Particle Filters for Recursive Model Selection in Linear and Nonlinear System Identification

V. Kadirkamanathan†, M. H. Jaward†, S. G. Fabri and M. Kadirkamanathan‡*

† Department of Automatic Control & Systems Engineering, University of Sheffield, UK

* Department of Power & Control Engineering, University of Malta, Malta

‡ Dragon Systems UK Ltd., Cheltenham, UK

Abstract

Recursive model selection can be addressed within the Bayesian framework, the multiple model algorithm being one such approach for linear Gaussian systems. The recent advances in nonlinear non-Gaussian estimation with the sequential Monte Carlo algorithms such as the particle filter allow the application of Bayesian inference to the development of recursive model selection algorithms for general nonlinear non-Gaussian systems. Such an algorithm is developed in this paper and applied to a linear auto-regressive (AR) and nonlinear auto-regressive (NAR) systems.

1 Introduction

Identification of unknown systems can be approached from a view point to replicate the input–output behaviour of the unknown system or alternatively, to determine the model structure of the system as well. The latter is often important for analysis and model-based control design.

The system identification problem can be posed as follows: Given the set of K input – output noisy observations $Z_K = \{z_k = (u_k, y_k) | k = 1, \cdots, K\}$ where u_k are the input and y_k are the output observations respectively, determine the underlying model $\mathcal{H}$ generating these observations, *ie.*,

$$\mathcal{H}: \quad y_k = f_{\mathcal{H}}(Z_{k-1}; w) + \nu \tag{1}$$

where ν is the noise and $f_{\mathcal{H}}(\cdot)$ is a function mapped by the model class $\mathcal{H}$ and for a given set of model parameters w, by the particular model within this class. For example, a linear second order autoregressive model will have, $f_{\mathcal{H}} = w_1 y_{k-1} + w_2 y_{k-2}$. Thus the identification problem consists of two sub-problems, namely, *model selection* that requires the best model class $\mathcal{H}$ be chosen and *parameter estimation* that requires the optimal parameters w for this model class be determined.

In the absence of any prior knowledge of the system to be identified, the candidate model class is infinite. For a practical identification procedure, *a priori* assumptions are therefore made. These typically are of the form that the unknown system is linear and/or that the system dynamics is limited to some maximum order. Once the search space for the model class is reduced to a tractable size, optimal parameter set for each model class can be determined and model comparison made. While selection criteria such as Akaike Information Criteria (AIC), Minimum Description Length (MDL) and Bayesian Information Criteria (BIC) have been developed, they are all based on the entire set of data and are therefore applicable to batch identification.

For recursive system identification, both, recursive parameter estimation and recursive model selection must be employed. Recursive parameter estimation is a well researched subject [10] and thus a significant number of algorithms are available. Except in a few special cases such as linear model with Gaussian noise, the interest in these algorithms are in obtaining a value for the parameters or if the parameters are viewed as random variables, then the expected values of the parameters. The random variable view of the unknown model parameters allows the state-space estimation algorithms to be applied to parameter estimation.

State estimation or filtering also has a long history of developments. It is well known that the optimal filter for linear model with Gaussian noise is the *Kalman filter* [1]. State estimation for nonlinear systems with non-Gaussian noise is a difficult problem and in general, the optimal solution cannot be expressed in closed-form. Sub-optimal solutions use some form of approximation such as model linearisation in the extended Kalman filter (EKF) and the probability distribution approximation in the Gaussian sum filter, depending on whether the distributions is unimodal or multimodal [5]. More recently, Monte Carlo sampling from the distribution is utilised in the development of the particle filter [6]. A particular advantage with the sample based approximation is its versatility and generality in applying it to the nonlinear non-Gaussian case [9].

Recursive model selection has been addressed in the target tracking problem domain [8], but only to the extent

of system state estimation by data association. There is very little work done in recursive model selection in the context of system identification. This is largely due to the two sub-problems of parameter estimation and model selection being treated separately.

Here, we adopt a state-space approach to the problem of recursive system identification and apply Bayesian inference. This unifies the two sub-problems and allows recursive model selection to be addressed within a single framework. The solution advocated in this paper is the one based on the sequential Monte Carlo algorithms developed for general filtering problems. The particle filtering approach has been used for model selection [2], but only within a block estimation scheme.

2 Recursive Parameter Estimation

The state-space formalism in state estimation or filtering is well suited to Bayesian inference. The optimal filter (Kalman filter) for the linear state-space model with Gaussian noise can be derived using Bayesian inference. The optimal filter (in the Bayesian sense) for the non-linear non-Gaussian case can be similarly specified, but of course, it can only be approximated in practice. The random variable view of states in state estimation can be invoked for the parameter estimation case as well and a state-space dynamical and observation equations specified as follows:

$$\begin{aligned} w_k &= w_{k-1} + \eta_k \\ y_k &= f(w_k) + \nu_k \end{aligned} \tag{2}$$

where η_k is a zero mean disturbance term introduced in the state equation which describes a *random walk* process in which the expected value of w_k is the same as the expected value of w_{k-1} (parameter remains unchanged over time). The observation equation is essentially the same as in equation (1) where the arguments Z_{k-1} has been excluded for simplicity.

The optimal recursive parameter estimate based on the state-space equations can be obtained by applying Bayesian inference separately to the state and observation equations. In the first case, this leads to the prediction stage:

$$p(w_k|Z_{k-1}) = \int p(w_k|w_{k-1})p(w_{k-1}|Z_{k-1})dw_{k-1} \tag{3}$$

where the probability distribution $p(w_k|w_{k-1})$ is based on the state equation and depends on the noise statistics of η_k. If η_k is zero mean Gaussian with covariance Q_k, *ie.*, $\eta_k \sim \mathbf{N}(0, Q_k)$ and $p(w_{k-1}|Z_{k-1}) \sim \mathbf{N}(\hat{w}_{k-1}, P_{k-1})$ is Gaussian, then the probability distribution for the prediction is also Gaussian $p(w_k|Z_{k-1}) \sim \mathbf{N}(\hat{w}_{k|k-1}, P_{k|k-1})$ with

$$\hat{w}_{k|k-1} = \hat{w}_{k-1} \quad P_{k|k-1} = P_{k-1} + Q_k \tag{4}$$

Bayesian inference on the observation equation leads to the correction stage:

$$p(w_k|Z_k) = \frac{p(z_k|w_k)p(w_k|Z_{k-1})}{p(z_k|Z_{k-1})} \tag{5}$$

where $p(z_k|w_k)$ is based on the observation equation and depends on the noise statistics of ν_k. The term $p(z_k|w_k)$ is known as the likelihood, $p(w_k|Z_{k-1})$ as the prior, $p(w_k|Z_k)$ as the posterior and $p(z_k|Z_{k-1})$ as the evidence. If $\nu_k \sim \mathbf{N}(0, R_k)$ and $p(w_k|Z_{k-1}) \sim \mathbf{N}(\hat{w}_{k|k-1}, P_{k|k-1})$ are Gaussians, then $p(w_k|Z_k) \sim \mathbf{N}(\hat{w}_k, P_k)$ is also Gaussian with

$$\begin{aligned} \hat{w}_k &= \hat{w}_{k|k-1} + g_k(y_k - f(\hat{w}_{k|k-1})) \\ g_k &= P_{k|k-1}\phi_k[R_k + \phi_k^T P_{k|k-1}\phi_k]^{-1} \\ P_k &= P_{k|k-1} - g_k\phi_k^T P_{k|k-1} \end{aligned} \tag{6}$$

as long as the function $f(w_k)$ is linear in the parameters w_k such that

$$f(w_k) = w_k^T \phi_k \tag{7}$$

where ϕ_k consists of the input, output terms from Z_{k-1}.

The equations (4) and (6) together form the complete set of Kalman filter equations for recursive parameter estimation in the linear Gaussian case. In general, the equations (3) and (5) have to be approximated to provide a recursive algorithm.

3 Recursive Model Selection

Recursive parameter estimation described in the previous section assumes that a candidate model class is either known or has been chosen. Hence, the inference equations (3) and (5) must be viewed as conditional densities given the model class. The denominator in the RHS of equation (5) is thus $p(z_k|Z_{k-1}, \mathcal{H})$ where $\mathcal{H}$ is the chosen candidate model class. As discussed previously, *a priori* assumptions have to be invoked to limit the search of a suitable $\mathcal{H}$ and once this is done, Bayesian inference for recursive model selection is straight-forward.

Let the candidate class of models be denoted $\mathcal{H}_h$ with $h \in \{1, \cdots, H\}$. Using Bayes' rule, the a posteriori model probability $P(\mathcal{H}_h|Z_k)$ is given by,

$$P(\mathcal{H}_h^k|Z_k) = \frac{p(z_k|\mathcal{H}_h^k, Z_{k-1})P(\mathcal{H}_h^k|Z_{k-1})}{p(z_k|Z_{k-1})} \tag{8}$$

where $p(z_k|\mathcal{H}_h^k, Z_{k-1})$ is the likelihood for the model, $P(\mathcal{H}_h^k|Z_{k-1})$ is the prior model probability and $P(\mathcal{H}_h^k|Z_k)$ is the posterior and the evidence being the normalising term. The notation $P(\mathcal{H}_h^k|Z_k)$ is used to indicate $P(\mathcal{H}_k = \mathcal{H}_h|Z_k)$ in the sense that at time k, it defines the probability that the unknown system model is $\mathcal{H}_h$ given the observations Z_k. The model likelihood is in fact the evidence term of the parameter estimation

stage where marginalisation over parameter has been done.

The prior for the model selection process at time k is obtained by defining a Markov chain with probabilities,

$$P(\mathcal{H}_h^k|\mathcal{H}_j^{k-1}) = \pi_{hj} \tag{9}$$

which is assumed known. It follows that,

$$P(\mathcal{H}_h^k|Z_{k-1}) = \sum_{j=1}^{H} \pi_{hj} P(\mathcal{H}_j^{k-1}|Z_{k-1}) \tag{10}$$

Such a transition process is defined in a similar spirit to the random walk model in parameter estimation. While the expected value of the parameter is known to be a constant, the random walk model allows the exploration of the appropriate estimate. Here, even though there is a single most probable candidate model, the transition process allows other model classes to be explored. Such a description in fact is a special case of the *hidden Markov model* (HMM) where at any time instant k, any one of the candidate model class may have generated the observation. HMMs are a nonstationary generalisation of the case considered here.

This natural hierarchy in model estimation and model selection is what lead to the use of parallel Kalman filters within the multiple model framework [8], which relies on the classical assumptions of linearity and Gaussianity. In the nonstationary case where the candidate model class may be subject to change, the optimal estimate requires an exponentially increasing (with k) number of models to be considered. Sub-optimal schemes have thus been developed in which only a fixed number of models are considered [8]. The multiple model framework has been applied to state estimation [8] and system identification [13] [11] amongst others.

4 Particle Filters

The use of Monte Carlo methods in state estimation was first proposed in [7]. In their work, the methods were used only to obtain estimates of the mean and covariance of the state estimate. With increasing computational power, Monte Carlo methods have become popular again. The concept of *particle filters*, essentially a Monte Carlo method, approximates the complete state posterior probability density by a pool of particles or samples [6]. Since then, a number of on-line or sequential Monte Carlo algorithms have been developed, a review of these are in [3], [4], [5]. The particle filters are essentially an SIR (sampling importance resampling) method, whereas an equally popular method is the MCMC (Markov chain Monte Carlo) method. In this paper, we follow the particle filtering approach.

Samples or particles are drawn from a continuous probability distribution and hence,

$$p(w_k|Z_k) \approx \frac{1}{N} \sum_{n=1}^{N} \delta(w_k - w_k^{(n)}) \tag{11}$$

where $w_k^{(n)}$, for $n = 1, \ldots, N$ are the N samples and $\delta(\cdot)$ is the dirac-delta function. This representation can now be used in solving the basic estimation equations (3) and (5). If N' samples are also drawn from the noise distribution,

$$p(w_k|w_{k-1}) = \frac{1}{N'} \sum_{n'=1}^{N'} \delta\left(w_k - w_{k-1} + \eta_{k-1}^{(n')}\right) \tag{12}$$

and given that the posterior for w_{k-1} is similarly represented in terms of N particles, the prediction stage becomes,

$$p(w_k|Z_{k-1}) = \frac{1}{N''} \sum_{n''=1}^{N''} \delta\left(w_k - w_k^{(n'')}\right) \tag{13}$$

where

$$w_k^{(n'')} = w_{k-1}^{(n)} + \eta_{k-1}^{(n')} \qquad \text{for each } n, n' \tag{14}$$

are the (NN') samples propagated from the prediction stage. This step as derived above results in an increase in the number of particles which may be used for increased accuracy [12]. We follow the standard particle filter approach for the sample propagation step, where,

$$w_k^{(n)} = w_{k-1}^{(n)} + \eta_{k-1}^{(n)} \qquad \text{for} \quad n = 1, \ldots, N \tag{15}$$

so that for each $w_{k-1}^{(n)}$ particle only one $\eta_{k-1}^{(n)}$ sample is generated.

The correction stage yields,

$$p(w_k|Z_k) = \sum_{n=1}^{N} q_k^{(n)} \delta(w_k - w_k^{(n)}) \tag{16}$$

where

$$q_k^{(n)} = \frac{\tilde{q}_k^{(n)}}{\sum_{n=1}^{N} \tilde{q}_k^{(n)}} \qquad \text{and} \qquad \tilde{q}_k^{(n)} = p(z_k|w_k^{(n)}) \tag{17}$$

are the normalised and un-normalised weights or probability mass associated with the particles or samples. This results in the posterior being represented in terms of weighted particles. The expected value of the parameter based on these samples can be computed by,

$$\hat{w}_k = \sum_{n=1}^{N} q_k^{(n)} w_k^{(n)} \tag{18}$$

A resampling from this posterior distribution is carried out to obtain equally weighted samples. Resampling is

achieved by generating a random number τ_n from the uniform distribution over $[0,1]$ for each $n = 1, \ldots, N$ and choosing the sample $w_k^{*(\nu)}$ such that

$$\sum_{i=0}^{\nu-1} q_i < \tau_n \leq \sum_{i=0}^{\nu} q_i \tag{19}$$

with $q_0 = 0$. The prediction and correction stages are repeated for each observation to obtain the sample based recursive posterior parameter distribution. The expected parameter value following the resampling is,

$$\hat{w}_k = \frac{1}{N} \sum_{n=1}^{N} w_k^{*(n)} \tag{20}$$

5 Model Selection Algorithm

The contribution in this paper is the development of a model selection scheme with particle filters as opposed to the use of MCMC methods [2]. The idea is to represent the joint distribution $p(\mathcal{H}_k, w_k)$ through samples or particles. One method of achieving this is to create a pool of samples with each candidate model containing different number of state particles. Note that since the candidate models have different orders, the dimensions of these state particles will also differ. The model selection scheme can also be used to test different hypothesis yielding state particles with the same dimension, such as in the application to fault detection.

Let N_h denote the total number of particles associated with the candidate model $\mathcal{H}_h$. At any given time k, the pool consists of the particles,

$$\left\{ \{(h, w_k^{(n_h)})\}_{n_h=1,.,N_h} \right\}_{h=1,.,H} \tag{21}$$

where $h \in 1, \cdots, H$ is the label associated with the candidate model and $w_k^{(n_h)} \sim p(w_k|\mathcal{H}_h, Z_k)$ is the associated state vector drawn from the posterior probability conditioned on the candidate model. The total number of particles in the pool is given by,

$$N = \sum_{h=1}^{H} N_h \tag{22}$$

The particle filter based model selection algorithm also essentially has two stages – prediction and correction. The prediction stage is based on using (10) to compute the prior probabilities

$$P(\mathcal{H}_h^k|Z_{k-1}) = \sum_{j=1}^{H} \pi_{hj} \frac{N_h^{k-1}}{N} \tag{23}$$

where N_h^{k-1} is the total number of samples from the posterior distribution at time $(k-1)$. If the pool at this predictive stage should reflect the joint distribution of the model order and states, a resampling step needs to be introduced. If the total number of particles chosen is very high, then this will result in a pool consisting of numbers of particles associated with each model order in proportion to the posterior model probability. A fast implementation scheme can be derived by first determining the numbers of particles expected in the pool, given by,

$$N_h^k = N \cdot P(\mathcal{H}_h^k|Z_k) \tag{24}$$

and then to carry out the resampling process for each candidate model. Effectively, this step implies using N_h^{k-1} particles with associated weights to resample N_h^k particles with equal weights. The resampling is carried out for each combination of $\mathcal{H}_{k-1}$ and $\mathcal{H}_k$ where the number of particles are

$$N_{hj} = \pi_{hj} N_h^{k-1} \tag{25}$$

The reduction or increase in the dimensionality of the particles are achieved simply by zeroing the higher order values or adding random values to the higher order terms. The particles now have equal weights $1/N$.

An alternative is to generate additional particles such that there are NH particles in total at this stage, as is done for the hidden Markov models (HMMs) [5]. Since by equation (10) each candidate model $\mathcal{H}_h$ undergoes a transition, N_h copies of $w_{k-1}^{(n_h)}$ are made to each of $h = 1, \cdots, H$ at the prediction stage at time k. However, this will create an unnecessary number of particles whose weights are likely to be low and thus will lead to unwarranted computational complexity.

The resampled particles are passed on to the particle filter parameter estimation algorithm involving the same prediction and correction step as described in the previous section. At the posterior parameter estimation stage, there are now N_h^k particles associated with each candidate model $\mathcal{H}_h$ with weights $q_k^{(n_h)}$.

Model posterior probabilities are thus given by,

$$P(\mathcal{H}_h^k|Z_k) = \sum_{n_h=1}^{N_h^k} q_k^{(n_h)} \tag{26}$$

which can be the basis for carrying out model comparison. The resampling process carried out at the end of the posterior parameter estimation of section 4 is not carried out at the same stage in the model selection algorithm so that in a single iteration there is only one resampling procedure. This is to avoid increase in variance in the sampling [5].

The algorithm can be summarised as follows:

- **Initialise**: All particles are generated from single model (of lowest order) with equal weights.

- **Iteration** for each time:
  - **Model prediction**: Determine the number of particles (25) for each transition and resample. Maintain associated particle weight $q_k^{(n)} = q_{k-1}^{(n)}$.
  - **Parameter prediction**: Propagate particles based on (15).
  - **Parameter correction**: Determine particle weights using (17).
  - **Model correction**: Determine model posterior probabilities with (26).

6 Example – Linear Non-Gaussian case

The particle filter based recursive model selection algorithm is applied to the problem of autoregressive (AR) model order selection problem. The simulated process is of third order, given by,

$$y_k = 0.5y_{k-1} - 0.95y_{k-2} + 0.5y_{k-3} + \nu_k \qquad (27)$$

where the driving disturbance is zero uniform uncorrelated noise in the interval $[-0.1, 0.1]$. With this non-Gaussian driving noise, the classical methods simply form an approximate scheme for model selection. The simulation parameter $K = 1000$ and the random walk disturbance variance is 0.002. The particle filter algorithm design parameters are, $N = 2000$, $H = 8$. The transition probability matrix is given by, $\pi_{h,h} = 0.95$ and $\pi_{h,h-1} = \pi_{h,h+1} = 0.025$ and others zero.

The results show, as expected, that the model selection algorithm chose model order 3 which has the highest probability in Figure 1. Model orders 2 and 4, being nearest to order 3, were the other two model classes with significant probabilities. Figure 2 shows the parameter values of the most probable model (order 3) converging over time. After the initial transition period, it is clear that the parameters have converged to the true values of $0.5, -0.95, 0.5$.

7 Example – Nonlinear Gaussian case

In the second example, a nonlinear Volterra system was considered, where

$$y_k = 0.1 - 0.5y_{k-1} - 0.9y_{k-2} - 0.1y_{k-1}^2 - 0.75y_{k-2}^2 + \nu_k \qquad (28)$$

This is a second order Volterra system with input order also equal to 2. Note that the second order Volterra system refers to the highest order of the polynomial in the expansion for $f(\cdot)$ being two whereas the input order refers to the presence of terms y_{k-1}, y_{k-2} in the functional. Also, the parameter associated with the term $y_{k-1}y_{k-2}$ in this model is zero.

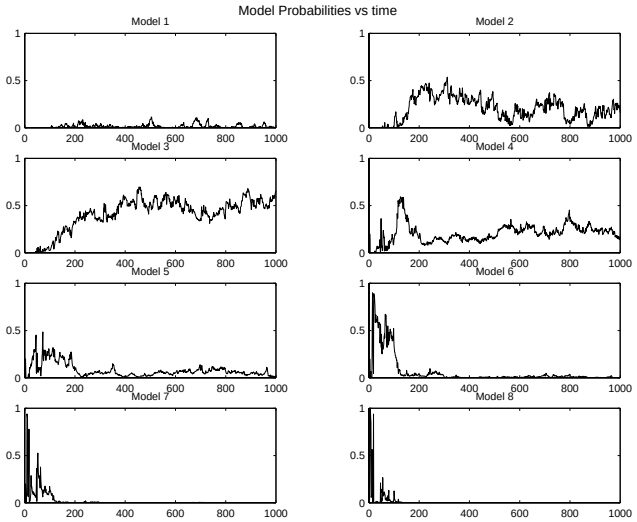

Figure 1: Model probabilities over time with orders [1–8] going from left to right and then top to bottom

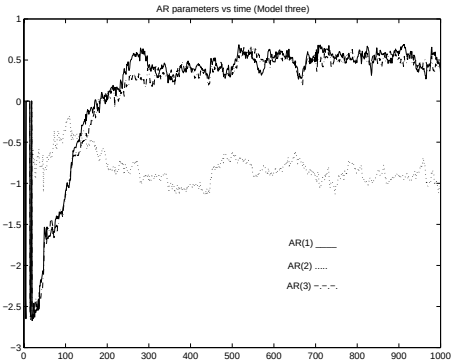

Figure 2: Model parameters variation over time for model with order 3

The model selection algorithm with the following parameters, $N = 2000$, $K = 250$, random walk being zero mean Gaussian with variance 0.0005 and the noise being Gaussian with variance of 0.1, was used. The task was to find the input order of the system which affects the accuracy of identification more strongly. The classes of models considered were second order Volterra series with input orders ranging from $0 - 4$, the model with input order τ being,

$$y_k = w_0 + \sum_{i=1}^{\tau} w_i y_{k-i} + \sum_{i=1}^{\tau}\sum_{j=1}^{i} w_{ij} y_{k-i}y_{k-j} + \nu_k \quad (29)$$

The model probability results obtained are shown in Figure 3. Again, as expected, the particle filter based model selection scheme estimated the most probable model, amongst those chosen, to be of input order 2 (model 3). Given that a random walk model was used for the parameter evolution, the models with the incorrect order have the ability to vary the parameters over time to account for the modelling errors. This is seen in Figure 3 where the probabilities for model orders 1 (model 2) and 3 (model 4) are not insignificant.

However, the inability of model with order 0 (model 1) to fit the data and the over-parametrization of model with order 4 (model 5) have resulted in their probabilities being near zero.

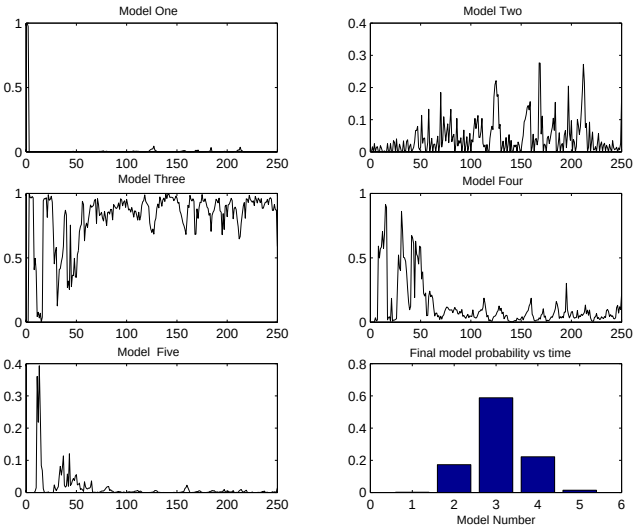

Figure 3: Model (orders 0,1,2,3,4) probabilities with time and model class probabilities at end time

Figure 4 shows the parameter estimates of the most probable model, *ie.*, model with order 2 (model 3) varying over time. The figure shows that the estimates are gradually converging to their true values, the variations being due to the level of noise in the data.

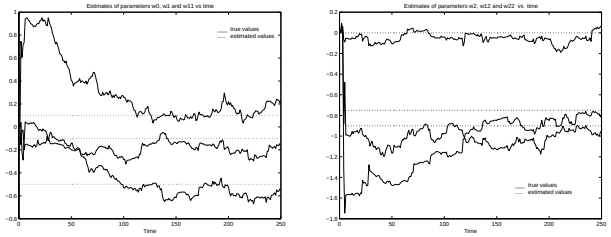

Figure 4: Parameter estimates (–) and their true values (- -) for model 3, over time.

8 Conclusions

A particle filter based recursive model selection algorithm for linear and nonlinear system identification has been introduced. The idea has been to create a pool of particles that reflect the posterior model probabilities as well as the model conditioned posterior parameter probability densities. A transition process is defined which allows jumps between candidate models of different orders with appropriate probabilities, thus allowing a wider search over the candidate models. Simulation results on a linear and nonlinear identification problems are used to demonstrate the operation and performance of the algorithm.

The two examples of linear and nonlinear identification considered here are both linear in the model parameters, albeit with non-Gaussian noise. The particle filter based algorithm is applicable to more general and nonlinear system and with non-Gaussian noise and thus the model selection algorithm can readily be extended to these cases. Work is currently on-going to extend the scheme to nonlinearly parametrized systems and apply these ideas to real signals.

References

[1] B. D. O. Anderson and J. B. Moore. *Optimal Filtering*. Prentice-hall, Englewood Cliffs, NJ, 1979.

[2] A. Doucet, C. Andrieu, W. J. Fitzgerald and J. M. Perez. Bayesian Computational Approaches to Model Selection. In *Nonlinear and Nonstationary Signal Processing* W.J. Fitzgerald, R.L. Smith, A.T. Walden and P. C. Young (eds.) , Cambridge: Newton Institute Series, Cambridge University Press, 2000.

[3] N. Bergman. Recursive Bayesian estimation: Navigation and tracking applications. *PhD Thesis*, Linkoping University, 1999.

[4] A. Doucet, S. Godsill and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, **10**:197-208, 2000.

[5] P. Fearnhead. Sequential Monte Carlo methods in filter theory. *PhD Thesis*, Oxford University, 1998.

[6] N. J.Gordon, D. J. Salmond and A. F. M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings-F*, **140**(2): 107-113, 1993.

[7] J. E. Handschin and D. Q. Mayne. Monte Carlo simulation techniques to estimation the conditional expectation in multi-stage non-linear filtering. estimation. *Int. J. Control*, **9**(5): 547–559, 1969.

[8] X. R. Li and Y. Bar Shalom. *Estimation and tracking principles, techniques and software*. Artech, Boston, 1993.

[9] J. S. Liu and R. Chen. Sequential Monte Carlo methods for dynamic systems. *JASA*, **93**(443): 1032-1044, 1998.

[10] L. J. Ljung and T. Soderstrom. *Theory and practice of recursive identification*. MIT Press, Cambridge, MA, 1983.

[11] V. Kadirkamanathan. Recursive nonlinear identification using multiple model algorithm. *Proc. IEEE Workshop on Neural Nets for Sig. Proc.*, **V**: 171-180, 1995.

[12] G. Kitagawa. A self-organising state-space model. *JASA*, **93**(443): 1203-1215, 1998.

[13] M. Millnert. Identification of ARX models with Markovian parameters. *Int. J. Control*, **45**(6): 2045-2058, 1987.