

NONLINEAR STATE ESTIMATION BY EVOLUTION STRATEGIES BASED PARTICLE FILTERS

Katsuji Uosaki ^{*,1} Toshiharu Hatanaka ^{*}

** Department of Information and Physical Sciences
Graduate School of Information Science and Technology
Osaka University
Suita, Osaka 565-0871, Japan
Phone: +81-6-6879-7833 Fax: +81-6-6879-7836
E-mail: uosaki@ist.osaka-u.ac.jp*

Abstract: There has been significant recent interest of particle filters for nonlinear state estimation. Particle filters evaluate a posterior probability distribution of the state variable based on observations in Monte Carlo simulation using so-called importance sampling. However, degeneracy phenomena in the importance weights deteriorate the filter performance. By recognizing the similarities and the differences of the processes between the particle filters and Evolution Strategies, a new filter, Evolution Strategies Based Particle Filter, is proposed to circumvent this difficulty and to improve the performance. The applicability of the proposed idea is illustrated by numerical studies.
Copyright@2005IFAC

Keywords: Nonlinear filters, Monte Carlo method, Particle filter, Bayesian approach, Evolution strategies, Extended Kalman filters.

1. INTRODUCTION

State estimation of dynamic systems using a sequence of their noisy observations is ubiquitous in control system science. This problem can be solved by a Bayesian approach, that is, inference on the unknown state can be performed according to the posterior probability distribution (pdf), which is obtained by combining a prior pdf for the unknown state with a likelihood function relating them to the observations. When observations come sequentially in time, recursive state estimation, which evaluates the evolving posterior pdf recursively in time, is often interested. However, the posterior pdf only admits an analytical expression for very restricted cases, including linear Gaussian state

space models where well-known Kalman filter (Anderson and Moore, 1979; Sorenson, 1985) can be applied. In many realistic problems, state space models include nonlinear and non-Gaussian elements that preclude a closed form of expression for the optimal state estimate and that many approximations have been proposed such as the extended Kalman filter (EKF) and Gaussian sum filter (Jazwinski, 1970; Goodwin and Agüero, 2002). By the recent progress of computing ability, "particle filtering," a simulation-based method for Bayesian sequential analysis attracts much attentions. In this approach, the integral in Bayes' rule is approximated by a weighted sum based on the discrete grid sequentially chosen by the importance sampling and the estimates are obtained based on corresponding importance weights (Doucet, 1998; Arulampalam, *et al.*, 2002). A common problem in the particle filter is the degeneracy phenomenon, where almost all im-

¹ This work is partially supported by the Grant-in-Aid for Scientific Research from the Japan Society of Promotion of Sciences (C)(2)14550447.

portance weights tend to zero after some iteration. It implies a large computational effort is devoted to updating the particles with negligible weights. Some modifications such as resampling particle filter have been proposed to resolve this difficulty. We propose here a novel particle filter, Evolution Strategies Based Particle Filter, by recognizing the similarities and differences of the operations in particle filters and Evolution Strategies (Schwefel, 1995), one of the evolutionary computation approaches. Numerical simulation studies have been conducted to exemplify the applicability of this approach to nonlinear filtering.

2. PARTICLE FILTER

Consider the following nonlinear state space model.

$$x_{k+1} = f(x_k, v_k) \quad (1)$$

$$y_k = g(x_k, w_k) \quad (2)$$

where x_k and y_k are the state variable and observation, respectively, f and g are known possibly nonlinear functions, v_k and w_k are independently identically distributed (i.i.d.) system noise and observation noise sequences, respectively. We assume v_k and w_k are mutually independent. Problem to be considered here is to find the best estimate of the state variable x_k in some sense based on the all available data of observations $y_{1:k} = \{y_1, y_2, \dots, y_k\}$. We can solve the problem by calculating the posterior pdf of the state variable x_k of time instant k based on all the available data of observation sequence $y_{1:k}$.

The posterior pdf $p(x_k|y_{1:k})$ of x_k based on the observation sequence $y_{1:k}$ satisfies the following recursion:

$$p(x_k|y_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|y_{1:k-1})dx_{k-1} \quad (3)$$

(Chapman-Kolmogorov equation)

$$p(x_k|y_{1:k}) = \frac{p(y_k|x_k)p(x_k|y_{1:k-1})}{p(y_k|y_{1:k-1})} \quad (4)$$

(Bayes' rule)

with a prior pdf $p(x_0|y_0) \equiv p(x_0)$ of the initial state variable x_0 . Here normalizing constant

$$p(y_k|y_{1:k-1}) = \int p(y_k|x_k)p(x_k|y_{1:k-1})dx_k$$

depends on the likelihood $p(y_k|x_k)$, which is determined by the observation equation (2).

Since a closed form solution is not admitted except in very restrictive cases such as linear Gaussian state space models, where the well-known Kalman filter (Anderson and Moore, 1979; Sorenson, 1985) can be applied, some approximations should be introduced. The most popular approximation approach is the extended Kalman filter (EKF) (Jazwinski, 1970; Goodwin and Agüero, 2002):

$$\begin{aligned} \hat{x}_{k|k-1} &= f(\hat{x}_{k-1|k-1}) \\ P_{k|k-1} &= \tilde{A}_k P_{k-1|k-1} \tilde{A}_k^T + Q \\ \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k (y_k - g(\hat{x}_{k|k-1})) \\ P_{k|k} &= (I - K_k \tilde{C}_k) P_{k|k-1} \\ K_k &= P_{k|k-1} \tilde{C}_k^T (\tilde{C}_k P_{k|k-1} \tilde{C}_k^T + R)^{-1} \\ \tilde{A}_k &= \left. \frac{df(x)}{dx} \right|_{x=\hat{x}_{k-1|k-1}} \\ \tilde{C}_k &= \left. \frac{dg(x)}{dx} \right|_{x=\hat{x}_{k|k-1}} \end{aligned} \quad (5)$$

This is applicable to nonlinear models with additive Gaussian noise and uses a linearization technique based on a first order Taylor expansions of the nonlinear system and observation equations about the current estimate. However, it approximates the posterior pdf to be Gaussian. If the true density is non-Gaussian, then a Gaussian can never describe it well. In such cases, approximate grid-based filters and particle filters will yield an improvement in performance. They approximate the true posterior pdf with the following weighted empirical distribution of a set of $n \gg 1$ samples $\{x_k^{(i)}, (i = 1, \dots, n)\}$ called as particles or discrete grids with associated importance weights $\{w_k^{(i)}, (i = 1, \dots, n)\}$, $w_k^{(i)} > 0$, $\sum_{i=1}^n w_k^{(i)} = 1$,

$$p(x_k|y_{1:k}) \approx \sum_{i=1}^n w_k^{(i)} \delta(x_k - x_k^{(i)}) \quad (6)$$

where $\delta(\cdot)$ is Dirac's delta function ($\delta(x) = 1$ for $x = 0$ and $\delta(x) = 0$ otherwise).

Here, the particles are generated and associated weights are chosen using the principle of "importance sampling" (Doucet *et al.*, 2001):

Suppose $p(x) \propto \pi(x)$ is a pdf from which it is difficult to draw samples, but for which $\pi(x)$ can be evaluated (and so $p(x)$). Let $x^{(i)}$ ($i = 1, \dots, n$) be samples that are easily generated from a pdf $q(x)$, called an importance density. Then a weighted approximation to the density $p(x)$ is given by

$$p(x) \approx \sum_{i=1}^n w^{(i)} \delta(x - x^{(i)}) \quad (7)$$

with the normalized weight of the i -th particle

$$w^{(i)} \propto \frac{\pi(x^{(i)})}{q(x^{(i)})} \quad (8)$$

So, if the samples $x_k^{(i)}$ in (6) were drawn from an importance density $q(x_k^{(i)}|y_{1:k})$, then the associated normalized weights are defined as by (8) to be

$$w_k^{(i)} \propto \frac{p(x_k^{(i)}|y_{1:k})}{q(x_k^{(i)}|y_{1:k})} \quad (9)$$

If the importance density $q(x_k|y_{1:k-1})$ is chosen to factorize such that

$$q(x_k|y_{1:k}) = q(x_k|x_{k-1}, y_{1:k})q(x_{k-1}|y_{1:k-1}). \quad (10)$$

Then we can obtain samples $x_k^{(i)}$ by augmenting each of the existing samples $x_{k-1}^{(i)}$ sampled from the im-

portance density $q(x_{k-1}|y_{1:k-1})$ with the new state sampled from $q(x_k|x_{k-1}, y_{1:k})$.

Noting that

$$\begin{aligned} p(x_k|y_{1:k}) &= \frac{p(y_k|x_k, y_{1:k-1})p(x_k|y_{1:k-1})}{p(y_k|y_{1:k-1})} \\ &= \frac{p(y_k|x_k, y_{1:k-1})p(x_k|x_{k-1}, y_{1:k-1})}{p(y_k|y_{1:k-1})} \\ &\quad \times p(x_{k-1}|y_{1:k-1}) \\ &\propto p(y_k|x_k)p(x_k|x_{k-1})p(x_{k-1}|y_{1:k-1}) \end{aligned} \quad (11)$$

we have

$$\begin{aligned} w_k^{(i)} &\propto \frac{p(y_k|x_k^{(i)})p(x_k^{(i)}|x_{k-1}^{(i)})p(x_{k-1}^{(i)}|y_{1:k-1})}{q(x_k^{(i)}|x_{k-1}^{(i)}, y_{1:k})q(x_{k-1}^{(i)}|y_{1:k-1})} \\ &= w_{k-1}^{(i)} \frac{p(y_k|x_k^{(i)})p(x_k^{(i)}|x_{k-1}^{(i)})}{q(x_k^{(i)}|x_{k-1}^{(i)}, y_{1:k})}. \end{aligned} \quad (12)$$

The particle filter with these steps is called ‘‘Sequential Importance Sampling Particle Filter’’ (SIS).

It is known that the SIS filter suffers from the degeneracy phenomenon, where all but one of the normalized importance weights are very close to zero after a few iterations. By this degeneracy, a large computational effort is wasted to updating trajectories whose contribution to the final estimate is almost zero. In order to prevent this phenomenon, several modifications have been introduced. Among them, resampling process is used often. Its idea is to eliminate trajectories whose normalized importance weights are small and to concentrate upon the trajectories with larger weights. It involves generating new grid points $x_k^{*(i)}$ ($i = 1, \dots, n$) by resampling from the grid approximation (6) randomly with probability

$$\Pr(x_k^{*(i)} = x_k^{(j)}) = w_k^{(j)} \quad (13)$$

and the weights are reset to $w_k^{*(i)} = 1/n$. A suitable measure of degeneracy is the estimate of the effective sample size N_{eff} introduced in (Kong *et al.*, 1994). It is given by

$$\hat{N}_{eff} = \frac{1}{\sum_{i=1}^n (w_k^{(i)})^2} \quad (14)$$

with the associated normalized weight $w_k^{(i)}$. For a predetermined threshold $N_{thres} \in [1, n]$, if the effective sample size N_{eff} is smaller than N_{thres} , it is decided that the severe degeneracy is occurred and resampling step should be introduced. Particle filter with this resampling process is called ‘‘Sampling Importance Resampling Particle Filter’’ (SIR).

3. EVOLUTIONARY COMPUTATION AND EVOLUTION STRATEGIES BASED PARTICLE FILTER

In this section, we propose a novel particle filter called ‘‘Evolution Strategies Based Particle Filter’’ (ESP) by

recognizing the similarities in some steps of the ‘‘Sampling Importance Resampling Particle Filter’’ (SIR) and Evolution Strategies (ES).

3.1 Evolutionary Computation

Evolutionary computation approach is a computational model of natural evolutionary processes as key elements in the design and implementation of computer-based problem solving systems. A variety of evolutionary computation approaches such as ‘Evolutionary Programming’ (EP) (Fogel, Owens and Walsh, 1965), ‘Evolution Strategies’ (ES) (Schwefel, 1995), ‘Genetic Algorithm’ (GA) (Holland, 1992), and ‘Genetic Programming’ (GP) (Koza, 1992) have been proposed and studied. Extensive survey and comments are given in (Bäck and Schwefel, 1993; Bäck, 1996; Fogel, 1995). The common conceptual base is simulating the evolution of individuals (candidate solutions) via processes of selection and perturbation. These processes depend on the perceived performance (fitness) of the individuals as defined by the environments.

Evolutionary computation approach maintains a population of structures that evolve according to rules of selection and other operators, such as recombination and mutation. Each individual is evaluated, receiving a measure of its fitness in the environment. *Selection (reproduction)* focuses attention on high-fitness individuals, thus exploiting the available fitness information. *Recombination* (also refer to as *crossover*) and *mutation* perturb those individuals, providing general heuristics for exploration. Here we explain Evolution Strategies (ES) briefly. ES is developed by Rechenberg and Schwefel (Schwefel, 1995) to solve hydrodynamic problems. It is applied to continuous function optimization in real-valued n -dimensional space. Mutation is applied more often to the solution rather than crossover. The simplest method can be implemented as follows: Let $\mathbf{x}^{(k)} = (x_1^{(k)}, \dots, x_n^{(k)}) \in \mathbf{R}^n$, ($k = 1, \dots, \mu$) be each individual in the population.

3.1.1. Generation of initial population

We generate an initial population of parent vectors $\{\mathbf{x}^{(k)}, (k = 1, \dots, \mu)\}$ randomly from a feasible range in each dimension.

3.1.2. Evolution operations

(1) Crossover

This process allows for mixing of parental information while passing it to their descendants. A typical crossover rule is

$$x'_j = x_{S,j} + \chi \cdot (x_{T,j} - x_{S,j}) \quad (15)$$

where S and T denote two parent individuals selected at random from the population and $\chi \in$

$[0, 1]$ is a uniform random or deterministic variable. The index j in x'_j indicates j -th component of new individuals. This is a similar operator used in differential evolution (Storn and Price, 1995)

(2) Mutation

This process introduces innovation into the population. It is realized by following additive process,

$$\begin{aligned}\sigma'_j &= \sigma_j \exp(\tau' N(0, 1) + \tau N_j(0, 1)) \\ x''_j &= x'_j + \sigma'_j N_j(0, 1)\end{aligned}$$

Here, $N(0, 1)$ denotes a realization of normal random variable with mean and unit variance, $N_j(0, 1)$ denotes random variable sampled anew for counter j of normal random variable with mean and unit variance and σ_j denote the mean step size. The factors τ and τ' are chosen depending the population size μ (Bäck and Schwefel, 1993). In this approach, small variations are much more frequent than larger variations, expressing the state of affairs on the phenotypic level in nature.

(3) Selection

This is the completely deterministic process choosing the individuals of higher fitness out of the union of parents and offspring or offspring only to form the next generation in order to evolve towards better search region.

- $(\mu + \lambda)$ -selection
This creates λ offspring from μ parents and selected the μ best individuals out of the union of parents and offspring.
- (μ, λ) -selection
This creates λ offspring from μ parents and selected the μ best individuals out of offspring ($\lambda \geq \mu$).

3.2 Evolution Strategies Based Particle Filter

We will propose here a novel particle filter based on Evolution Strategies by recognizing the fact the importance sampling and resampling processes in SIR filter are corresponding to mutation and selection processes in ES. Resampling process in SIR filter selects offsprings with probability

$$w_k^{(i)} \propto \frac{p(y_k | x_k^{(i)}) p(x_k^{(i)} | x_{k-1}^{(i)}) p(x_{k-1}^{(i)} | y_{1:k-1})}{q(x_k^{(i)} | x_{k-1}^{(i)}, y_{1:k}) q(x_{k-1}^{(i)} | y_{1:k-1})} \quad (16)$$

and this corresponds to selection process in ES with fitness function $w_k^{(i)}$. On the other hand, the importance sampling process in SIR filter samples $x_k^{(i)}$ according to the importance density $q(x_k^{(i)} | x_{k-1}^{(i)}, y_{1:k})$, and this corresponds to mutation process in ES from the viewpoint of generating offsprings $x_k^{(i)}$ from the parents $x_{k-1}^{(i)}$ with extrapolation by $f(x_{k-1})$ and perturbation by v_k . The main difference is resampling in

SIR is carried out randomly and the weights are reset as $1/n$, while the selection in ES is deterministic and the fitness function is never reset. Hence, by replacing the resampling process in SIR by the deterministic selection process in ES, we can derive a new particle filter as follows.

Based on the particles $x_{k-1}^{(i)}$ ($i = 1, \dots, n$) sampled from the importance density $q(x_{k-1} | y_{1:k-1})$, we generate ℓ $x_k^{(i,j)}$, ($j = 1, \dots, \ell$) sampled from the importance density function $q(x_k | x_{k-1}^{(i)}, y_{1:k})$. Corresponding weights $w_k^{(i,j)}$ are evaluated by

$$\begin{aligned}w_k^{(i,j)} &= w_{k-1}^{(i)} \frac{p(y_k | x_k^{(i,j)}) p(x_{k-1}^{(i)} | x_{k-1}^{(i)})}{q(x_k^{(i,j)} | x_{k-1}^{(i)}, y_{1:k})} \\ & \quad i = 1, \dots, n, j = 1, \dots, \ell\end{aligned}$$

From the set of $n\ell$ particles and weights $\{x_k^{(i,j)}, w_k^{(i,j)}, (i = 1, \dots, n, j = 1, \dots, \ell)\}$, we choose n sets with the larger weights, and set as $x_k^{(i)}, w_k^{(i)}$ ($i = 1, \dots, n$). This process corresponds to $(n, n\ell)$ -selection in ES. Hence, we call this particle filter using $(n, n\ell)$ -selection in ES as Evolution Strategies based particle filter Comma (ESP(,)). When we add the particles $x_k^{(i,0)} = f(x_{k-1}^{(i)})$, ($i = 1, \dots, n$) in addition to $n\ell$ $x_k^{(i,j)}$, ($i = 1, \dots, n, j = 1, \dots, \ell$) sampled from the importance density function $q(x_k | x_{k-1}^{(i)}, y_{1:k})$ as above and evaluate the weights $w_k^{(i,j)}$, ($i = 1, \dots, n, j = 0, \dots, \ell$) by (17), and then choose n sets of $(x_k^{(i)}, w_k^{(i,j)})$ with larger weights from the ordered set of $n(\ell + 1)$ particles $\{x_k^{(i,j)}, w_k^{(i,j)}, (i = 1, \dots, n, j = 0, \dots, \ell)\}$, we can obtain another ESP filter. Since this ESP filter uses the selection corresponding to $(n+n\ell)$ -selection in ES, we can call this filter as Evolution Strategies based particle filter Plus (ESP(+)). The algorithms are summarized in Fig.1.

4. NUMERICAL EXAMPLE

To exemplify the applicability of the proposed ESP filters, we carried out a numerical simulation. We consider the following nonlinear state space model with known parameters.

$$\begin{aligned}x_k &= \frac{x_{k-1}}{2} + \frac{25x_{k-1}}{1 + x_{k-1}^2} + 8 \cos(1.2k) + v_k \\ &= f(x_{k-1}) + v_k\end{aligned} \quad (17)$$

$$y_k = \frac{x_k^2}{20} + w_k \quad (18)$$

and v_k and w_k are i.i.d. zero-mean normal random variates with variance 10 and 1, respectively. The normal distribution with mean $f(x_{k-1}^{(i)})$ and variance 10 is chosen as the importance density $q(x_k | x_{k-1}^{(i)}, y_{1:k})$. A sample behavior of the true state and corresponding observation processes is shown in Fig.2. Sample paths

Procedure ESP

```

For  $k = 0$ 
   $i = 1, \dots, n$ , sample  $x_0^{(i)} \sim q(x_0|y_0)$ ;
   $i = 1, \dots, n$ , evaluate the weight
   $w_0^{(i)} = p(y_0|x_0^{(i)})p(x_0^{(i)})/q(x_0^{(i)}|y_0)$ .
For  $k \geq 1$ 
   $i = 1, \dots, n$ 
  set  $x_k^{(i,0)} = f(x_{k-1}^{(i)})$ 
   $j = 1, \dots, \ell$ 
  sample  $\tilde{x}_k^{(i,j)} \sim q(x_k|x_{k-1}^{(i)}, y_{1:k})$ ;
   $i = 1, \dots, n$  and  $j = \underline{0}, 1, \dots, \ell$ ,
  evaluate the weight
   $w_k^{(i,j)} = w_{k-1}^{(i)} \frac{p(y_k|\tilde{x}_k^{(i,j)})p(\tilde{x}_k^{(i,j)}|x_{k-1}^{(i)})}{q(\tilde{x}_k^{(i)}|\tilde{x}_k^{(i)}, y_{1:k})}$ 
  Sort the set of pairs  $\{\tilde{x}_k^{(i,j)}, w_k^{(i,j)}\}$ 
   $w_k^{(i,j)}$  ( $i = 1, \dots, n, j = \underline{0}, 1, \dots, \ell$ )
  by the size of  $w_k^{(i,j)}$  in de-
  scending order.
  Take the first  $n$   $x_k^{(i)}$  from the
  ordered set  $\{\tilde{x}_k^{(i)}, \tilde{w}_k^{(i)}\}$ .
   $i = 1, \dots, n$ , normalize the
  weight
   $w_k^{(i)} = w_k^{(i)} / \sum_{i=1}^n w_k^{(i)}$ .
  Let  $p(x_k|y_{1:k}) \approx \sum_{i=1}^n \tilde{w}_k^{(i)} \delta(x_k - x_k^{(i)})$ 

```

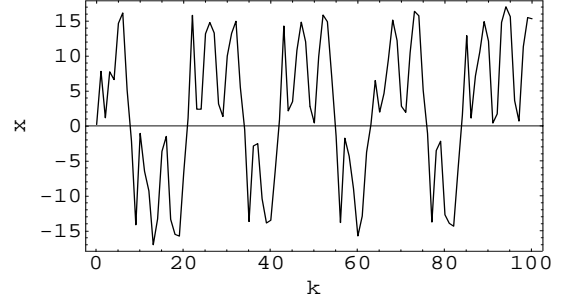
Fig. 1. Algorithm for ESP filters. ESP(+): with the underlined part; ESP(): without the underlined part

of the estimates by the particle filters (SIS ($n = 200$), SIR ($n = 100, N_{eff} = 50$), and the proposed ESP(.) ($n = 100, \ell = 2$) and ESP(+) ($n = 100, \ell = 1$)) are given in Fig.3, and that of EKF as well for comparison. Particle filters, especially SIR and proposed ESP filters, show well behaviors in nonlinear state estimation, while the estimate by EKF cannot follow the true state. Figure4 shows the 2-dimensional plots of squared errors at $k = 1000$ and processing time [s] until $k = 1000$. ESP filters show similar performance as SIR both in squared estimation errors and processing time, and their fluctuations are smaller than SIR. It implies that ESP filters are more stable than SIR.

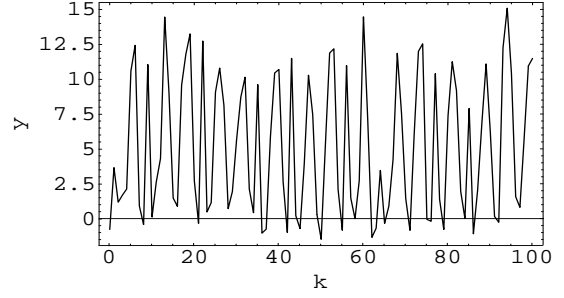
The design parameters, of course, affect the convergence performance. Here, we show the effect of the design parameters in ESP(.) filter. Numerical simu-

Table 1. Design parameters and mean squared errors in ESP(.)

$n \setminus \ell$	2	5	10
10	25.64 \pm 5.89	42.71 \pm 4.56	67.17 \pm 7.80
20	19.29 \pm 4.05	26.87 \pm 4.85	41.30 \pm 6.88
50	13.61 \pm 2.65	15.92 \pm 2.46	22.30 \pm 2.36
100	12.13 \pm 1.78	13.45 \pm 1.99	17.37 \pm 4.19



(a) True state



(b) Observation

Fig. 2. Sample behavior of state and observation processes

lations are carried out for several combinations of n and ℓ of ESP(.), and the mean squared errors in 10 simulation runs are given in Table1. It is suggested that larger n provides the better performance and ℓ should be 2. The other choice of evolution processes, such as inclusion of crossover, may improve the performance, and their better choice will be pursued.

5. CONCLUSIONS

Recognizing the similarities and the differences between the importance sampling and resampling process in SIR filter and mutation and selection processes in ES, we propose a novel particle filters, ESP(.) and ESP(+) filters, by substituting (μ, λ) -selection and $(\mu + \lambda)$ -selection in ES into resampling process in SIR. Introducing of other evolution operations such as crossover and modification of mutation will have the potential to create high performance particle filters.

REFERENCES

- Anderson, B.D.O. and J.B. Moore (1979). *Optimal Filtering*. Prentice-Hall, NJ.
- Arulampalam, S., S. Maskell, N. Gordon and T. Clapp (2002). A tutorial on particle filters for on-line non-linear/non-Gaussian Bayesian tracking. *IEEE Trans. on Signal Processing*, **SP-50**, (2) 174–188.
- Bäck, T. (1996). *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford University Press, London.

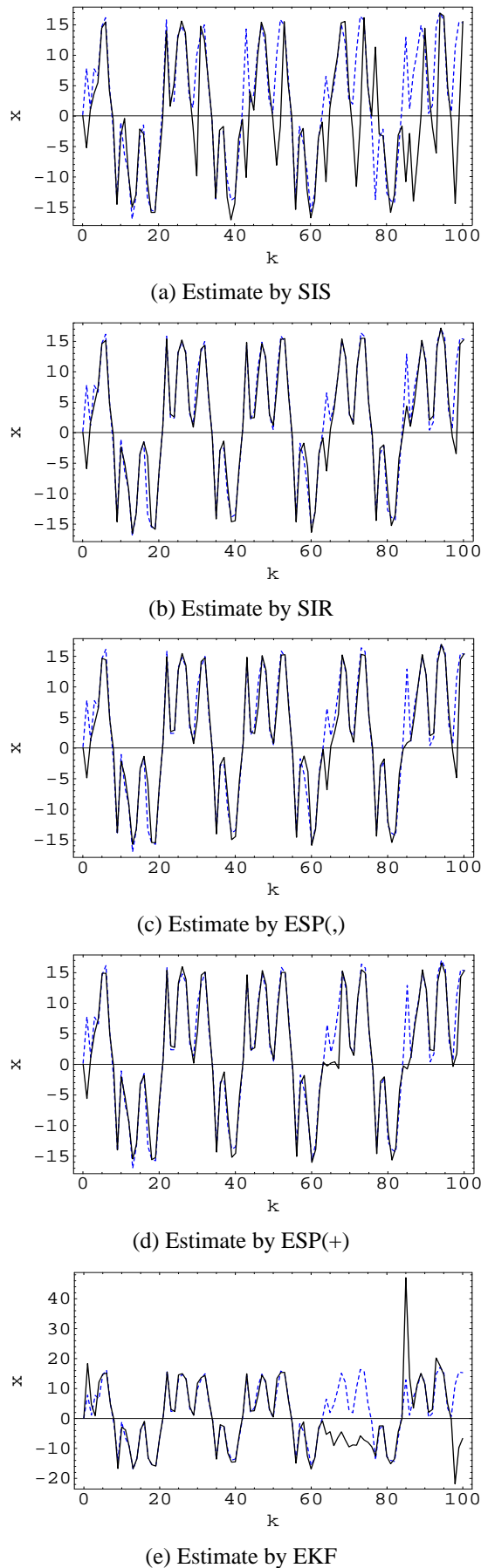


Fig. 3. Sample paths of state estimates (solid line: estimate, dashed line: true state)

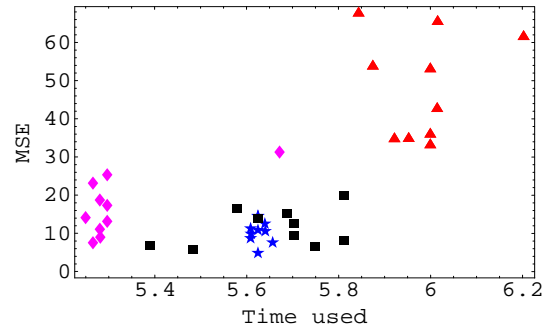


Fig. 4. Squared estimation errors and processing time (triangle: SIS, box: SIR, star: ESP(.), diamond: ESP(+))

Bäck, T. and H.-P. Schwefel (1993). An overview of evolutionary algorithms for parameter optimization. *Evolutionary Computation*, **1**, (1) 1–23.

Doucet, A. (1998). On sequential simulation-based methods for Bayesian filtering. *Technical Report CUED/F-INFENG/TR 310*. Department of Engineering, Cambridge University.

Doucet, A., N. de Freitas and N. Gordon (Eds.) (2001). *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, NY.

Fogel, D. B. (1995). *Evolutionary Computation*. IEEE Press, NJ.

Fogel, L. J., A. J. Owens and M. J. Walsh (1965). Artificial intelligence through a simulation of evolution. *Proc. of the 2nd Cybernetics Science Symp.*, 131–155, reprinted in Fogel, D. B. (eds.), *Evolutionary Computation, The Fossil Record*, 230–254, IEEE Press, NY.

Goodwin, G.C. and J. C. Agüero (2002). State parameter estimation for linear and nonlinear systems. *Proc. 7th International Conf. on Control, Automation, Robotics and Vision*.

Holland, J. H. *Adaptation in Natural and Artificial Systems*. MIT Press.

Jazwinski, A.H. (1970). *Stochastic Process and Filtering Theory*. Academic Press, NY.

Kalman, R. and R. Bucy (1961). New results in linear filtering and prediction theory. *J. of Basic Engineering, Trans. ASME Series D*, **83**, 95–108.

Kong, A., J.S. Liu and W.H. Wong (1994). Sequential imputations and Bayesian missing data problems. *J. of American Statistical Association*, **89**, 278–288.

Koza, J. R. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press.

Schwefel, H.-P. (1995). *Evolution and Optimum Seeking*. J. Wiley, NY.

Sorenson, H. (1985). *Kalman Filtering: Theory and Application*. IEEE Press, NJ.

Storn, R. and K. Price (1995). Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces. *Technical Report TR-95-012*. ICSI.