gamma},\theta)$, is required at sample times where the observations are missing. One can write $p(x_t, y_t|y_{t_1:t_\gamma},\theta) = p(y_t|x_t,\theta)p(x_t|y_{t_1:t_\gamma},\theta)$. In this expression, the first term on the right hand side, $p(y_t|x_t,\theta)$, is the density function of the observations given the state, and the second term, $p(x_t|y_{t_1:t_\gamma},\theta)$, is the smoother. Since y_t is missing, it is possible to obtain an estimate of y_t for a given x_t from the density function $p(y_t|x_t,\theta)$. Therefore, one can obtain the following particle approximation of $p(y_t|x_t,\theta)$

$$p(y_t|x_t,\theta) \sim \sum_{i=1}^{N} p(y_t^{(i)}|x_t^{(i)},\theta)\delta(x_t - x_t^{(i)})\delta(y_t - y_t^{(i)}) \quad (15)$$

It is in place to mention that a better approximation of $p(y_t|x_t,\theta)$ can be obtained by using more than one estimate of y_t for every $x_t^{(i)}$. While this approach increases the accuracy, it comes at an increase in computational burden, and hence the approximation in (15) is used. The required joint distribution can now be approximated as $p(x_t, y_t|y_{t_1:t_\gamma},\theta) = \sum_{i=1}^{N} \bar{w}_{t|x}^{(i)}\delta(x_t - x_t^{(i)})\delta(y_t - y_t^{(i)})$, where $\bar{w}_{t|x}^{(i)} = \frac{p(y_t^{(i)}|x_t^{(i)},\theta)p(x_t^{(i)}|y_{t_1:t_\gamma},\theta)}{\sum_{i=1}^{N} p(y_t^{(i)}|x_t^{(i)},\theta)p(x_t^{(i)}|y_{t_1:t_\gamma},\theta)}$.

5. IDENTIFICATION ALGORITHM

By combining the equations for the filter, smoother and the joint density functions, one can approximate the Q function. Once an approximation of the Q function is available, it is possible to maximize it with respect to the parameter vector and obtain the next iterate of the EM algorithm. The approximate Q function can be written as

$$Q(\theta',\theta) \approx \sum_{i=1}^{N} \bar{w}_{1|1}^{(i)}\log[p(x_1^{(i)}|y_{t_1:t_\gamma},\theta)] + \sum_{t=2}^{T}\sum_{i=1}^{N} \bar{w}_{t,t-1}^{(i)}$$
$$\log[p(x_t^{(i)}|x_{t-1}^{(i)},y_{t_1:t_\gamma},\theta)] + \sum_{t=t_1}^{t_\alpha}\sum_{i=1}^{N} \bar{w}_{t|T}^{(i)}\log[p(y_t|x_t^{(i)},\theta)]$$
$$+ \sum_{t=s_1}^{s_\beta}\sum_{i=1}^{N} \bar{w}_{t|x}^{(i)}\log[p(y_t^{(i)}|x_t^{(i)},\theta)] \quad (16)$$

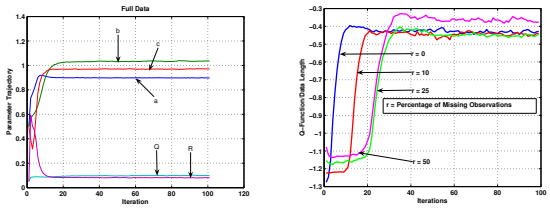
Then the EM algorithm can be summarized as

(0). **Initialization**: Initialize the parameter vector to θ_0. Set $i = 0$. (1).**Expectation**: Evaluate the approximate Q function according to (16) using $\theta' = \theta_i$. (2). **Maximization**: Maximize the Q function with respect to θ and call the maximizing parameter, θ_{i+1}. Maximization can be performed using any standard optimization algorithm. Then set $\theta = \theta_{i+1}$. (3). **Iterate**: Repeat steps 1 and 2 until the change in parameter vector is within a specified tolerance level.

6. EXAMPLES

In this section, two examples are presented to illustrate the algorithms developed in this article. The first example is taken from Goodwin and Agüero [2005] and the second example is a chemical reactor from Morningred et al. [1992].

Synthetic: Consider the following nonlinear process (Goodwin and Agüero [2005]) $x_{t+1} = ax_t + bu_t + w_t, y_t = c\cos(x_t) + v_t$ where $w_t \sim \mathcal{N}(0, Q)$, $v_t \sim \mathcal{N}(0, R)$, and $a = 0.9$, $b = c = 1$, $Q = R = 0.1$. In order to compare the results from the proposed algorithm with those reported in (Goodwin and Agüero [2005]), similar simulation conditions are used to the extent possible. As in (Goodwin and Agüero [2005]), the following initial parameter estimates are used $\hat{a} = \hat{b} = \hat{c} = 0.5$, $\hat{Q} = \hat{R} = 0.05$ with a white input variance of unity. In the first simulation experiment, $T = 100$ measurements are collected, and all the available data is used in the algorithm with $N = 150$ particles. The model parameters converged to a neighborhood of the true parameters in about 100 iterations. The trajectory of the parameters is shown in figure 1(a). The trajectory of the Q-function [2] is shown in figure 1. In the original form of EM algorithm, the likelihood function is expected to increase after each iteration. However, as seen in figure 1, the likelihood function is not a monotonically increasing function. This feature of the approximate EM algorithm proposed in this article is due to the fact that only an approximation of the expectation algorithm is used and not the exact expected value of the complete likelihood function. In the second experiment on this model, 10% of the available measurements are removed randomly. In other words, an observation is taken if a uniformly distributed random variable, l, in the interval $[0, 1]$ is less than 0.1. Similar experiments are conducted with 25% and 50% of the data missing. The trajectories of the log-likelihood function are shown in figure 1. The parameter values after 100 iterations from each of these experiments are shown in table 1. In all the experiments, estimated parameters settled into a neighborhood of the true parameters. As the percentage of missing data increases, the algorithm takes lot more iterations to settle close to the true parameters (see figure 1). In fact, the experiment with 50% missing data does not settle into a neighborhood of the true log-likelihood even after 100 iterations. The relatively noticeable variance in the Q-function is due to the small data length. In the next example, a much larger data set is chosen resulting in smaller variance in the Q-function.



(a) Parameter trajectories for the full data case.

(b) The trajectory of the Q-function for different experiments.

Fig. 1. Synthetic Example.

Table 1. Parameter values after 100 iterations

Parameter	% missing data			
	0%	10%	25%	50%
a	0.8985	0.8969	0.8936	0.8972
b	1.0366	1.0353	1.0511	0.9997
c	0.9712	0.9763	1.0043	0.9819
Q	0.0992	0.0968	0.1082	0.0942
R	0.0817	0.0859	0.0769	0.0772

Adiabatic CSTR: The governing equations of a popular CSTR are given below (Henson and Seborg [1997])

$$\frac{dC_A}{dt} = \frac{q}{V}(C_{Ai} - C_A) - k_0 C_A e^{-E_A/T}$$
$$\frac{dT}{dt} = \frac{q}{V}(T_i - T) - \frac{\Delta H}{\rho C_p} k_0 C_A e^{-E_A/T} - \frac{\rho_c C_{pc}}{\rho C_p V} q_c (1 - e^{-\frac{hA}{q_c \rho_c C_{pc}}})(T - T_c)$$

where C_A is the concentration of the reactant in the reactor, T is the temperature in the reactor, q is the flow rate, V is the volume of the reactor, C_{Ai} and T_i are inflow concentration and temperature, $k_0 C_A e^{-E_A/T}$ is the reaction rate, ΔH is the reaction heat, ρ and ρ_c are the densities of the reactant and the cooling fluid respectively, C_p and C_{pc} are the corresponding specific heats, h and A are the effective heat transfer coefficient and area respectively, T_c and q_c are the temperature and flow rate of the cooling fluid. The parameters and operating conditions of this CSTR are given in Henson and Seborg [1997] [3]. For simulation purposes, the above differential equations are discretized and noise is added. The CSTR is operated around a steady state corresponding to $C_A = 0.1 mol/L$ and $T = 438.54K$ with the following noise covariance matrices $Q = 2.5 \times 10^{-7}[0.1 \;\; 0; 0 \;\; 1]$ and $R = 2.5 \times 10^{-5}[0.1 \;\; 0; 0 \;\; 1]$ and discretizing sample time, $\Delta t = 0.02$ is chosen. In order to reduce the number of parameters, the state and measurement covariance matrices are parametrized as follows: $Q = q^2 \times 10^{-5}[0.1 \;\; 0; 0 \;\; 1]$ and $R = r^2 \times 10^{-3}[0.1 \;\; 0; 0 \;\; 1]$. The initial guess for the parameter vector is $\theta_1 = 6 \times 10^{10}$, $\theta_2 = 14.4 \times 10^{12}$, $\theta_3 = 6 \times 10^2$, $q = \sqrt{0.05}$, and $r = \sqrt{0.05}$. This example posed a couple of unforeseen challenges. It is found that 'large' levels of noise in the state equation lead to an unstable system. On the other hand, it is well-known that small noise levels result in an EM algorithm that is extremely slow at converging (Petersen et al. [2005]). In fact during simulations, the Q-function barely changes even after 20 iterations. Therefore, in order to speed up the EM algorithm, an overrelaxed EM algorithm (Salakhutdinov et al. [2003]) is implemented. The idea behind overrelaxed algorithm is to hasten the movement of parameter vector in the direction in which it

[2] The variable plotted is proportional to the average log-likelihood function

[3] not reproduced in this paper to save space

Table 2. Parameter values after 300 iterations

Parameter	% missing data			
	0%	10%	25%	50%
$\theta_1 \times 10^{-10}$	7.2001	7.1996	7.1999	7.2019
$\theta_2 \times 10^{-12}$	14.4031	14.4065	14.4067	14.4069
$\theta_3 \times 10^{-2}$	8.6	45.25	13.56	8.92
q	0.1629	0.1657	0.1624	0.1795
r	0.1601	0.1587	0.1585	0.1594

is moving by taking steps larger than those suggested by the EM algorithm. If these large steps result in a decrease in the value of Q-function, then the corresponding value of θ is thrown away and the basic version of EM algorithm is started again from the previous value of parameter vector.

A plot showing the Q-function as a function of the iterations is shown in figure 2. As in the previous example, as the percentage of missing data increases, the algorithm gets slower. The parameter values after 300 iterations are shown in table 2. All the parameters, except θ_3, converge to a neighborhood of the true parameter values. It is found that the sensitivity of the estimated log-likelihood function to changes in θ_3 is smaller than its variance. This is supported by the fact that the theoretical average Q-function is 24.2631, while the Q-function in the simulations converges to a neighborhood of this true value (about 24.1 from figure 2) even though θ_3 is not in the neighborhood of its true value.

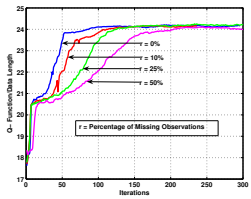


Fig. 2. The trajectory of the Q-function for different experiments.

7. CONCLUSIONS

An identification algorithm based on Expectation Maximization is developed for nonlinear state-space models to handle missing observations. The expectation step in the EM algorithm is performed by using particle approximations of state filter, smoother, and a joint density function between the state and observations. The convergence of EM algorithm depends on the percentage of missing observations. The higher the missing observations, the slower the EM algorithm. The proposed algorithm is computationally intensive, however, this problem is mitigated to an extent by using fast computational algorithms for evaluation of sum of exponential functions.

REFERENCES

C. Andrieu, A. Doucet, S.S. Singh, and V.B. Tadic. Particle methods for change detection, system identification, and control. *Proceedings of the IEEE*, 92:423–438, 2004.

A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. B*, 39:1–38, 1977.

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

S. Gibson, A. Wills, and B. Ninness. Maximum-likelihood parameter estimation of bilinear systems. *IEEE Transactions on Information Theory*, 50(10):1581–1596, 2005.

G.C. Goodwin and J.C. Agüero. Approximate EM algorithms for paramter and state estimation in nonlinear stochastic models. *Proceedings of IEEE Conference on Decision and Control*, pages 368–373, 2005.

R. D. Gudi, S. L. Shah, and M. R. Gray. Adaptive multirate state and parameter estimation strategies with application to a bioreactor. *AIChE Journal*, 41(11): 2451–2464, 1995.

M.A. Henson and D.E. Seborg. *Nonlinear Process Control*. Prentice Hall, New Jersy, USA, 1997.

A. J. Isaksson. Analysis of identified 2-D noncausal models. *IEEE Transactions on Information Theory*, 39: 525–534, 1993.

M. Klaas, M. Briers, D. Nando, and A. Doucet. Fast particle smoothing: If I had a million particles. *International Conference on Machine Learning*, 2006.

D. Li, S.L. Shah, T. Chen, and K. Qi. Application of dual-rate modeling to CCR octane quality inferential control. *IEEE Transactions on Control Systems Technology*, 11 (1):43–51, 2003.

J.D. Morningred, B. E. Paden, D. E. Seborg, and D. A. Mellichamp. An adaptive nonlinear predictive controller. *Chemical Engineering Science*, 47:755–762, 1992.

K.B. Petersen, O. Winther, and K.L. Hansen. On the slow convergence of em and vbem in low-noise linear models. *Neural Computation*, 17(9):1921–1926, 2005.

G. Poyiadjis, A. Doucet, and S.S. Singh. Maximum likelihood paramter estimation in general state-space models using particle methods. *Joint Statistical Meetings*, 2005.

H. Raghavan, A. K. Tangirala, R. B. Gopaluni, and S. L. Shah. Identification of chemical processes with irregular output sampling. *Control Engineering Practice*, 14(5): 467–480, 2006.

S. Roweis and Ghahramani. *Kalman Filtering and Neural Networks*, chapter Learning Nonlinear Dynamical Systems using the EM algorithm. John Wiley, 2001.

R. Salakhutdinov, S.T. Roweis, and Z. Ghahramani. Adaptive overrelaxed bound optimization methods. In *Proceedings of International Conference on Machine Learning*, pages 664–671, 2003.

T.B. Schön, A. Wills, and B. Ninness. Maximum likelihood nonlinear system estimation. In *Proceedings of IFAC Symposium on System Identification*, pages 1003–1008, 2006.

R.H. Shumway and D.S. Stoffer. *Time Series Analysis and Its Applications*. Springer, 2000.

S. Tatiraju, M. Soroush, and R. Mutharasan. Multirate nonlinear state and parameter estimation in a bioreactor. *Biotechnology and Bioengineering*, 63(1):22–32, 1999.

L. Wang and P. Gawthrop. On the estimation of continuous transfer functions. *International Journal of Control*, 74(9):889–904, 2001.

J.I. Yuz and G.C. Goodwin. On sampled-data models for nonlinear systems. *IEEE Transactions on Automatic Control*, 50(10):1477– 1489, October 2005.