Particle filters for combined state and parameter estimation

Hubert Y. Chan[a] and Michael A. Kouritzin[a]

[a]MITACS-PINTS,
Department of Mathematical Sciences at the University of Alberta,
Edmonton, Canada

ABSTRACT

Filtering is a method of estimating the conditional probability distribution of a signal based upon a noisy, partial, corrupted sequence of observations of the signal. Particle filters are a method of filtering in which the conditional distribution of the signal state is approximated by the empirical measure of a large collection of particles, each evolving in the same probabilistic manner as the signal itself.

In filtering, it is often assumed that we have a fixed model for the signal process. In this paper, we allow unknown parameters to appear in the signal model, and present an algorithm to estimate simultaneously both the parameters and the conditional distribution for the signal state using particle filters. This method is applicable to general nonlinear discrete-time stochastic systems and can be used with various types of particle filters. It is believed to produce asymptotically optimal estimates of the state and the true parameter values, provided reasonable initial parameter estimates are given and further estimates are constrained to be in the vicinity of the true parameters.

We demonstrate this method in the context of search and rescue problem using two different particle filters and compare the effectiveness of the two filters to each other.

Keywords: nonlinear filtering, target tracking, particle methods, system identification

1. INTRODUCTION

We consider a single-target tracking problem in which the signal, X_t, which we wish to track moves according to the discrete-time stochastic equation

$$X_{t+1} = f(\beta_1, X_t) + g(\beta_2, X_t)w_{t+1}, \tag{1}$$

where X_t is an n-dimensional vector representing the signal's state, β_1 and β_2 are d_1- and d_2-dimensional unknown parameter vectors, $f(\cdot,\cdot)$ and $g(\cdot,\cdot)$ are, respectively, $n \times 1$ and $n \times s$ matrices, and w_t is an s-dimensional noise vector with mean 0. We define β to be the $d \mathrel{\hat=} d_1 + d_2$-dimensional vector formed by combining β_1 and β_2, namely $\beta^T = (\beta_1^T, \beta_2^T)$ where β^T means the transpose of β. Our observations of the signal, Y_t, are described by the equation

$$Y_t = h(X_t) + v_t, \tag{2}$$

where Y_t, $h(X_t)$ and v_t are m-dimensional vectors, v_t is the observation noise, with mean zero independent components. Moreover, $\{v_t\}_{t=1}^{\infty}$ is an independent and identically distributed sequence of second order random variables. $h(X_t)$ is often nonlinear and partial in the sense that its value is based only on some of the components of X_t. In our example, we will take v_t and w_t to be Gaussian, although in general, this is not required.

Further author information: (Send correspondence to H.Y.C.)
H.Y.C.: E-mail: hubert@math.ualberta.ca
M.A.K.: E-mail: mkouritz@math.ualberta.ca
PINTS web page: http://www.math.ualberta.ca/pints/

Signal Processing, Sensor Fusion, and Target Recognition X, Ivan Kadar, Editor, Proceedings of SPIE Vol. 4380 (2001) © 2001 SPIE · 0277-786X/01/$15.00

1.1. Particle Filters

Particle filters approximate the conditional distribution of the signal, given the observations, by a finite sum of Dirac measures. They can be applied to any problem for which the signal model can be simulated, hence are useful for problems in which other filtering techniques cannot be applied due to, for example, a high dimensional signal state or non-linear dynamic model. They can be constructed to be asymptotically optimal, often making them a better choice than inherently sub-optimal methods such as the extended Kalman filter, or Interacting Multiple Model methods.

Particle filtering algorithms can be divided into three steps: initializing the particles according to the assumed initial distribution for the signal, evolving each particle independently according to the signal model, and selecting or weighting particles according to their likelihood given the observation, repeating the second and thirds steps at each observation time. The algorithm presented in this paper adds a fourth step to estimate the parameter vectors, using a least-squares method.

2. COMBINED STATE AND PARAMETER ESTIMATION

2.1. Objective

The problem is to simultaneously estimate the conditional distribution for the signal state given the observations, that is, to estimate

$$P(X_t \in dx|\mathcal{Y}_t), \tag{3}$$

where $\mathcal{Y}_t \doteq \sigma\{Y_0, \ldots, Y_t\}$, and to find $\hat{\beta}_t$ such that

$$\hat{\beta}_t = \operatorname*{argmin}_{\beta} \left(\frac{1}{t} \sum_{i=0}^{t-1} \|Y_{i+1} - E^{\beta}(Y_{i+1}|\mathcal{Y}_i)\|^2 \right), \tag{4}$$

where $\|\cdot\|$ is the Euclidean norm, and E^{β} denotes the fact that we take the expectation under the assumption that β is the true parameter value.

However, since we cannot calculate $P(X_t \in dx|\mathcal{Y}_t)$ nor $\operatorname{argmin}_{\beta} \left(\frac{1}{t} \sum_{i=0}^{t-1} \|Y_{i+1} - E^{\beta}(Y_{i+1}|\mathcal{Y}_i)\|^2 \right)$ exactly, we compare our estimates for $E(X_t|\mathcal{Y}_t)$ against the signal's actual state, and our estimate for $\hat{\beta}$ against the known parameters, in order to measure the algorithm's effectiveness.

Particle filtering techniques are asymptotically optimal as the number of particles approaches infinity if the model for the signal is completely known. However, there is currently no known particle method which has been proved to be optimal in estimating both the signal's state and model parameters.

2.2. Particle filters

We consider two particle methods: a weighted particle method used in the analytic work of Kurtz and Xiong,[1] and a branching particle method introduced by Kouritzin and analysed by Blount and Kouritzin.[2]

Particle filtering algorithms consist of three steps: initializing the particles, evolving the particles, and selecting or weighting the particles. The last two steps are repeated after each observation time.

We will denote the set of particles as $\{\xi_t^i\}_{i=1}^{N_t}$, where N_t is the number of particles at time t. If the number of particles is constant, we write N in place of N_t.

Initialization: In the intialization stage, each particle's state is independently initialized according to the assumed initial distribution of the signal.

Evolution: Each particle is evolved according to the signal model, for example, as described in section 3.1. We denote the resulting set of particles $\{\xi_{t+1^-}^i\}_{i=1}^{N_t}$. Hence,

$$\xi_{t+1^-}^i = f(\beta_1, \xi_t^i) + g(\beta_2, \xi_t^i) w_{t+1}^i, \tag{5}$$

where $\{w_{t+1}^i\}_{i=1}^{N_t}$ are independent random variables with the same distribution as w_{t+1}.

Selection: Particles are branched or weighted based on their likelihood given the current observation. Particle methods differ in this step. We denote the resulting set of particles $\{\xi_{t+1}^i\}_{i=1}^{N_{t+1}}$.

For the branching method, each particle, ξ_t^i, is assigned a ζ_t^i value between -1 and 1. If this value is greater than 0, the particle is branched into two independent particles with probability ζ_t^i, and if it is less than 0, the particles is killed with probability $-\zeta_t^i$.

In the weighted method, each particle is assigned a weight, M_t^i. The weights are governed by the stochastic equation

$$M_t^i = M_0^i + \sum_{s=0}^{t} \Delta_t M_s^i \int_U h(\xi_t^i, u) Y(du), \tag{6}$$

where U is the observation space. This is the discrete-time version of the system analysed in Kurtz and Xiong.[1]

State estimation is given by a finite sum of Dirac measures. In the case of the branching particle system, we have, by the analysis in Blount and Kouritzin,[2] for a Borel measurable subset A of the signal's state space,

$$\frac{1}{N_t} \sum_{j=1}^{N_t} \delta_{\xi_t^j}(A) \overset{N_0 \to \infty}{\longrightarrow} P(X_t \in A | \mathcal{Y}_t). \tag{7}$$

In the case of the weighted particle system, we have, by the law of large numbers and conditional independence,

$$\frac{1}{\sum_{j=1}^{N} M_t^j} \sum_{j=1}^{N} M_t^j \delta_{\xi_t^j}(A) \overset{N \to \infty}{\longrightarrow} P(X_t \in A | \mathcal{Y}_t). \tag{8}$$

2.3. Parameter estimation

To perform parameter estimation, we perform some extra initialization and add an extra step after the particle evolution step. We initialize our guess, $\hat{\beta}_0$, at the parameter vector, β, to some good guess, and we set the $d \times d$ (in our case, 3×3) matrix P_0 to some large value. In most practical problems, it is set to KI_d, where I_d is the $d \times d$ identity matrix, and $K > 0$ is a large constant. The intent is that $P_0^{-1} \approx 0$.

In the new parameter estimation step of the algorithm, we update our guess at β as follows. We define

$$\hat{Y}_{t+1} = E^{\beta}[h(X_{t+1}) | \mathcal{Y}_t]|_{\beta = \beta_t}, \tag{9}$$

and approximate as

$$\hat{Y}_{t+1} \approx \frac{1}{N_t} \sum_{i=1}^{N_{t+1}} h(\xi_{t+1-}^i). \tag{10}$$

Next, we define the $d \times m$ matrix φ_t as

$$\varphi_t = E\left[\left(\begin{array}{c} \frac{\partial f^T(\hat{\beta}_t^1, X_t)}{\partial \beta^1} \\ \sum_{j=1}^{s} \frac{\partial [g_j(\beta^2, X_t) w_{t+1}^j]^T}{\partial \beta^2} \end{array} \right) \frac{dh^T(X_{t+1})}{dX} \middle| \mathcal{Y}_t \right], \tag{11}$$

which we approximate as

$$\varphi_t \approx \frac{1}{N_t} \sum_{i=1}^{N_t} \left[\left(\begin{array}{c} \frac{\partial f^T(\hat{\beta}_t^1, \xi_t^i)}{\partial \beta^1} \\ \sum_{j=1}^{s} \frac{\partial [g_j(\beta^2, \xi_t^i) w_{t+1}^{i,j}]^T}{\partial \beta^2} \end{array} \right) \frac{dh^T(\xi_{t+1-}^i)}{dX} \right]. \tag{12}$$

We define

$$P_{t+1} \circeq \left(\sum_{i=0}^{t} \varphi_t \varphi_t^T \right)^{-1}, \tag{13}$$

then update P_t by

$$P_{t+1}^{-1} = P_t^{-1} + \varphi_t \varphi_t^T, \tag{14}$$

which, by the matrix inverse lemma, can be calculated by

$$P_{t+1} = P_t - P_t \varphi_t A_t \varphi_t^T P_t, \tag{15}$$

where

$$A_t = (I_m + \varphi_t^T P_t \varphi_t)^{-1}. \tag{16}$$

Finally, we update our guess at the parameter vector using

$$\hat{\beta}_{t+1} = \hat{\beta}_t + P_t \varphi_t A_t [Y_{t+1} - \hat{Y}_{t+1}]. \tag{17}$$

2.4. Combined algorithm

The resulting algorithm for combined state and parameter estimation is summarized as follows.

Initialization: Each particle's state is independently initialized according to the assumed initial distribution of the signal. β_0 is initialized to some good guess of the parameters, and P_0 is initialized to KI_d.

Evolution: Each particle is evolved according to the signal model. We use our current guess at the parameters in the signal model:

$$\xi^i_{t+1^-} = f(\hat{\beta}^1_t, \xi^i_t) + g(\hat{\beta}^2_t, \xi^i_t) w_{t+1}. \tag{18}$$

Parameter Estimation: Our guess at the parameter vector is updated using

$$\hat{\beta}_{t+1} = \hat{\beta}_t + P_t \varphi_t A_t [Y_{t+1} - \hat{Y}_{t+1}], \tag{19}$$

where $\hat{Y}_{t+1}, A_t, P_t$, and φ_t are calculated via equations 10, 12, 15, and 16.

Selection: Particles are branched or weighted based on their likelihood given the current observation.

The evolution, parameter estimation, and selection steps are repeated at each observation time.

The algorithm is derived in detail in Chan et al.[4]

3. SEARCH AND RESCUE EXAMPLE

In this paper, we use the problem of a dinghy lost at sea, which we observe from above the ocean surface with, for example, a helicopter using an infra-red camera, similar to the problem used by Ballantyne et al.[3]

We assume that the signal's initial location is uniform over the observed ocean surface, and the selection stage is described in further detail for a similar observation function in Ballantyne et al.

3.1. Signal Model

The dinghy has six state components: $x_t, y_t, \theta_t, \dot{x}_t, \dot{y}_t$, and $\dot{\theta}_t$. The variables x_t and y_t represent the x- and y-coordinates of the dinghy's location, and θ_t represents the orientation. $\dot{x}_t, \dot{y}_t$ and $\dot{\theta}_t$ represent the change in x- and y-coordinates, and change in orientation.

We assume that the dinghy only drifts, being pushed around randomly by the waves. We model this by letting the $\dot{x}_t, \dot{y}_t$, and $\dot{\theta}_t$ components be sums of independent zero mean Gaussian random variables. We also add friction terms to the velocities in order to make the model more realistic.

The unknown parameter vectors represent the fact that we do not know the average force of the waves. We model this by defining $\dot{x}_t, \dot{y}_t$ and $\dot{\theta}_t$ as the nominal change, rather than the actual change, in x, y, and θ. In other words, $\dot{x}_t$, multiplied by the unknown parameter, is the actual change in x, and likewise for $\dot{y}_t$ and $\dot{\theta}_t$.

For our problem, then, we let $\beta_1 = [\beta_1^1, \beta_1^2, \beta_1^3]$ be a 3-dimensional vector where β_1^1 is the multiplier for $\dot{x}$, β_1^2 is the multiplier for $\dot{y}$, and β_1^3 is the multiplier for $\dot{\theta}$. We let β_2 be null.

We define the functions f and g used in the model by

$$f(X) = X + \begin{pmatrix} X^{\dot{x}}\beta_1^1\Delta_t \\ X^{\dot{y}}\beta_1^2\Delta_t \\ X^{\dot{\theta}}\beta_1^3\Delta_t \\ \mathcal{F}^{\dot{x}}(X)\Delta_t \\ \mathcal{F}^{\dot{y}}(X)\Delta_t \\ \mathcal{F}^{\dot{\theta}}(X)\Delta_t \end{pmatrix}, \tag{20}$$

where Δ_t is the time step between observations and $\mathcal{F}$ represents friction, and

$$g(X) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{21}$$

Friction is modelled as

$$\mathcal{F}^{\dot{x}}(X) = -f_\ell X^{\dot{x}}\sqrt{D(X^{\dot{x}}, X^{\dot{y}}, X^{\theta})}, \tag{22}$$

$$\mathcal{F}^{\dot{y}}(X) = -f_\ell X^{\dot{y}}\sqrt{D(X^{\dot{x}}, X^{\dot{y}}, X^{\theta})}, \tag{23}$$

$$\mathcal{F}^{\dot{\theta}}(X) = -f_\theta X^{\dot{\theta}}, \tag{24}$$

where

$$D(\dot{x}, \dot{y}, \theta) = \begin{cases} \frac{\dot{x}^2+\dot{y}^2}{(\dot{x}\cos\theta+\dot{y}\sin\theta)^2+\frac{1}{4}(\dot{y}\cos\theta+\dot{x}\sin\theta)^2} & \text{if } \dot{x} \neq 0 \text{ or } \dot{y} \neq 0 \\ 0 & \text{if } \dot{x} = \dot{y} = 0 \end{cases}, \tag{25}$$

and f_ℓ and f_θ are known constants. The result of defining friction in this manner is that friction is increased if the direction of the dinghy's motion is perpendicular do its orientation, and decreased if the direction of the dinghy's motion is parallel to its orientation.

3.2. Target Observations

The observations consist of a sequence of images representing an overhead view of the ocean surface. The images have a higher mean value for pixels which coincide with the polygon representation of the dinghy. The polygon representation of the dinghy, with size parameter s, is constructed as follows:

- Place a square with sides of length $2s$ perpendicular to the raster grid and centred at the point (x, y).
- Add a triangle of height s to the right side of the box so that the base of the triangle is the side of the box.
- Rotate about (x, y) the resulting polygon by the angle θ.

Let S_X be the set of points in the polygon representation of the signal X. Now we define the function $h'_{(\ell,m)}(X)$, where (ℓ, m) describes the location of a pixel in the image, and X is the signal's state, as

$$h'_{(\ell,m)}(X) = \begin{cases} 0 & \text{if } (\ell, m) \notin S_X \\ \Delta_t & \text{if } (\ell, m) \in S_X \end{cases}. \tag{26}$$

We then convolve h' with the Friedrich mollifier to get a function which is differentiable. In particular, we define $\phi_1(X)$ as

$$\phi_1(X) \mathrel{\hat{=}} \begin{cases} c\exp\left(-\frac{1}{1-\|X\|^2}\right) & \text{if } \|X\|^2 < 1 \\ 0 & \text{otherwise} \end{cases}, \tag{27}$$

where c is a constant chosen such that $\int_{-\infty}^{\infty} \phi_1(X)dX = 1$, $\|X\|$ is the Euclidean norm in 3-space: $\|X\| = \sqrt{(X^x)^2 + (X^y)^2 + (X^\theta)^2}$, and let $\phi_\epsilon(X) \doteq \epsilon^{-3}\phi_1(\frac{X}{\epsilon})$ where ϵ is a fixed value. Finally, our observation function $h(X)$ is defined as

$$\begin{aligned} h_{(\ell,m)}(X) &\doteq (h'_{(\ell,m)} * \phi_\epsilon)(X) \\ &= \int \phi(X - \mathcal{X}) h'_{(\ell,m)}(\mathcal{X}) d\mathcal{X}. \end{aligned} \tag{28}$$

3.3. Parameter Estimation

From our definition of f (equation 20),

$$\frac{\partial f^T(\hat{\beta}_t^1, X_t)}{\partial \beta^1} = \begin{pmatrix} X_t^{\dot{x}}\Delta_t & 0 & 0 & 0 & 0 & 0 \\ 0 & X_t^{\dot{y}}\Delta_t & 0 & 0 & 0 & 0 \\ 0 & 0 & X_t^{\dot{\theta}}\Delta_t & 0 & 0 & 0 \end{pmatrix}, \tag{29}$$

and since β^2 is void, so is $\sum_{j=1}^{s} \frac{\partial [g_j(\beta^2, \bar{X}_t^d) w_{t+1}^{i,j}]^T}{\partial \beta^2}$. So our estimation of φ_t, from equation 12, becomes

$$\varphi_t \approx \frac{1}{N}\sum_{i=1}^{N}\left[\begin{pmatrix} \xi_t^{i,\dot{x}}\Delta_t & 0 & 0 & 0 & 0 & 0 \\ 0 & \xi_t^{i,\dot{y}}\Delta_t & 0 & 0 & 0 & 0 \\ 0 & 0 & \xi_t^{i,\dot{\theta}}\Delta_t & 0 & 0 & 0 \end{pmatrix} \frac{dh^T(\xi_{t+1}^i)}{dX}\right]. \tag{30}$$

By our definition of h (equation 28), its derivative is

$$\begin{aligned} \frac{dh_{(\ell,m)}^T}{dX}(X) &= \left(h'_{(\ell,m)}(X)\right)^T * \frac{d\phi(X)}{dX} \\ &= \int \frac{d\phi}{dX}(X - \mathcal{X}) \left(h'_{(\ell,m)}(\mathcal{X})\right)^T d\mathcal{X}, \end{aligned} \tag{31}$$

where

$$\frac{d\phi}{dX}(X) = \begin{cases} -\frac{2c}{(1-\|X\|^2)^2} \exp(-\frac{1}{1-\|X\|^2}) \begin{pmatrix} X^x \\ X^y \\ X^\theta \\ 0 \\ 0 \\ 0 \end{pmatrix} & \text{if } \|X\|^2 < 1 \\ 0 & \text{otherwise} \end{cases}. \tag{32}$$

4. FILTER COMPARISONS

In our comparison test, we take the dinghy size parameter to be $s = 2$, which means that the dinghy area is 20 pixels. The observation area is a 20×20 image. We take the observation noise to be -3.0103dB. We set β to be $[0.5 \quad 0.5 \quad 0.2]$, and our initial guess β_0 to be $[0.45 \quad 0.45 \quad 0.18]$.

A graph showing the mean-squared error in position estimate of the two filters is provided in figure 1, as well as graphs showing the RMS error in the estimates of each of the three parameters in figures 2 – 4. The graphs show the results of only one run.

5. CONCLUSIONS

The algorithm was able to estimate the signal state extremely well, and we were able to show convergence towards the correct parameters.

However, we were limited to use a very small observation size due to the time required to calculate A_t. As a result of this, we were forced to decrease the size of the dinghy, causing the dinghy to move at sub-pixel levels between

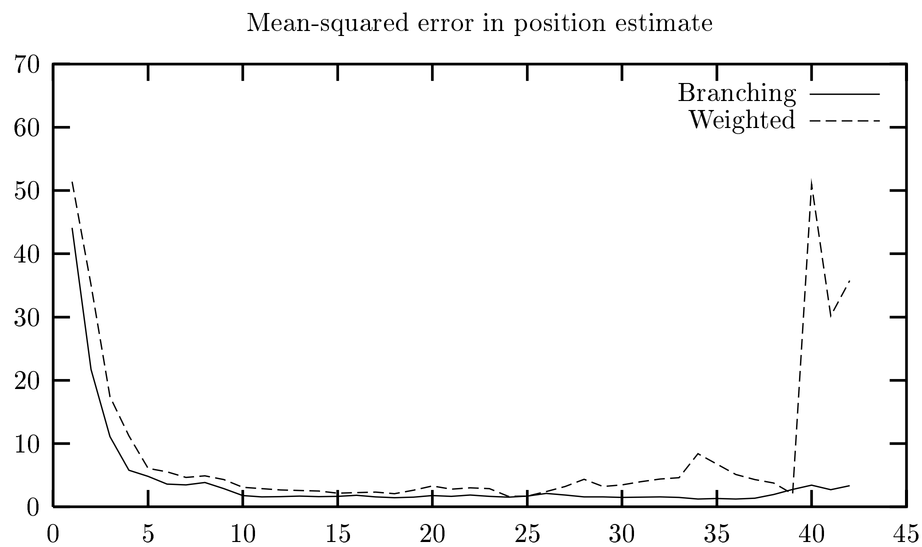

Figure 1. RMS error in estimated position

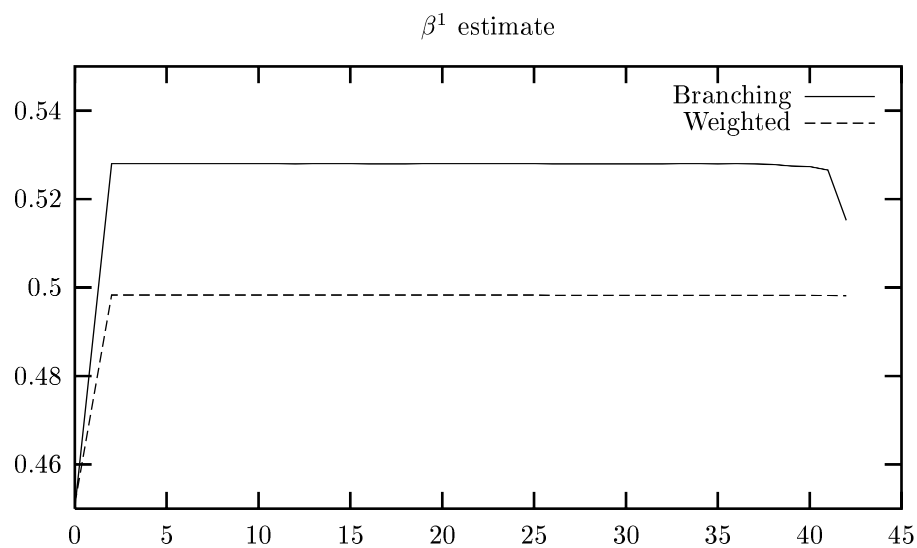

Figure 2. Estimate in β^1 — true value is 0.5

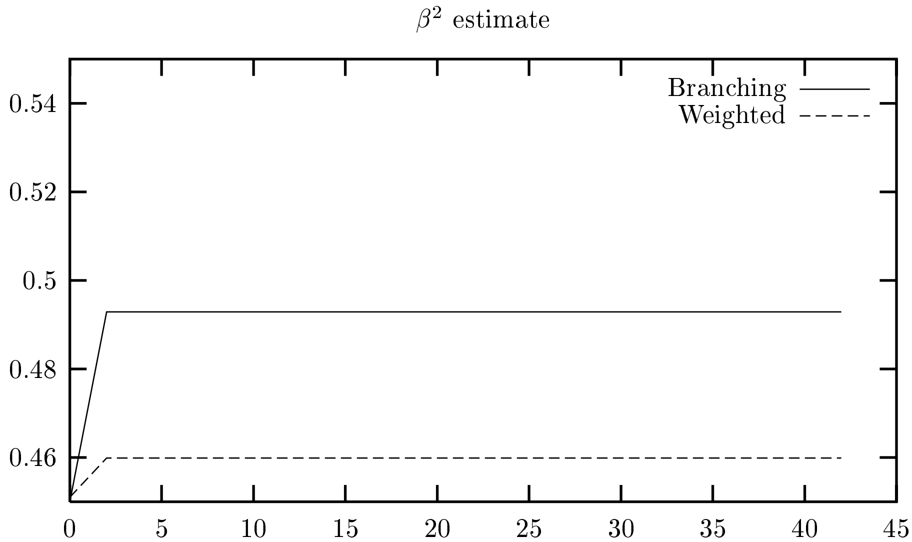

Figure 3. Estimate in β^2 — true value is 0.5

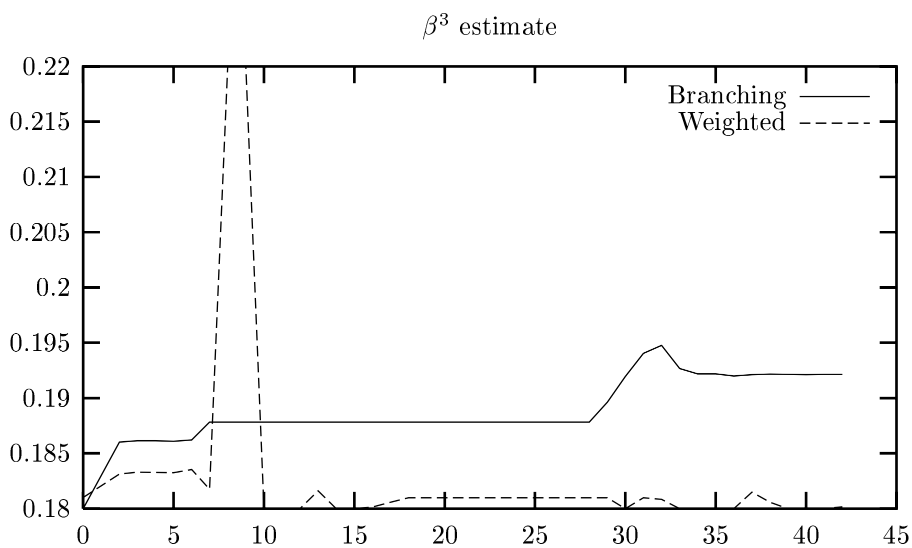

Figure 4. Estimate in β^3 — true value is 0.2

observation times. The effect of this is that the parameters have very little effect on the observation, giving the algorithm very little information. As well, the dinghy would leave the observation area after relatively few frames, which would not allow the algorithm sufficient time to guess the parameters to a sufficient degree of accuracy.

We propose to try to increase the speed of calculating A_t by using a functions of square matrices approach, which should allow us to use a larger observation size and a larger dinghy size. This would also allow us to run the simulations for a longer amount of time, giving better estimates for the parameters, while continuing to give us excellent state estimates.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the support and sponsorship of Lockheed Martin Naval Electronic and Surveillance Systems, Lockheed Martin Canada, VisionSmart of Edmonton, Acoustic Positioning Research, the Pacific Institute of Mathematical Sciences, the Natural Science and Engineering Research Council (NSERC) through the Prediction in Interacting Systems (PINTS) centre of the Mathematics of Information Technology and Complex Systems (MITACS) network of centres of excellence, and the University of Alberta.

REFERENCES

1. T. G. Kurtz and J. Xiong, "Particle representations for a class of nonlinear SPDEs," *Stochastic Processes and their Applications* **83**, pp. 103–126, 1999.
2. D. Blount and M. A. Kouritzin, "On the convergence to multiple timepoint forward equations." Preprint.
3. D. J. Ballantyne, H. Y. Chan, and M. A. Kouritzin, "A novel branching particle method for tracking," in *Signal and Data Processing of Small Targets*, O. E. Drummond, ed., *Proceedings of SPIE* **4048**, pp. 277–287, 2000.
4. H. Chan, M. Kouritzin, and R. Li, "Particle system observers for combined state and parameter estimation." Preprint.