Novel approach to nonlinear/non-Gaussian Bayesian state estimation

N.J. Gordon
D.J. Salmond
A.F.M. Smith

Indexing terms: Kalman filter, Sequential estimation, Bayesian filter

Abstract: An algorithm, the bootstrap filter, is proposed for implementing recursive Bayesian filters. The required density of the state vector is represented as a set of random samples, which are updated and propagated by the algorithm. The method is not restricted by assumptions of linearity or Gaussian noise: it may be applied to any state transition or measurement model. A simulation example of the bearings only tracking problem is presented. This simulation includes schemes for improving the efficiency of the basic algorithm. For this example, the performance of the bootstrap filter is greatly superior to the standard extended Kalman filter.

1 Introduction

Bayesian methods provide a rigorous general framework for dynamic state estimation problems. The Bayesian approach is to construct the probability density function (PDF) of the state based on all the available information. For the linear-Gaussian estimation problem, the required PDF remains Gaussian at every iteration of the filter, and the Kalman filter relations propagate and update the mean and covariance of the distribution. For nonlinear or non-Gaussian problems there is no general analytic (closed form) expression for the required PDF.

The extended Kalman filter (EKF) is the most popular approach to recursive nonlinear estimation [1]. Here the estimation problem is linearised about the predicted state so that the Kalman filter can be applied. In this case the required PDF is still approximated by a Gaussian, which may be a gross distortion of the true underlying structure and may lead to filter divergence. Other analytic approximations include the Gaussian sum filter [2] and methods based on approximating the first two moments of the density [3, 4]. A more direct numerical approach is to evaluate the required PDF over a grid in state space [5–9]. The choice of an efficient grid is nontrivial, and in a multidimensional state space a very large number of grid points may be necessary. A significant computation must be performed at each point.

© IEE, 1993

Paper 9241F (E5), first received 27th April and in revised form 8th October 1992

N.J. Gordon and D.J. Salmond are with the Defence Research Agency, Farnborough, Hampshire, United Kingdom

A.F.M. Smith is with the Department of Statistics, Imperial College, London, United Kingdom

In this paper we propose a new way of representing and recursively generating an approximation to the state PDF. The central idea is to represent the required PDF as a set of random samples, rather than as a function over state space [10–12]. As the number of samples becomes very large, they effectively provide an exact, equivalent, representation of the required PDF. Estimates of moments (such as mean and covariance) of percentiles of the state vector PDF can be obtained directly from the samples. If necessary, a functional estimate of the PDF could also be constructed from the samples [13] and from this estimates of highest posterior density (HPD) intervals or the mode can be obtained. In Reference 11, an adaptive importance sampling algorithm is presented. This results in large mixture distributions and the technique is computationally expensive.

A recursive algorithm, which we call the bootstrap filter, for propagating and updating these samples for the discrete time problem (see Section 2) is presented and justified in Section 3. The filter is so called because the key update stage of the algorithm (Bayes rule) is implemented as a weighted bootstrap. This was motivated by a result from Smith and Gelfand [14]. A related algorithm is described in the paper by Muller [12]. These sampling techniques avoid the need to define a grid in state space; the samples being naturally concentrated in regions of high probability density. They also have the great advantage of being able to handle any functional nonlinearity, and system or measurement noise of any distribution.

In Section 4, two simulation examples are presented. The second of these is the bearings-only tracking problem. The performance of the bootstrap filter is compared with that of the EKF.

2 Recursive Bayesian estimation

We are concerned with the discrete time estimation problem. The state vector, $x_k \in \mathbb{R}^n$ is assumed to evolve according to the following system model

$$x_{k+1} = f_k(x_k, w_k) \tag{1}$$

where $f_k: \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is the system transition function and $w_k \in \mathbb{R}^m$ is a zero mean, white-noise sequence independent of past and current states. The PDF of w_k is assumed to be known. At discrete times, measurements

This work has been carried out with the support of DRA Farnborough.

$y_k \in \mathbb{R}^p$ become available. These measurements are related to the state vector via the observation equation

$$y_k = h_k(x_k, v_k) \tag{2}$$

where $h_k: \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}^p$ is the measurement function and $v_k \in \mathbb{R}^r$ is another zero mean, white-noise sequence of known PDF, independent of past and present states and the system noise. It is assumed that the initial PDF $p(x_1 \mid D_0) \equiv p(x_1)$ of the state vector is available together with the functional forms f_i and h_i for $i = 1, \ldots, k$. The available information at time step k is the set of measurements $D_k = \{y_i: i = 1, \ldots, k\}$.

The requirement is to construct the PDF of the current state x_k, given all the available information: $p(x_k \mid D_k)$. In principle this PDF may be obtained recursively in two stages: prediction and update. Suppose that the required PDF $p(x_{k-1} \mid D_{k-1})$ at time step $k-1$ is available. Then using the system model it is possible to obtain the prior PDF of the state at time step k

$$p(x_k \mid D_{k-1}) = \int p(x_k \mid x_{k-1}) p(x_{k-1} \mid D_{k-1})\, dx_{k-1} \tag{3}$$

Here the probabilistic model of the state evolution, $p(x_k \mid x_{k-1})$, which is a Markov model, is defined by the system equation and the known statistics of w_{k-1}

$$p(x_k \mid x_{k-1}) = \int p(x_k \mid x_{k-1}, w_{k-1}) p(w_{k-1} \mid x_{k-1})\, dw_{k-1}$$

Since by assumption $p(w_{k-1} \mid x_{k-1}) = p(w_{k-1})$ we have

$$p(x_k \mid x_{k-1}) = \int \delta(x_k - f_{k-1}(x_{k-1}, w_{k-1})) \times p(w_{k-1})\, dw_{k-1} \tag{4}$$

where $\delta(\cdot)$ is the Dirac delta function. The delta function arises because if x_{k-1} and w_{k-1} are known, then x_k is obtained from a purely deterministic relationship (eqn. 1). Then at time step k a measurement y_k becomes available and may be used to update the prior via Bayes rule

$$p(x_k \mid D_k) = \frac{p(y_k \mid x_k) p(x_k \mid D_{k-1})}{p(y_k \mid D_{k-1})} \tag{5}$$

where the normalising denominator is given by

$$p(y_k \mid D_{k-1}) = \int p(y_k \mid x_k) p(x_k \mid D_{k-1})\, dx_k \tag{6}$$

The conditional PDF of y_k given x_k, $p(y_k \mid x_k)$, is defined by the measurement model and the known statistics of v_k

$$p(y_k \mid x_k) = \int \delta(y_k - h_k(x_k, v_k)) p(v_k)\, dv_k \tag{7}$$

In the update equation, eqn. 5, the measurement y_k is used to modify the predicted prior from the previous time step to obtain the required posterior of the state.

The recurrence relations of eqns. 3 and 5 constitute the formal solution to the Bayesian recursive estimation problem. Analytic solutions to this problem are only available for a relatively small and restrictive choice of system and measurement models, the most important being the Kalman filter, which assumes f_k and h_k are linear and w_k and v_k are additive Gaussian noise of known variance (see for example Reference 15 or 16). Considerations of realism imply that these assumptions are unreasonable for many applications. Hence the need to find a method of implementation which allows the models to be constructed realistically rather than conveniently.

3 Bayesian bootstrap filter

3.1 Filter algorithm

Suppose we have a set of random samples $\{x_{k-1}(i): i = 1, \ldots, N\}$ from the PDF $p(x_{k-1} \mid D_{k-1})$. The bootstrap filter is an algorithm for propagating and updating these samples to obtain a set of values $\{x_k(i): i = 1, \ldots, N\}$, which are approximately distributed as $p(x_k \mid D_k)$. Thus the filter is an approximate mechanisation (simulation) of the relations in eqns. 3 and 5.

Prediction: Each sample is passed through the system model to obtain samples from the prior at time step k: $x_k^*(i) = f_{k-1}(x_{k-1}(i), w_{k-1}(i))$, where $w_{k-1}(i)$ is a sample drawn from the PDF of the system noise $p(w_{k-1})$.

Update: On receipt of the measurement y_k, evaluate the likelihood of each prior sample and obtain a normalised weight for each sample

$$q_i = \frac{p(y_k \mid x_k^*(i))}{\sum_{j=1}^{N} p(y_k \mid x_k^*(j))} \tag{8}$$

Thus define a discrete distribution over $\{x_k^*(i): i = 1, \ldots, N\}$, with probability mass q_i associated with element i. Now resample N times from the discrete distribution to generate samples $\{x_k(i): i = 1, \ldots, N\}$, so that for any j, $\Pr\{x_k(j) = x_k^*(i)\} = q_i$.

The above steps of prediction and update form a single iteration of the recursive algorithm. To initiate the algorithm, N samples $x_1^*(i)$ are drawn from the known prior $p(x_1)$. These samples feed directly into the update stage of the filter. We contend that the samples $x_k(i)$ are approximately distributed as the required PDF $p(x_k \mid D_k)$.

3.2 Justification

Prediction: The prediction phase of the algorithm is intuitively reasonable. If $x_{k-1}(i)$ is a sample from $p(x_{k-1} \mid D_{k-1})$ and $w_{k-1}(i)$ is a sample from $p(w_{k-1})$, then $x_k^*(i) = f_{k-1}(x_{k-1}(i), w_{k-1}(i))$ is distributed as $p(x_k \mid D_{k-1})$. Repeating this process for each of the N samples $x_{k-1}(i)$ in turn gives the values $\{x_k^*(i): i = 1, \ldots, N\}$ which are independently distributed as $p(x_k \mid D_{k-1})$.

Update: The justification for the update phase relies on a result from Smith and Gelfand [14]. They show that Bayes theorem can be implemented as a weighted bootstrap. Suppose that samples $\{x_k^*(i): i = 1, \ldots, N\}$ are available from a continuous density function $G(x)$ and that samples are required from the PDF proportional to $L(x)G(x)$, where $L(x)$ is a known function. The theorem states that a sample drawn from the discrete distribution over $\{x_k^*(i): i = 1, \ldots, N\}$ with probability mass $L(x_k^*(i))/\sum L(x_k^*(j))$ on $x_k^*(i)$, tends in distribution to the required density as N tends to infinity. If $G(x)$ is identified with $p(x_k \mid D_{k-1})$ (the prior) and $L(x)$ with $p(y_k \mid x_k)$ (the likelihood), then this theorem provides a justification for the update procedure.

3.3 Discussion

As already noted, the great strength of this technique is that no restrictions are placed on the functions f_k and h_k, or on the distributions of the system or measurement noise. The only requirements are that

(*a*) $p(x_1)$ is available for sampling
(*b*) the likelihood $p(y_k | x_k)$ is a known functional form
(*c*) $p(w_k)$ is available for sampling.

Also the basic algorithm is very simple and easy to program. The resampling update stage is performed by drawing a random sample u_i from the uniform distribution over (0, 1]. The value $x_k^*(M)$ corresponding to

$$\sum_{j=0}^{M-1} q_j < u_i \leqslant \sum_{j=0}^{M} q_j$$

where $q_0 = 0$, is selected as a sample for the posterior. This procedure is repeated for $i = 1, \ldots, N$. It would also be straightforward to implement this algorithm on massively parallel computers, raising the possibility of real time operation with very large sample sets.

The output of the algorithm as a set of samples of the required posterior is convenient for many applications. For example, the posterior probability of the state falling within any region of interest may be estimated by calculating the proportion of samples within that region. Likewise, if there is good reason to believe that the posterior is unimodal, it is easy to estimate the HPD region for each component of the state vector (i.e. for the marginals). It is also straightforward to obtain estimates of the mean and covariance of the state, and indeed any function of the state.

The justification for the bootstrap filter given in Section 3.2 is based on asymptotic results. It is most difficult to prove any general result for a finite number of samples. Likewise it is most difficult to make any precise, provable statement on the crucial question of how many samples are required to give a satisfactory representation of the densities for filter operation. However it is clear that the required number N depends on at least three factors

(*a*) the dimension of the state space
(*b*) the typical 'overlap' between the prior and the likelihood
(*c*) the required number of time steps.

Taking the first of these, one must expect N to rise rapidly with the dimension of the space. The rate of increase is governed by the interdependencies between the components of the state vector. In the most benign case of independent components, the required number of state vector samples should not increase with the dimension of the space.

The dependency of N on factors (*b*) and (*c*) is a direct consequence of the resampling update stage of the filter. If the region of state space where the likelihood $p(y_k | x_k)$ takes significant values is small in comparison with the region where the prior $p(x_k | D_{k-1})$ is significant, many of the samples $x_k^*(i)$ will receive a very small weighting q_i, and will not be selected in the resampling procedure. Thus, samples of the prior remote from the likelihood are effectively wasted, and those nearby are reselected many times. The problem is exacerbated if the likelihood falls in a region of low prior density, where there are relatively few samples (also see Reference 14). Through this process, the representation of the PDF may become most inadequate within a few time steps. Indeed if there is no system noise, all of the N samples may rapidly collapse to a single value. In principle one could overcome this difficulty by the 'brute force' approach of employing an enormous number of samples. However to develop practical filters for interesting problems, a more subtle approach is necessary. In the simulation example of Section 4.2, we suggest and implement two modifications to the basic algorithm for this purpose: a roughening procedure to jitter the resampled values and a prior editing scheme to boost the number of prior samples in the vicinity of significant likelihood. There is much scope for research into the development of such modifications to the basic algorithm to improve efficiency.

The sampling technique proposed by Muller [12] differs from our method in the update stage; the prediction phase being identical. Muller's update scheme is an accept/reject procedure. Each prior sample $x_k^*(i)$ is considered in turn. It is accepted as a posterior sample with a probability proportional to $p(y_k | x_k^*(i))$, otherwise it is rejected. One disadvantage of this scheme is that the sample size is random and decreasing. The main subject of Reference 12 is an algorithm for expanding the prior sample by resampling from an envelope density fitted to the original prior sample.

4 Simulation

We present two examples which illustrate the operation of the bootstrap filter. Estimation performance is compared with the standard EKF. The first example is a univariate nonstationary growth model taken from References 8 and 10. The second is a practical, bearings-only tracking problem, over a four-dimensional state space.

4.1 One-dimensional nonlinear example

Consider the following nonlinear model [8]

$$x_k = 0.5x_{k-1} + 25x_{k-1}/(1 + x_{k-1}^2) + 8 \cos(1.2(k-1)) + w_k \quad (9)$$

$$y_k = x_k^2/20 + v_k \quad (10)$$

where w_k and v_k are zero-mean Gaussian white noise with variances 10.0 and 1.0, respectively. This example is severely nonlinear, both in the system and the measurement equation. Note that the form of the likelihood $p(y_k | x_k)$ adds an interesting twist to the problem. For measurements $y_k < 0$ the likelihood is unimodal at zero and symmetric about zero. However, for positive measurements the likelihood is symmetric about zero with modes at $\pm (20y_k)^{1/2}$.

The initial state was taken to be $x_0 = 0.1$ and Fig. 1 shows a 50 step realisation of eqn. 9. The EKF and

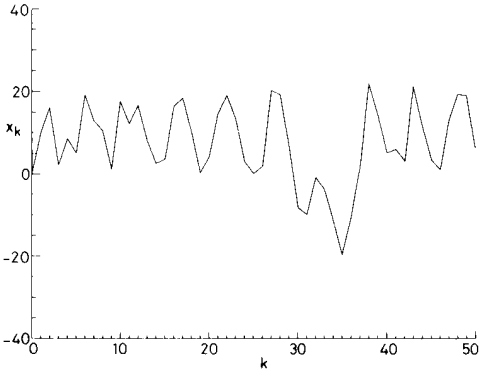

Fig. 1 *50 point realisation of eqn. 9*
Initial state $x_0 = 0.1$

bootstrap filters were both initialised with the prior PDF $p(x_0) = N(0, 2)$. Fig. 2 shows the result of applying the EKF to 50 measurements generated according to eqn. 10.

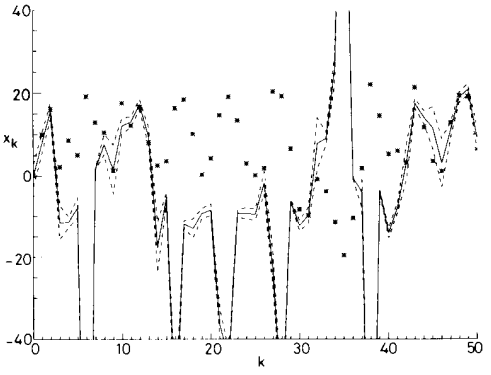

Fig. 2 *EKF estimate of posterior mean*
* True state
——— EKF estimate of posterior mean
------- Estimated limits of 95% probability region

The true state is represented by a star, EKF mean is given as a solid line and the dashed lines give the 95% probability region (note that for convenience we refer to the interval between the 2.5 and 97.5 percentile points as the 95% probability region). The graph has been clipped at ± 40. The actual value of the state is within the 95% probability region on only about 30% of occasions.

Fig. 3 shows the result of directly applying the bootstrap algorithm of Section 3.1 with a sample size of

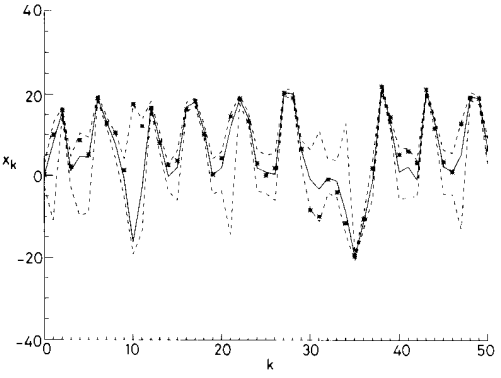

Fig. 3 *Bootstrap filter estimate of posterior mean*
* True state
——— Bootstrap estimate of posterior mean
------- Estimated limits of 95% probability region

$N = 500$. The solid line gives the mean of the posterior samples and the dashed lines estimate the 95% probability region. These may not represent the true 95% HPD region, since, as Fig. 4 shows, the posterior can be bimodal in this example. At a couple of time steps, the actual state is just outside these percentile estimates, and quite often it is close to one of the limits. However, most of the time the actual state is very close to the posterior mean, and performance is obviously greatly superior to the EKF. Running the bootstrap filter with larger sample sets gave results indistinguishable from Fig. 3, and this is taken as confirmation that our sample set size is sufficient. The relatively high system noise probably accounts for the reasonable performance of the basic algorithm using only 500 samples: the system noise automatically roughens the prior samples.

Fig. 4 shows estimates of the posterior density from both the bootstrap filter and the EKF at $k = 21$. The

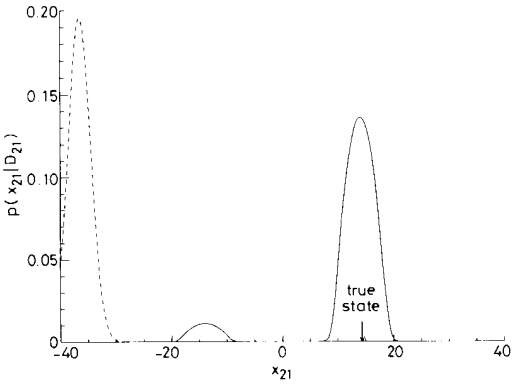

Fig. 4 *Estimated posterior PDF at time step 21*
——— Bootstrap estimate
------- EKF estimate

bootstrap PDF is a kernel density estimate [13] reconstructed from the posterior samples. It has a bimodal structure, with the true value of x_{21} located close to the larger mode. The Gaussian PDF from the EKF is remote from the true state value at this time step.

4.2 Bearings-only tracking example

In this example, the target moves within the x-y plane according to the standard second-order model

$$\boldsymbol{x}_k = \Phi \boldsymbol{x}_{k-1} + \Gamma \boldsymbol{w}_k \tag{11}$$

where $\boldsymbol{x}_k = (x, \dot{x}, y, \dot{y})_k^T$, $\boldsymbol{w}_k = (w_x, w_y)_k^T$

$$\Phi = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \Gamma = \begin{pmatrix} 0.5 & 0 \\ 1 & 0 \\ 0 & 0.5 \\ 0 & 1 \end{pmatrix}$$

Here x and y denote the cartesian co-ordinates of the target: there should be no confusion with the notation for the state vector and measurement vector in our earlier general discussion of the filter. The system noise is a zero mean Gaussian white noise process with covariance Q: $E[\boldsymbol{w}_k \boldsymbol{w}_j^T] = Q\delta_{jk}$, where $Q = qI_2$ and I_2 is the 2×2 identity matrix. A fixed observer at the origin of the plane takes noisy measurements z_k of the target bearing

$$z_k = \tan^{-1}(y_k/x_k) + v_k \tag{12}$$

The measurement noise is a zero mean Gaussian white noise process with variance r: $E[v_k v_j] = r\delta_{kj}$. Before measurements are taken at $k = 1$, the initial state vector is assumed to have a Gaussian distribution with known mean $\bar{\boldsymbol{x}}_1$, and covariance

$$M_1 = \begin{pmatrix} \sigma_1^2 & 0 & 0 & 0 \\ 0 & \sigma_2^2 & 0 & 0 \\ 0 & 0 & \sigma_3^2 & 0 \\ 0 & 0 & 0 & \sigma_4^2 \end{pmatrix}$$

A target trajectory and associated measurements over 24 time steps have been generated according to eqns. 11 and 12 with the parameter values (arbitrary units): $\sqrt{(q)} = 0.001$, $\sqrt{(r)} = 0.005$ and the actual initial state of the target $\boldsymbol{x}_1 = (-0.05, 0.001, 0.7, -0.055)^T$. Fig. 5 gives

the target path in the x-y plane, with the position of the target at each time step being shown by a triangle. The observer is located at the origin and is shown by a cross. Note that with this trajectory realisation, the target initially travels on a gentle curve before passing the observer on a fairly straight course.

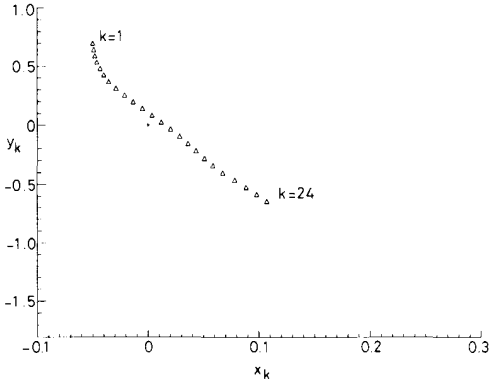

Fig. 5 *True target track in the x-y plane*
△ Target position
\+ Observer position

The prior distribution parameters were set to $\bar{x}_1 = (0.0, 0.0, 0.4, -0.05)^T$ and $\sigma_1 = 0.5$, $\sigma_2 = 0.005$, $\sigma_3 = 0.3$ and $\sigma_4 = 0.01$. The results of applying the EKF and the bootstrap filter with 4000 samples are shown in Fig. 6.

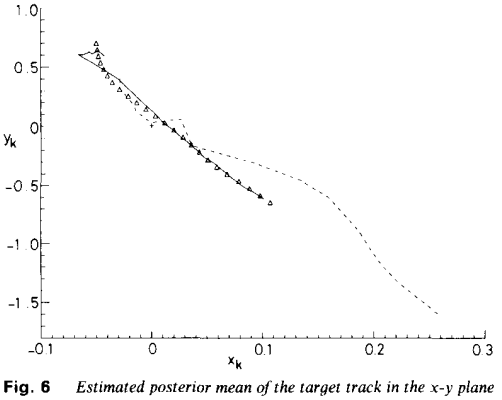

Fig. 6 *Estimated posterior mean of the target track in the x-y plane*
△ Target position
——— Bootstrap estimate
- - - - EKF estimate

The continuous line shows the trajectory estimate (the mean of the posterior PDF) from the bootstrap filter and the dashed line gives the EKF estimate. After an initial period of uncertainty the bootstrap filter quickly homes onto the target, whereas the EKF rapidly diverges. The performance of the bootstrap filter is clearly superior to that of the EKF for this example.

Although the mean of the posterior distribution has been chosen as a point estimate of the target state, there is of course more information available from the bootstrap filter. This is typified by Fig. 7 which shows the (x, y) co-ordinates of 500 out of the 4000 samples from the posterior PDF at time step 24. The true state is shown by a triangle. The skewed nature of the samples towards the lower right corner highlights the non-Gaussian nature of the PDF.

A more detailed analysis of the filter performance for the x and $\dot{x}$ co-ordinates is given in Figs. 8 and 9 for the bootstrap filter, and in Figs. 10 and 11 for the EKF. For the bootstrap filter, the actual co-ordinate value is practically always within the 95% probability region.

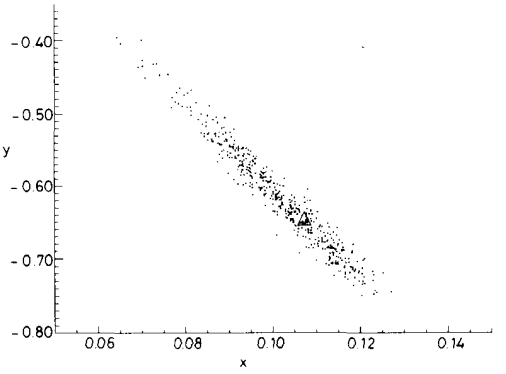

Fig. 7 *Scatterplot of 500 bootstrap x-y samples from* $p(x_{24} | D_{24})$
△ Target position

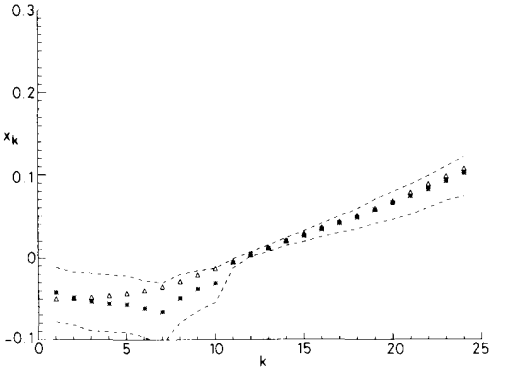

Fig. 8 *Bootstrap estimate of the posterior mean and 95% probability region: x co-ordinate*
△ True target state
* Bootstrap estimated target state
- - - - Estimated limits of 95% probability region

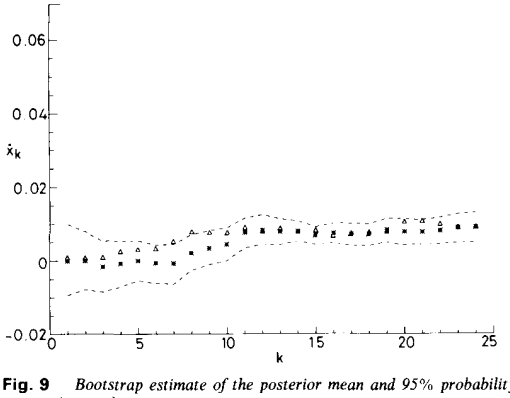

Fig. 9 *Bootstrap estimate of the posterior mean and 95% probability region: $\dot{x}$ co-ordinate*
△ True target state
* Bootstrap estimated target state
- - - - Estimated limits of 95% probability region

However, the EKF is consistently over-optimistic about its tracking performance, and serious divergence occurs after $k = 13$.

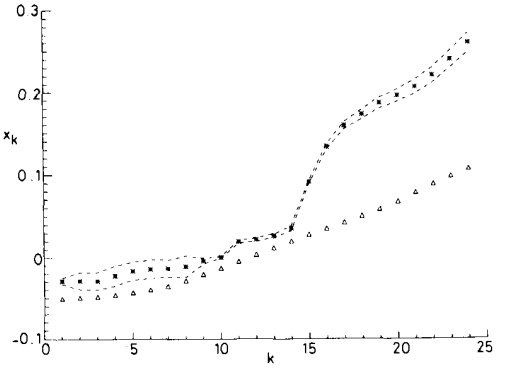

Fig. 10 *EKF estimate of the posterior mean and 95% probability region: x co-ordinate*

△ True target state
* EKF estimated target state
- - - - Estimated limits of 95% probability region

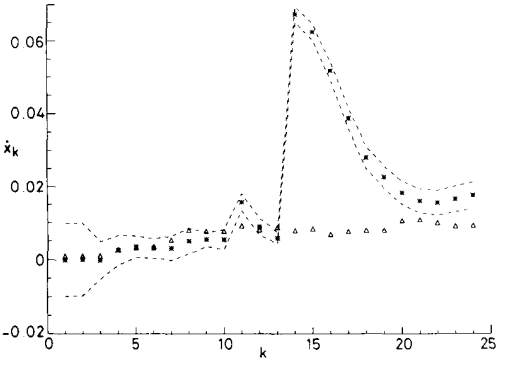

Fig. 11 *EKF estimate of the posterior mean and 95% probability region: $\dot{x}$ co-ordinate*

△ True target state
* EKF estimated target state
- - - - Estimated limits of 95% probability region

The implementation of the bootstrap filter for this example using a modest number of samples requires some ingenuity. This is for reasons discussed in Section 3.3: at the start of the track and as the target approaches the observer, there is only a small 'overlap' between the prior and the likelihood. Two schemes have been implemented for combating the consequent reduction in the number of truly distinct sample values.

Roughening: The first of these is a roughening procedure. An independent jitter c_i is added to each sample drawn in the update procedure. c_i is a sample from $N(0, J_k)$, where J_k is a diagonal covariance matrix. The standard deviation σ of the Gaussian jitter corresponding to a particular component of the state vector is given by $\sigma = KEN^{-1/d}$, where E is the length of the interval between the maximum and the minimum samples of this component (before roughening), d is the dimension of the state space, and K is a constant tuning parameter. By taking the standard deviation of the jitter to be inversely proportional to the dth root of the sample size, the degree of roughening is normalised to the spacing between nodes of the corresponding uniform rectangular grid of N points. In this example, $K = 0.2$ has been chosen. Thus the standard deviation of the Gaussian jitter is 20% of the node spacing on the equivalent rectangular grid. Clearly the choice of K is a compromise. Too large a value would blur the distribution but too small a value would produce tight clusters of points around the original samples.

Prior editing: The second procedure artificially boosts the number of samples of the prior in the vicinity of the likelihood. If one is prepared to delay the state estimate by one time step, this may be achieved by subjecting the samples from the update stage to a coarse, pragmatic acceptance test (which effectively edits the prior):

(*a*) Take the sample $x_k(i)$ and pass it through the roughening procedure and system model to generate $x^*_{k+1}(i)$.

(*b*) Assuming that the measurement from the next time step is available, calculate $v_{k+1}(i) = z_{k+1} - h(x^*_{k+1}(i))$.

(*c*) If $|v_{k+1}(i)| > 6\sqrt{(r)}$, then the sample $x^*_{k+1}(i)$ is remote from the likelihood and so is most unlikely to be selected at the $k + 1$th update stage. In this case reject the sample $x_k(i)$ and repeat step (*a*) to draw a replacement sample from the update stage k. Otherwise accept $x_k(i)$, increment i and repeat if $i \leqslant N$.

Note that this procedure has the effect of a crude, single stage smoothing operation on the samples $x_k(i)$. Thus the accepted samples $x_k(i)$ are vaguely distributed as $p(x_k \mid D_{k+1})$. However, the bootstrap results shown in Figs. 8 and 9 are obtained from samples taken without editing by the acceptance test, and so correctly represent the unsmoothed PDF $p(x_k \mid D_k)$.

By employing these techniques of roughening and of accepting only 'useful' samples, the results presented have been obtained by propagating only $N = 4000$ samples. In four-dimensional space a grid of this size would have only about 8 points on each co-ordinate. Note that the number of samples rejected by the prior editing test is in some sense a measure of the useful information contained in the measurement. The few measurements taken as the target flies past the observer are most informative, and for these time steps the number of rejected samples rose to about 100 000. Before and after the fly-past period, the target is moving approximately along a radius vector of the observer, and so the observed target-bearing is almost constant. During these periods the number of rejected samples is low (between about 10 and 100) as successive measurements contribute little new information. This effect is also clear in Fig. 8: the limits of the 95% probability region converge during fly-past. The EKF performance after time step 15 is especially poor because the filter has seriously misinterpreted the valuable measurements at fly-past.

It should be noted that the EKF results are from a naive application of the filter, directly to the given system and measurement model. A reparameterisation of the problem using modified polar co-ordinates (see Aidala and Hammel [17]), to help separate the observable and unobservable elements of the state, may well have performed better. Also the prior editing test employed with the bootstrap filter has no direct equivalent in the EKF formulation. However, it is possible to perform a gating test on each bearing measurement. For example if an EKF innovation were unexpectedly large, the corresponding measurement could be ignored. With the accep-

tance threshold set to ± 3 standard deviations, 10 of the bearing measurements after the target passed the observer were ignored. Although the EKF was then acting only as a predictor, the estimation error was much reduced. However the actual state was still only rarely within the 95% probability region.

5 Conclusions

A new algorithm, the bootstrap filter, for implementing recursive Bayesian filters has been presented. The required posterior distribution is produced as a set of samples, and the method is not restricted by considerations of analytical tractability.

However for many interesting problems, the number of truly distinct values in the sample set may rapidly collapse. To remedy this difficulty, two somewhat *ad hoc* schemes (a roughening and a prior editing procedure) have been proposed. These procedures have been implemented in the simulation of a nonlinear bearings-only tracking problem with a four-dimensional state vector. The simulation shows that even with a modest number of samples, the bootstrap filter is far superior to the standard extended Kalman filter.

Further work is necessary to derive practical methods, with rigorous theoretical foundations, for avoiding the collapse in the number of distinct values in the sample set. It would also be most useful to develop some guide to the number of samples required for satisfactory bootstrap performance. Further Monte Carlo simulation studies should be performed to provide a quantitative assessment of filter performance for important nonlinear problems.

6 References

1 JAZWINSKI, A.H.: 'Stochastic processes and filtering theory' (Academic Press, New York, 1970)

2 ALSPACH, D.L., and SORENSON, H.W.: 'Nonlinear Bayesian estimation using Gaussian sum approximation', *IEEE Trans. Auto. Control*, 1972, **17**, pp. 439–447

3 MASRELIEZ, C.J.: 'Approximate non-Gaussian filtering with linear state and observation relations', *IEEE Trans. Auto. Control*, 1975, **20**, pp. 107–110

4 WEST, M., HARRISON, P.J., and MIGON, H.S.: 'Dynamic generalised linear models and Bayesian forecasting (with discussion)', *J. Amer. Statistical Assoc.*, 1985, **80**, pp. 73–97

5 BUCY, R.S.: 'Bayes theorem and digital realization for nonlinear filters', *J. Astronaut. Sci.*, 1969, **17**, pp. 80–94

6 KRAMER, S.C., and SORENSON, H.W.: 'Recursive Bayesian estimation using piece-wise constant approximations', *Automatica*, 1988, **24**, (6), pp. 789–801

7 SORENSON, H.W.: 'Recursive estimation for nonlinear dynamic systems', *in* SPALL, J.C. (Ed.): 'Bayesian analysis of time series and dynamic models' (Dekker, 1988)

8 KITTAGAWA, G.: 'Non-Gaussian state-space modelling of nonstationary time series (with discussion)', *J. Amer. Statistical Assoc.*, 1987, **82**, pp. 1032–1063

9 POLE, A., and WEST, M.: 'Efficient numerical integration in dynamic models'. Research report, 136, Department of statistics, University of Warwick, 1988

10 CARLIN, B.P., POLSON, N.G., and STOFFER, D.S.: 'A Monte-Carlo approach to nonnormal and nonlinear state space modelling', *J. Amer. Statistical Assoc.*, 1992, **87**, pp. 493–500

11 WEST, M.: 'Modelling with mixtures (with discussion)', *in* BERNARDO, J.M., BERGER, J.O., DAWID, A.P., and SMITH, A.F.M. (Eds.): 'Bayesian statistics 4' (Oxford University Press, 1992), pp. 503–524

12 MULLER, P.: 'Monte Carlo integration in general dynamic models', *Contemp. Math.*, 1991, **115**, pp. 145–163

13 SILVERMAN, B.W.: 'Density estimation for statistics and data analysis' (Chapman & Hall, London, 1986)

14 SMITH, A.F.M., and GELFAND, A.E.: 'Bayesian statistics without tears: a sampling-resampling perspective', *Amer. Stat.*, 1992, **46**, pp. 84–88

15 HO, Y.C., and LEE, R.C.K.: 'A Bayesian approach to problems in stochastic estimation and control', *IEEE Trans. Auto. Control.*, 1964, **9**, pp. 333–339

16 HARRISON, P.J., and STEVENS, C.F.: 'Bayesian forecasting (with discussion)', *J. R. Stat. Soc. B*, 1976, **38**, pp. 205–247

17 AIDALA, V.J., and HAMMEL, S.E.: 'Utilization of modified polar coordinates for bearings-only tracking', *IEEE Trans. Auto. Control*, 1983, **AC-28**, (3), pp. 283–294