EURASIP Journal on Applied Signal Processing 2004:15, 2278–2294

A New Class of Particle Filters for Random Dynamic Systems with Unknown Statistics

Joaquín Míguez
Departamento de Electrónica e Sistemas, Universidade da Coruña, Facultade de Informática, Campus de Elviña s/n, 15071 A Coruña, Spain
Email: jmiguez@udc.es

Mónica F. Bugallo
Department of Electrical and Computer Engineering, State University of New York at Stony Brook, Stony Brook, NY 11794-2350, USA
Email: monica@ece.sunysb.edu

Petar M. Djurić
Department of Electrical and Computer Engineering, State University of New York at Stony Brook, Stony Brook, NY 11794-2350, USA
Email: djuric@ece.sunysb.edu

Received 4 May 2003; Revised 29 January 2004

In recent years, particle filtering has become a powerful tool for tracking signals and time-varying parameters of random dynamic systems. These methods require a mathematical representation of the dynamics of the system evolution, together with assumptions of probabilistic models. In this paper, we present a new class of particle filtering methods that do not assume explicit mathematical forms of the probability distributions of the noise in the system. As a consequence, the proposed techniques are simpler, more robust, and more flexible than standard particle filters. Apart from the theoretical development of specific methods in the new class, we provide computer simulation results that demonstrate the performance of the algorithms in the problem of autonomous positioning of a vehicle in a 2-dimensional space.

Keywords and phrases: particle filtering, dynamic systems, online estimation, stochastic optimization.

1. INTRODUCTION

Many problems in signal processing can be stated in terms of the estimation of an unobserved discrete-time random signal in a dynamic system of the form

$$\mathbf{x}_t = f_x(\mathbf{x}_{t-1}) + \mathbf{u}_t, \quad t = 1, 2, \ldots, \tag{1}$$

$$\mathbf{y}_t = f_y(\mathbf{x}_t) + \mathbf{v}_t, \quad t = 1, 2, \ldots, \tag{2}$$

where

(a) $\mathbf{x}_t \in \mathbb{R}^{L_x}$ is the signal of interest, which represents the system state at time t;

(b) $f_x : \mathbb{R}^{L_x} \to I_x \subseteq \mathbb{R}^{L_x}$ is a (possibly nonlinear) state transition function;

(c) $\mathbf{u}_t \in \mathbb{R}^{L_x}$ is the state perturbation or system noise at time t;

(d) $\mathbf{y}_t \in \mathbb{R}^{L_y}$ is the vector of observations collected at time t, which depends on the system state;

(e) $f_y : \mathbb{R}^{L_x} \to I_y \subseteq \mathbb{R}^{L_y}$ is a (possibly nonlinear) transformation of the state;

(f) $\mathbf{v}_t \in \mathbb{R}^{L_y}$ is the observation noise vector at time t, assumed statistically independent from the system noise $\mathbf{u}_t$.

Equation (1) describes the dynamics of the system state vector and, hence, it is usually termed *state equation* or *system equation*, whereas (2) is commonly referred to as *observation equation* or *measurement equation*. It is convenient to distinguish the structure of the dynamic system defined by the functions f_x and f_y from the associated probabilistic model, which depends on the probability distribution of the noise signals and the a priori distribution of the state, that is, the statistics of $\mathbf{x}_0$.

We denote the a priori probability density function (pdf) of a random signal s as $p(s)$. If the signal s is statistically dependent on some observation z, then we write the conditional (or a posteriori) pdf as $p(s|z)$. From the Bayesian point of view, all the information relevant for the estimation

of the state at time t is contained in the so-called *filtering pdf*, that is, the a posteriori density of the system state given the observations up to time t,

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}), \tag{3}$$

where $\mathbf{y}_{1:t} = \{\mathbf{y}_1, \ldots, \mathbf{y}_t\}$. The density (3) usually involves a multidimensional integral which does not have a closed-form solution for an arbitrary choice of the system structure and the probabilistic model. Indeed, analytical solutions can only be obtained for particular setups. The most classical example occurs when f_x and f_y are linear functions and the noise processes are Gaussian with known parameters. Then the filtering pdf of $\mathbf{x}_t$ is itself Gaussian, with mean $\mathbf{m}_t$ and covariance matrix $\mathbf{C}_t$, which we denote as

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) = \mathcal{N}(\mathbf{m}_t, \mathbf{C}_t), \tag{4}$$

where the posterior parameters $\mathbf{m}_t$ and $\mathbf{C}_t$ can be recursively computed, as time evolves, using the elegant algorithm known as Kalman filter (KF) [1]. Unfortunately, the assumptions of linearity and Gaussianity do not hold for most real-world problems. Although modified Kalman-like solutions that account for general nonlinear and non-Gaussian settings have been proposed, including the extended KF (EKF) [2] and the unscented KF (UKF) [3], such techniques are based on simplifications of the system dynamics and suffer from severe degradation when the true dynamic system departs from the linear and Gaussian assumptions.

Since general analytical solutions are intractable, Bayesian estimation in nonlinear, non-Gaussian systems must be addressed using numerical techniques. Deterministic approaches, such as classical numerical integration procedures, turn out ineffective or too demanding except for very low-dimensional systems and, as a consequence, methods based on the Monte Carlo methodology have progressively gained momentum. Monte Carlo methods are simulation-based techniques aimed at estimating the a posteriori pdf of the state signal given the available observations. The pdf estimate consists of a random grid of weighted points in the state space $\mathbb{R}^{L_x}$. These points, usually termed *particles*, are Monte Carlo samples of the system state that are assigned nonnegative *weights*, which can be interpreted as probabilities of the particles.

The collection of particles and their weights yield an empirical measure which approximates the continuous a posteriori pdf of the system state [4]. The recursive update of this measure whenever a new observation is available is known as *particle filtering* (PF). Although there are other popular Monte Carlo methods based on the idea of producing empirical measures with random support, for example, Markov Chain Monte Carlo (MCMC) techniques [5], PF algorithms have recently received a lot of attention because they are particularly suitable for real-time estimation. The sequential importance sampling (SIS) algorithm [6, 7] and the bootstrap filter (BF) [8, 9] are the most successful members of the PF class of methods [10]. Existing PF techniques rely on

(i) the knowledge of the probabilistic model of the dynamic system (1)-(2), which includes the densities $p(\mathbf{x}_0)$, $p(\mathbf{u}_t)$, and $p(\mathbf{v}_t)$,

(ii) the ability to numerically evaluate the likelihood $p(\mathbf{y}_t|\mathbf{x}_t)$ and to sample from the transition density $p(\mathbf{x}_t|\mathbf{x}_{t-1})$.

Therefore, the practical performance of PF algorithms in real-world problems heavily depends on how accurate the underlying probabilistic model of choice is. Although this may seem irrelevant in engineering problems where it is relatively straightforward to associate the observed signals with realistic and convenient probability distributions, in many situations, this is not the case. Many times, it is very hard to find an adequate model using the information available in practice. In other occasions, the working models obtained after a lot of effort (involving, e.g., time-series analysis techniques) are so complicated that they render any subsequent signal processing algorithm impractical due to its high complexity.

In this paper, we introduce a new PF approach to deal with uncertainty in the probabilistic modeling of the dynamic system (1)-(2). We start with the requirement that the ultimate objective of PF is to yield an estimate of the signals of interest $\mathbf{x}_{0:t}$, given the observations $\mathbf{y}_{1:t}$. If a suitable probabilistic model is at hand, good signal estimates can be computed from the filtering pdf (3) induced by the model, and a PF algorithm can be employed to recursively build up a random grid that approximates the posterior distribution. However, it is often possible to use signal estimation methods that do not explicitly rely on the a posteriori pdf, for example, blind detection in digital communications can be performed according to several criteria, such as the constrained minimization of the received signal power [11] or the constant modulus method [12, 13]. Such approaches are very popular because they are based on a simple figure of merit, and this simplicity leads to robust and easy-to-design algorithms.

The same advantages in robustness and easy design can be gained in PF whenever a complex (yet possibly not strongly tied to physical reality) probabilistic model can be substituted by a simpler reference for signal estimation. Contrary to initial intuition, estimation techniques based on ad hoc, heuristic, or, simply, alternative references, different from the state posterior distribution, are not precluded by the use of the PF methodology. It is important to realize that the procedure for sequential build-up of the random grid is not tied to the concept of a posteriori pdf. We will show that, by simply specifying a stochastic mechanism for generating particles, the PF methodology can be successfully used to build an approximation of any function of the system state that admits a recursive decomposition. Specifically, we propose a new family of PF algorithms in which the statistical reference of the a posteriori state pdf is substituted by a user-defined cost function that measures the quality of the state signal estimates according to the available observations. Hence, methods within the new class are termed cost-reference particle filters (CRPFs), in contrast to conventional statistical-reference particle filters (SRPFs).

As long as a recursive decomposition of the cost function is found, a PF algorithm, similar to the SIS and bootstrap methods, can be used to construct a random-grid approximation of the cost function in the vicinity of its minima. For this reason, CRPFs yield *local* representations of the cost function specifically built to facilitate the computation of minimum-cost estimates of the state signal.

The remainder of this paper is organized as follows. The fundamentals of the CRPF family are introduced in Section 2. This includes three basic building blocks: the *cost* and *risk* functions, which provide a measure of the quality of the particles, and the stochastic mechanism for particle generation and sequential update of the random grid. Since the usual tools for PF algorithm design (e.g., proposal distributions, auxiliary variables, etc.) do not necessarily extend to the new framework, this section also contains a discussion on design issues. In particular, we identify the factors on which the choice of the cost and risk functions will usually depend, and derive consequent design guidelines, including a useful choice of parameters that leads to a simple interpretation of the algorithm and its connection with the theory of stochastic approximation (SA) [14].

Due to the change in the reference, convergence results regarding SRPFs are not valid for CRPFs. Section 3 is devoted to the problem of identifying sufficient conditions for *asymptotically optimal propagation* (AOP) of particles. The stochastic procedure for drawing new samples of the state signal and propagating the existing particles using the new samples is the key for the convergence of the algorithm. We term this particle propagation step as asymptotically optimal when the increment in the average cost of the particles in the filter after propagation is minimal. A set of sufficient conditions for optimal propagation, related to the properties of the sampling and weighting methods, is provided.

Section 4 is devoted to the discussion of *resampling* in the new family of PF techniques. We argue that the objective of this important algorithmic step is different from its usual role in conventional PF algorithms, and exploit this difference to propose a *local resampling* scheme suitable for a straightforward implementation using parallel VLSI hardware (note that resampling is a major bottleneck for the parallel implementation of PF methods [15]).

Computer simulation results that illustrate the validity of our approach are presented in Section 5. In particular, we tackle the problem of positioning a vehicle that moves along a 2-dimensional space. An instance of the proposed CRPF class of methods that employs a simple cost function is compared with the standard auxiliary BF [9] technique. Finally, Section 6 contains conclusions.

2. COST-REFERENCE PARTICLE FILTERING

The basics of the new family of PF methods are introduced in this section. We start with a general description of the CRPF technique, where key concepts, namely, the cost and risk functions, particle propagation, and particle selection, are introduced. The second part of the section is devoted to practical design issues. We suggest guidelines for the design of CRPFs and propose a simple choice of the algorithm parameters that lead to a straightforward interpretation of the CRPF technique.

2.1. Sequential algorithm

The ultimate aim of the method is the online estimation of the sequence of system states from the available observations, that is, we intend to estimate $\mathbf{x}_t|\mathbf{y}_{1:t}$, $t = 0, 1, 2, \ldots$, according to some reference function that yields a quantitative measure of quality. In particular, we propose the use of a real *cost* function with a recursive additive structure, that is,

$$\mathcal{C}(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}, \lambda) = \lambda \mathcal{C}(\mathbf{x}_{0:t-1}|\mathbf{y}_{1:t-1}, \lambda) + \triangle\mathcal{C}(\mathbf{x}_t|\mathbf{y}_t), \quad (5)$$

where $0 < \lambda < 1$ is a forgetting factor, $\triangle\mathcal{C} : \mathbb{R}^{L_x} \times \mathbb{R}^{L_y} \to \mathbb{R}$ is the *incremental cost* function, and $\mathcal{C}(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}, \lambda)$ complies with the definition

$$\mathcal{C} : \mathbb{R}^{(t+1)L_x} \times \mathbb{R}^{tL_y} \times \mathbb{R} \longrightarrow \mathbb{R}. \quad (6)$$

We should remark that (5) is not the only recursive decomposition that can be employed. A straightforward alternative is to choose a cost function which is built at time t as the convex sum

$$\mathcal{C}(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}, \lambda) = \lambda \mathcal{C}(\mathbf{x}_{0:t-1}|\mathbf{y}_{1:t-1}, \lambda) + (1-\lambda)\triangle\mathcal{C}(\mathbf{x}_t|\mathbf{y}_t). \quad (7)$$

This form of cost function is perfectly valid for the definition and construction of CRPFs and choosing it would not affect (or would affect trivially) the arguments presented in the rest of this paper, including the asymptotic convergence results in Section 3. However, we will constrain ourselves to the familiar form of (5) for simplicity.

A high value of $\mathcal{C}(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}, \lambda)$ means that the state sequence $\mathbf{x}_{0:t}$ is not a good estimate given the sequence of observations $\mathbf{y}_{1:t}$, while a low value of $\mathcal{C}(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}, \lambda)$ indicates that $\mathbf{x}_{0:t}$ is close to the true state signal. The sequence $\mathbf{x}_{0:t}$ is said to have a high cost, in the former case, or a low cost, in the latter case. Particularly notice the recursive structure in (5), where the cost of a sequence up to time $t-1$ can be updated by solely looking at the state and observation vectors at time t, $\mathbf{x}_t$, and $\mathbf{y}_t$, respectively, which are used to compute the cost increment $\triangle\mathcal{C}(\mathbf{x}_t|\mathbf{y}_t)$. The forgetting factor λ avoids attributing an excessive weight to old observations when a long series of data are collected, hence allowing for potential adaptivity.

We also introduce a one-step *risk* function of the form

$$\begin{aligned} &\mathcal{R} : \mathbb{R}^{L_x} \times \mathbb{R}^{L_y} \longrightarrow \mathbb{R}, \\ &\mathbf{x}_{t-1}, \mathbf{y}_t \rightsquigarrow \mathcal{R}(\mathbf{x}_{t-1}|\mathbf{y}_t) \end{aligned} \quad (8)$$

that measures the adequacy of the state at time $t-1$ given the new observation $\mathbf{y}_t$. It is convenient to view the risk function $\mathcal{R}(\mathbf{x}_{t-1}|\mathbf{y}_t)$ as a prediction of the cost increment $\triangle\mathcal{C}(\mathbf{x}_t|\mathbf{y}_t)$ that can be obtained before $\mathbf{x}_t$ is actually propagated. Hence, a natural choice of the risk function is

$$\mathcal{R}(\mathbf{x}_{t-1}|\mathbf{y}_t) = \triangle\mathcal{C}(f_x(\mathbf{x}_{t-1})|\mathbf{y}_t). \quad (9)$$

The proposed estimation technique proceeds sequentially in a similar manner as the BF. Given a set of M state samples and associated costs up to time t, that is, the weighted-particle set (wps)

$$\Xi_t = \{\mathbf{x}_t^{(i)}, \mathcal{C}_t^{(i)}\}_{i=1}^{M}, \tag{10}$$

where $\mathcal{C}_t^{(i)} = \mathcal{C}(\mathbf{x}_{0:t}^{(i)}|\mathbf{y}_{1:t}, \lambda)$, the grid of state trajectories is randomly propagated when $\mathbf{y}_{t+1}$ is observed in order to build an updated wps Ξ_{t+1}. The state and observation signals are those described in the dynamic system (1)-(2). We only add the following mild assumptions:

(1) the initial state is known to lie in a bounded interval $I_{\mathbf{x}_0} \subset \mathbb{R}^{L_x}$;
(2) the system and observation noise are both zero mean.

Assumption (1) is needed to ensure that we initially draw a set of samples that is not infinitely far from the true state $\mathbf{x}_0$. Notice that this is a structural assumption, not a probabilistic one. Assumption (2) is made for the sake of simplicity because zero-mean noise is the rule in most systems.

The sequential CRPF algorithm based on the structure of system (1)-(2), the definitions of cost and risk functions given by (5) and (8), respectively, and assumptions (1) and (2), is described below.

(1) *Time* $t = 0$ *(initialization).* Draw M particles from the uniform distribution in the interval $I_{\mathbf{x}_0}$,

$$\mathbf{x}_0^{(i)} \sim \mathcal{U}(I_{\mathbf{x}_0}), \tag{11}$$

and assign them a zero cost. The initial wps

$$\Xi_0 = \{\mathbf{x}_0^{(i)}, \mathcal{C}_0^{(i)} = 0\}_{i=1}^{M} \tag{12}$$

is obtained.

(2) *Time* $t+1$ *(selection of the most promising trajectories).* The goal of the selection step is to replicate those particles with a low cost while high-cost particles are discarded. As usual in PF methods, selection is implemented by a resampling procedure [7]. We point out that, differently from the standard BF, resampling in CRPFs does not produce equally weighted particles. Instead, each particle preserves its own cost. Notice that the equal weighting of resampled particles in standard PF algorithms comes from the use of a statistical reference. In CRPF, preserving the particle costs after resampling actually shifts the random grid representation of the cost function toward its local minima. Such a behavior is sound, as we are interested in *minimum cost* signal estimates. Further issues related to resampling are discussed in Section 4.

For $i = 1, 2, \ldots, M$, compute the one-step risk of particle i and let

$$\mathcal{R}_{t+1}^{(i)} = \lambda \mathcal{C}_t^{(i)} + \mathcal{R}(\mathbf{x}_t^{(i)}|\mathbf{y}_{t+1}) \tag{13}$$

which yields a predictive cost of the trajectory $\mathbf{x}_{0:t}$ according to the new observation $\mathbf{y}_t$. Define a probability mass function (pmf) of the form

$$\hat{\pi}_{t+1}^{(i)} \propto \mu(\mathcal{R}_{t+1}^{(i)}), \tag{14}$$

where $\mu : \mathbb{R} \to [0, +\infty)$ is a monotonically decreasing function. An intermediate wps is obtained by resampling the trajectories $\{\mathbf{x}_t^{(i)}\}_{i=1}^{M}$ according to the pmf $\hat{\pi}_{t+1}^{(i)}$. Specifically, we select $\hat{\mathbf{x}}_t^{(i)} = \mathbf{x}_t^{(k)}$ with probability $\hat{\pi}_{t+1}^{(k)}$, and build the set $\hat{\Xi}_{t+1} = \{\hat{\mathbf{x}}_t^{(i)}, \hat{\mathcal{C}}_t^{(i)}\}_{i=1}^{M}$, where $\hat{\mathcal{C}}_t^{(i)} = \mathcal{C}_t^{(k)}$ if and only if $\hat{\mathbf{x}}_t^{(i)} = \mathbf{x}_t^{(k)}$.

(3) *Time* $t+1$ *(random particle propagation).* Select an arbitrary conditional pdf of the state $p_{t+1}(\mathbf{x}_{t+1}|\mathbf{x}_t)$ with the constraint that

$$E_{p_{t+1}(\mathbf{x}_{t+1}|\mathbf{x}_t)}[\mathbf{x}_{t+1}] = f_x(\mathbf{x}_t), \tag{15}$$

where $E_{p(s)}[\cdot]$ denotes expected value with respect to the pdf in the subindex. Using the selected propagation density, draw new particles

$$\mathbf{x}_{t+1}^{(i)} \sim p_{t+1}(\mathbf{x}_{t+1}|\hat{\mathbf{x}}_t^{(i)}) \tag{16}$$

and update the associated costs

$$\mathcal{C}_{t+1}^{(i)} = \lambda \hat{\mathcal{C}}_t^{(i)} + \triangle \mathcal{C}_{t+1}^{(i)}, \tag{17}$$

where

$$\triangle \mathcal{C}_{t+1}^{(i)} = \triangle \mathcal{C}(\mathbf{x}_{t+1}^{(i)}|\mathbf{y}_{t+1}) \tag{18}$$

for $i = 1, 2, \ldots, M$.

As a result, the updated wps $\Xi_{t+1} = \{\mathbf{x}_{t+1}^{(i)}, \mathcal{C}_{t+1}^{(i)}\}_{i=1}^{M}$ is obtained.

(4) *Time* $t+1$ *(estimation of the state).* Estimation procedures are better understood if a pmf $\pi_{t+1}^{(i)}$, $i = 1, 2, \ldots, M$, is assigned to the particles in Ξ_{t+1}. The natural way to define this pmf is according to the particle costs, that is,

$$\pi_{t+1}^{(i)} \propto \mu(\mathcal{C}_{t+1}^{(i)}), \tag{19}$$

where μ is a monotonically decreasing function.

The minimum cost estimate at time $t+1$ is trivially computed as

$$\begin{aligned} i_0 &= \arg\max_i \{\pi_{t+1}^{(i)}\}, \\ \tilde{\mathbf{x}}_{0:t+1}^{\min} &= \mathbf{x}_{t+1}^{(i_0)} \end{aligned} \tag{20}$$

and its physical meaning is obvious. An equally useful estimate can be computed as the mean value of $\mathbf{x}_{t+1}^{(i)}$ according to the pmf $\pi_{t+1}^{(i)}$, that is,

$$\tilde{\mathbf{x}}_{t+1}^{\text{mean}} = \sum_{i=1}^{M} \pi_{t+1}^{(i)} \mathbf{x}_{t+1}^{(i)}. \tag{21}$$

Note that $\tilde{\mathbf{x}}_{t+1}^{\text{mean}}$ can also be regarded as a minimum cost estimate because the particle set Ξ_{t+1} is a random-grid local representation of the cost function in the vicinity of its minima. In fact, estimator (21) has slight advantages over (20). Namely, the averaging of particles according to the pmf $\pi_{t+1}^{(i)}$ yields an estimated state trajectory which is smoother than the one resulting from simply choosing the particle with the least cost at each time step. Besides, computing the mean of the particles under $\pi_{t+1}^{(i)}$ may result in an estimate with a slightly smaller cost than the least cost particle, since $\tilde{\mathbf{x}}_{t+1}^{\text{mean}}$ is obtained by interpolation of particles around the least cost state.

Sufficient conditions for the mean estimate (21) to attain an asymptotically minimum cost are given in Section 3.

We will refer to the general procedure described above as a CRPF algorithm. It is apparent that many implementations are possible for a single problem, so in the next section, we discuss the choice of the functions and parameters involved in the method.

2.2. Design issues

An instance of the CRPF class of algorithms is selected by choosing

(i) the cost function $\mathcal{C}(\cdot|\cdot)$,
(ii) the risk function $\mathcal{R}(\cdot|\cdot)$,
(iii) the monotonically decreasing function $\mu : \mathbb{R} \to [0, +\infty)$ that maps costs and risks into the resampling and estimation pmfs, as indicated in (14) and (19), respectively,
(iv) the sequence of pdfs $p_{t+1}(\mathbf{x}_{t+1}|\mathbf{x}_t)$ for particle generation.

The cost and risk functions measure the quality of the particles in the filter. Recall that the risk is conveniently interpreted as a prediction of the cost of a particle, given a new observation, before random propagation is actually carried out (see the selection step in Section 2.1). Therefore, the cost and the risk should be closely related, and we suggest to choose $\mathcal{R}(\cdot|\cdot)$ according to (9). Whenever possible, both the cost and risk functions should be

(i) strictly convex in the range of values of $\mathbf{x}_t$, where the state is expected to lie, in order to avoid ambiguities in the estimators (20) and (21) as well as in the selection (resampling) step,
(ii) easy to compute in order to facilitate the practical implementation of the algorithm,
(iii) dependent on the complete state and observation signals, that is, it should involve all the elements of $\mathbf{x}_t$ and $\mathbf{y}_t$.

A simple, yet useful and physically meaningful, choice of $\mathcal{C}(\cdot|\cdot,\cdot)$, $\mathcal{R}(\cdot|\cdot)$ that will be used in the numerical examples of Section 5 is given by

$$\mathcal{C}(\mathbf{x}_0) = 0, \tag{22}$$

$$\triangle\mathcal{C}(\mathbf{x}_t|\mathbf{y}_t) = ||\mathbf{y}_t - f_y(\mathbf{x}_t)||^q, \tag{23}$$

$$\mathcal{R}(\mathbf{x}_t|\mathbf{y}_{t+1}) = ||\mathbf{y}_{t+1} - f_y(f_x(\mathbf{x}_t))||^q, \tag{24}$$

where $q \geq 1$ and $\|\mathbf{v}\| = \sqrt{\mathbf{v}^{\mathrm{T}}\mathbf{v}}$ denotes the norm of $\mathbf{v}$. Given a fixed and bounded sequence of observations $\mathbf{y}_{1:t}$, the optimal (minimum cost) sequence of state vectors is

$$\begin{aligned}\mathbf{x}_{0:t}^{\text{opt}} &= \arg\min_{\mathbf{x}_{0:t}} \{\mathcal{C}(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}, \lambda)\} \\ &= \arg\min_{\mathbf{x}_{0:t}} \left\{ \sum_{k=0}^{t} \lambda^{(t-k)} \triangle\mathcal{C}(\mathbf{x}_t|\mathbf{y}_t) \right\}.\end{aligned} \tag{25}$$

We call $\mathbf{x}_{0:t}^{\text{opt}}$ optimal because it is obtained by minimization of the continuous cost function, and it is in general different from the minimum cost estimate obtained by CRPF, which we have denoted as $\tilde{\mathbf{x}}_{0:t}^{\min}$ in Section 2.1.

With the assumed choice of cost and risk functions given by (22)–(24), the invertible observation function $f_y : \mathbb{R}^{L_x} \to I_y \subseteq \mathbb{R}^{L_y}$, *and* $\mathbf{y}_t \in I_y$, for all $t \geq 1$, it is straightforward to derive a pointwise solution of the form[1]

$$\mathbf{x}_t^{\text{opt}} = \arg\min_{\mathbf{x}_t} \{\triangle\mathcal{C}(\mathbf{x}_t|\mathbf{y}_t)\} = f_y^{-1}(\mathbf{y}_t). \tag{26}$$

Therefore, as the CRPF algorithm randomly selects and propagates the sample states with the least cost, it can be understood (again, under assumption of (22)–(24)) as a numerical stochastic method for approximately solving the set of (possibly nonlinear) equations

$$\mathbf{y}_t - f_y(\mathbf{x}_t) = \mathbf{0}, \quad t = 1, 2, \ldots. \tag{27}$$

Furthermore, setting $q = 1$ in (23) and (24), we obtain a Monte Carlo estimate of the mean absolute deviation solution of the above set of equations, while $q = 2$ results in a stochastic optimization of the least squares type.

This interpretation of the CRPF algorithm as a method for numerically solving (27) allows to establish a connection between the proposed methodology and the theory of SA [14], which is briefly commented upon in Appendix A.

The function $\mu : \mathbb{R} \to [0, +\infty)$ should be selected to guarantee an adequate discrimination of low-cost particles from those presenting higher costs (recall that we are interested in computing a local representation of the cost function in the vicinity of its minima). As shown in Section 5, the choice of μ has a direct impact on the algorithm performance. Specifically, notice that the accuracy of the selection step is highly dependent on the ability of μ to assign large probability masses to lower-cost particles.

[1] Note, however, that additional solutions may exist at $\nabla_{\mathbf{x}} f_y(\mathbf{x}) = \mathbf{0}$ depending on $f_y(\cdot)$.

A straightforward choice of this function is

$$\mu_1(\mathcal{C}_t^{(i)}) = \mathcal{C}_t^{(i)^{-1}}, \quad \mathcal{C}_t^{(i)} \in \mathbb{R} \setminus \{0\}, \tag{28}$$

which is simple to compute and potentially useful in many systems. It has a serious drawback, however, in situations where the range of the costs, that is, $\max_i\{\mathcal{C}_t^{(i)}\} - \min_i\{\mathcal{C}_t^{(i)}\}$, is much smaller than the average cost $(1/M)\sum_{i=1}^{M}\mathcal{C}_t^{(i)}$. In such scenarios, μ_1 yields nearly uniform probability masses and the algorithm performance degrades. Better discrimination properties can be achieved with an adequate modification of μ_1, for example, with

$$\mu_2(\mathcal{C}_t^{(i)}) = \frac{1}{(\mathcal{C}_t^{(i)} - \min_k\{\mathcal{C}_t^{(k)}\} + \delta)^{\beta}}, \tag{29}$$

where $0 < \delta < 1$ and $\beta > 1$. When compared with μ_1, μ_2 assigns larger masses to low-cost particles and much smaller masses to higher-cost samples. The discrimination ability of μ_2 is enhanced by reducing the value of δ (i.e., $\delta \simeq 0$) and/or increasing β. The relative merit of μ_2 over μ_1 is experimentally illustrated in Section 5.

The last selection to be made is the pdf for particle propagation, $p_{t+1}(\mathbf{x}_{t+1}|\mathbf{x}_t)$, in step 3 of the CRPF algorithm. The theoretical properties required for *optimal* propagation are explored in Section 3. From a practical and intuitive[2] point of view, it is desirable to use easy-to-sample pdfs with a large enough variance to avoid losing tracks of the state signal, but not too large, to prevent the generation of too dispersed particles. A simple strategy implemented in the simulations of Section 5 consists of using zero-mean Gaussian densities with adaptively selected variance. Specifically, the particle i is propagated from time t to time $t+1$ as

$$\mathbf{x}_{t+1}^{(i)} \sim \mathcal{N}\left(f_x(\hat{\mathbf{x}}_{t-1}^{(i)}), \sigma_t^{2,(i)}\mathbf{I}_{L_x}\right), \tag{30}$$

where $\mathbf{I}_{L_x}$ is the $L_x \times L_x$ identity function and the variance $\sigma_t^{2,(i)}$ is recursively computed as

$$\sigma_t^{2,(i)} = \frac{t-1}{t}\sigma_{t-1}^{2,(i)} + \frac{\left\|\mathbf{x}_t^{(i)} - f_x(\hat{\mathbf{x}}_{t-1}^{(i)})\right\|^2}{tL_x}. \tag{31}$$

This adaptive-variance technique has appeared useful and efficient in our simulations, as illustrated in Section 5, but alternative approaches (including the simple choice of a fixed variance) can also be successfully exploited.

3. CONVERGENCE OF CRPF ALGORITHMS

In this section, we assess the convergence of the proposed CRPF algorithm. In particular, we seek sufficient conditions for AOP of the particles from time $t-1$ to time t. Let $\Xi_t = \{\mathbf{x}_t^{(i)}, \mathcal{C}_t^{(i)}\}_{i=1}^{M}$ be the wps computed at time t. We say that Ξ_t has been obtained by AOP from Ξ_{t-1} if and only if

$$\lim_{M\to\infty} \left|\triangle\mathcal{C}(\mathbf{x}_t^{\text{opt}}|\mathbf{y}_t) - \overline{\triangle\mathcal{C}_t}\right| = 0 \quad \text{(in some sense)}, \tag{32}$$

where $\mathbf{x}_t^{\text{opt}}$ is the optimal state according to (26) and

$$\overline{\triangle\mathcal{C}_t} = \sum_{i=1}^{M}\varpi_t^{(i)}\triangle\mathcal{C}_t^{(i)}, \tag{33}$$

with a pmf $\varpi_t^{(i)} \propto \mu(\triangle\mathcal{C}_t^{(i)})$, is the mean incremental cost at time t. The results presented in this section prove that AOP can be ensured by adequately choosing the propagation density and function $\mu : \mathbb{R} \to [0,\infty)$ that relates the cost to the pmf's $\hat{\pi}_t^{(i)}$ and $\pi_t^{(i)}$. Notice that $\pi_t^{(i)} = \varpi_t^{(i)}$ when $\lambda = 0$.

A corollary of the AOP convergence theorem is also established that provides sufficient conditions for the mean state estimate given by (21), for the case $\lambda = 0$, to be asymptotically optimal in terms of its incremental cost.

3.1. Preliminary definitions

Some preliminary definitions are necessary before stating and proving sufficient AOP conditions. If the selection and propagation steps of the CRPF method are considered jointly, it turns out that, at time t, M particles are sampled as

$$\mathbf{x}_t^{(i)} \sim p_t^{M'}(\mathbf{x}), \tag{34}$$

where $M' < \infty$ denotes the number of particles available at time $t-1$ and sampling the pdf $p_t^{M'}(\mathbf{x})$ amounts to resampling M times in $\Xi_{t-1} = \{\mathbf{x}_{t-1}^{(i)}, \mathcal{C}_{t-1}^{(i)}\}_{i=1}^{M'}$ and then propagating the resulting particles and updating the costs to build the new wps $\Xi_t = \{\mathbf{x}_t^{(i)}, \mathcal{C}_t^{(i)}\}_{i=1}^{M}$ (note that we explicitly allow $M \neq M'$). Although other possibilities exist, for example, as described in Section 4, when multinomial resampling is used in the selection step of the CRPF algorithm, the pdf in (34) is a finite mixture of the form

$$p_t^{M'}(\mathbf{x}) = \sum_{k=1}^{M'}\hat{\pi}_t^{(k)}p_t(\mathbf{x}|\mathbf{x}_{t-1}^{(k)}). \tag{35}$$

We also introduce the following notation for a ball centered at $\mathbf{x}_t^{\text{opt}}$ with radius $\varepsilon > 0$:

$$S\{\mathbf{x}_t^{\text{opt}}, \varepsilon\} = \{\mathbf{x} \in \mathbb{R}^{L_x} : \|\mathbf{x} - \mathbf{x}_t^{\text{opt}}\| < \varepsilon\}, \tag{36}$$

and we write

$$S^M\{\mathbf{x}_t^{\text{opt}}, \varepsilon\} = \{\mathbf{x} \in \{\mathbf{x}_t^{(i)}\}_{i=1}^{M} : \|\mathbf{x} - \mathbf{x}_t^{\text{opt}}\| < \varepsilon\} \tag{37}$$

for its discrete counterpart built from the particles in Ξ_t.

[2] Part of this intuition is confirmed by the convergence theorem in Section 3.

3.2. Convergence theorem

Lemma 1. *Let* $\{\mathbf{x}_t^{(i)}\}_{i=1}^M$ *be a set of particles drawn at time* t *using the propagation pdf* $p_t^{M'}(\mathbf{x})$ *as defined by* (34), *let* $\mathbf{y}_{1:t}$ *be a fixed bounded sequence of observations, and let* $\triangle\mathcal{C}(\mathbf{x}|\mathbf{y}_t) \geq 0$ *be a continuous cost function, bounded in* $S\{\mathbf{x}_t^{\text{opt}}, \varepsilon\}$, *with a minimum at* $\mathbf{x} = \mathbf{x}_t^{\text{opt}}$.

If *the three following conditions are met:*

(1) *any ball with center at* $\mathbf{x}_t^{\text{opt}}$ *has a nonzero probability under the propagation density, that is,*

$$\int_{S\{\mathbf{x}_t^{\text{opt}},\varepsilon\}} p_t^{M'}(\mathbf{x})d\mathbf{x} = \gamma > 0 \quad \forall \varepsilon > 0, \tag{38}$$

(2) *the supremum of the function* $\mu(\triangle\mathcal{C}(\cdot|\cdot))$ *for points outside* $S(\mathbf{x}_t^{\text{opt}}, \varepsilon)$ *is a finite constant, that is,*

$$\mathbf{S}_{\text{out}} = \sup_{\mathbf{x}_t \in \mathbb{R}^{Lx} \setminus S(\mathbf{x}_t^{\text{opt}},\varepsilon)} \{\mu(\triangle\mathcal{C}(\mathbf{x}_t|\mathbf{y}_t))\} < \infty, \tag{39}$$

(3) *the supremum of the function* $\mu(\triangle\mathcal{C}(\cdot|\cdot))$ *for points inside* $S^M(\mathbf{x}_t^{\text{opt}}, \varepsilon)$ *converges to infinity faster than the identity function, that is,*

$$\lim_{M\to\infty} \frac{M}{\mathbf{S}_{\text{in}}} = 0, \tag{40}$$

where

$$\mathbf{S}_{\text{in}} = \sup_{\mathbf{x}_t \in S^M(\mathbf{x}_t^{\text{opt}},\varepsilon)} \{\mu(\triangle\mathcal{C}(\mathbf{x}_t|\mathbf{y}_t))\}, \tag{41}$$

then the set function $\mu_t : A \subseteq \{\mathbf{x}_t^{(i)}\}_{i=1}^M \to [0, \infty)$ *defined as*

$$\mu_t(A \subseteq \{\mathbf{x}_t^{(i)}\}_{i=1}^M) = \sum_{\mathbf{x}\in A} \mu(\triangle\mathcal{C}(\mathbf{x}|\mathbf{y}_t)) \tag{42}$$

is an infinite discrete measure (see definition in, e.g., [16]) that satisfies

$$\lim_{M\to\infty} \Pr\left[1 - \frac{\mu_t(S^M(\mathbf{x}_t^{\text{opt}},\varepsilon))}{\mu_t(\{\mathbf{x}_t^{(i)}\}_{i=1}^M)} \geq \delta\right] = 0 \quad \forall \delta > 0, \tag{43}$$

where $\Pr[\cdot]$ *denotes probability, that is,*

$$\lim_{M\to\infty} \frac{\mu_t(S^M(\mathbf{x}_t^{\text{opt}},\varepsilon))}{\mu_t(\{\mathbf{x}_t^{(i)}\}_{i=1}^M)} = 1 \quad \textit{(i.p.)}, \tag{44}$$

where i.p. stands for "in probability."

See Appendix B for a proof.

Theorem 1. *If conditions* (38), (39), *and* (40) *in Lemma 1 hold true, then the mean incremental cost at time* t,

$$\overline{\triangle\mathcal{C}_t} = \sum_{i=1}^M \omega_t^{(i)} \triangle\mathcal{C}(\mathbf{x}_t^{(i)}|\mathbf{y}_t), \tag{45}$$

converges to the minimal incremental cost as $M \to \infty$,

$$\lim_{M\to\infty} \left|\triangle\mathcal{C}(\mathbf{x}_t^{\text{opt}}|\mathbf{y}_t) - \overline{\triangle\mathcal{C}_t}\right| = 0 \quad \textit{(i.p.)}. \tag{46}$$

See Appendix C for a proof.

Finally, an interesting corollary that justifies the use of the mean estimate (21) can be easily derived from Lemma 1 and Theorem 1.

Corollary 1. *Assuming* (38), (39), *and* (40) *in Lemma 1, and forgetting factor* $\lambda = 0$, *the mean cost estimate is asymptotically optimal, that is,*

$$\lim_{M\to\infty} \left|\triangle\mathcal{C}(\tilde{\mathbf{x}}_t^{\text{mean}}|\mathbf{y}_t) - \triangle\mathcal{C}_t(\mathbf{x}_t^{\text{opt}}|\mathbf{y}_t)\right| = 0 \quad \textit{(i.p.)}, \tag{47}$$

where

$$\tilde{\mathbf{x}}_t^{\text{mean}} = \sum_{i=1}^M \pi_t^{(i)} \mathbf{x}_t^{(i)}. \tag{48}$$

See Appendix D for a proof.

3.3. Discussion

Theorem 1 states that conditions (38)–(40) are sufficient to achieve AOP (i.p.). The validity of this result clearly depends on the existence of a propagation pdf, $p_t^{M'}[\cdot]$, and a measure μ_t with *good* properties in order to meet the required conditions.

It is impossible to guarantee that condition (38) holds true in general, as the value of $\mathbf{x}_t^{\text{opt}}$ is a priori unknown, but if the number of particles is large enough and they are evenly distributed on the state space, it is reasonable to expect that the region around $\mathbf{x}_t^{\text{opt}}$ has a nonzero probability. Intuitively, if the wps is locked to the system state at time $t-1$, using the system dynamics to propagate the particles to time t should keep the filtering algorithm locked to the state trajectory. Indeed, our computer simulation experiments give evidence that the propagation pdf is not a critical weakness, and the proposed sequence of Gaussian densities given by (30) and (31) yields a remarkably good performance.

Conditions (39) and (40) are related to the choice of μ or, equivalently, the measure μ_t. For the proposed cost model given by (22) and (23), it is simple to show that condition (39) holds true, both for $\mu = \mu_1$ and $\mu = \mu_2$, as defined in (28) and (29), respectively. The analysis of condition (40) is more demanding and will not be addressed here. An educated intuition, also supported by the computer simulation results in Section 5, points in the direction of selecting $\mu = \mu_2$ with a small enough value of δ.

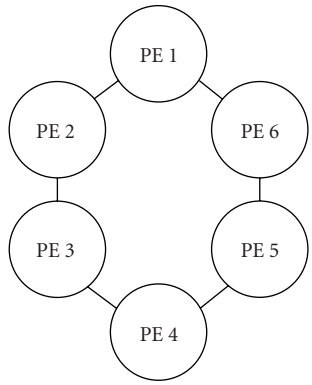

FIGURE 1: $M = 6$ processors in a ring configuration for parallel implementation of the local resampling algorithm.

4. RESAMPLING AND PARALLEL IMPLEMENTATION

Resampling is an indispensable algorithmic component in sequential methods for statistical reference PF, which, otherwise, suffer from weight degeneracy and do not converge to useful solutions [4, 7, 15]. However, resampling also becomes a major obstacle for efficient implementation of PF algorithms in parallel VLSI hardware devices because it creates full data dependencies among processing units [15]. Although some promising methods have been recently proposed [15, 17], parallelization of resampling algorithms remains an open problem.

The selection step in CRPFs (see Section 2.1) is much less restrictive than resampling in conventional SRPFs. Specifically, while resampling methods in SRPFs must ensure that the probability distribution of the resampled population is an unbiased and unweighted approximation of the original distribution of the particles [4], selection in CRPFs is only aimed at ensuring that the particles are close to the locations that produce cost function minima. We have found evidence of state estimates obtained by CRPF being better when the random grid of particles comprises small regions of the state space around these minima. Therefore, selection algorithms can be devised with the only and mild constraint that they do not increase the average cost of particles.

Now we briefly describe a simple resampling technique for CRPFs that lends itself to a straightforward parallelization. Figure 1 shows an array of independent processors connected in a ring configuration. We assume, for simplicity, that the number of processors is equal to the number of particles M, although the algorithm is easily generalized to a smaller number of processing elements (PEs). The ith PE (PE_i) contains the triple $\{\mathbf{x}_t^{(i)}, \mathcal{C}_t^{(i)}, \mathcal{R}_{t+1}^{(i)}\}$ in its memory. The proposed *local resampling* technique proceeds in two steps.

(i) PE_i transmits $\{\mathbf{x}_t^{(i)}, \mathcal{C}_t^{(i)}, \mathcal{R}_{t+1}^{(i)}\}$ to PE_{i+1} and PE_{i-1} and receives the corresponding information from its neighbors. This communication step can be typically carried out in a single cycle and, when complete, PE_i contains three particles $\{\mathbf{x}_t^{(k)}, \mathcal{C}_t^{(k)}, \mathcal{R}_{t+1}^{(k)}\}_{k=i-1}^{i+1}$.

(ii) Each PE draws a single particle with probabilities according to the risks, that is, for the ith PE:

$$\hat{\mathbf{x}}_t^{(i)} = \mathbf{x}_t^{(k)}, \quad \hat{\mathcal{C}}_t^{(i)} = \mathcal{C}_t^{(k)}, \quad k \in \{i-1, i, i+1\}, \tag{49}$$

with probability $\hat{\pi}_t^{(k)} = \mu(\mathcal{R}_{t+1}^{(k)}) / \sum_{l=i-1}^{i+1} \mu(\mathcal{R}_{t+1}^{(k)})$.

Note that, in two simple steps, the algorithm stochastically selects those particles with smaller risks. It is apparent that the method lends easily to parallelization, with very limited communication requirements. The term *local resampling* comes from the observation that low-risk particles are only locally spread by the method, that is, a PE containing a high-risk particle can only get a low-risk sample from its two neighbors.

5. COMPUTER SIMULATIONS

In this section, we present computer simulations that illustrate the validity of our approach. We have considered the problem of autonomous positioning of a vehicle moving along a 2-dimensional space. The vehicle is assumed to have means to estimate its current speed every T_s seconds and it also measures, with the same frequency, the power of three radio signals emitted from known locations and with known attenuation coefficients. This information can be used by a particle filter to estimate the actual vehicle position.

Following [18], we model the positioning problem by the state-space system

(i) state equation:

$$\mathbf{x}_t = \mathbf{G}_x \mathbf{x}_{t-1} + \mathbf{G}_v \mathbf{v}_t + \mathbf{G}_u \mathbf{u}_t; \tag{50}$$

(ii) observation equation:

$$y_{i,t} = 10 \log_{10} \left(\frac{P_{i,0}}{||\mathbf{r}_i - \mathbf{x}_t||^{\alpha_i}} \right) + w_{i,t}, \tag{51}$$

where $\mathbf{x}_t \in \mathbb{R}^2$ indicates the position of the vehicle in the 2-dimensional reference set, $\mathbf{G}_x = \mathbf{I}_2$ and $\mathbf{G}_v = \mathbf{G}_u = T_s \mathbf{I}_2$ are known transition matrices, $\mathbf{v}_t \in \mathbb{R}^2$ is the observable vehicle speed, which is assumed constant during the interval $((t-1)T_s, tT_s)$, and $\mathbf{u}_t$ is a noise process that accounts for measurement errors of the speed. The vector $\mathbf{y}_t = [y_{1,t}, y_{2,t}, y_{3,t}]^T$ collects the received power from three emitters located at known reference locations $\mathbf{r}_i \in \mathbb{R}^2$, $i = 1, 2, 3$, that transmit their signals with initial power $P_{i,0}$ through a fading channel with attenuation coefficient α_i, and, finally, $\mathbf{w}_t = [w_{1,t}, w_{2,t}, w_{3,t}]^T$ is the observation noise. Each time step represents T_s seconds, the position vectors $\mathbf{x}_t$ and $\mathbf{r}_i$ have units of meters (m), the speed is given in m/s, and the received power is measured in dB. The initial vehicle position $\mathbf{x}_0$ is drawn from a standard 2-dimensional Gaussian distribution, that is, $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_2)$.

We have applied the proposed CRPF methodology for solving the positioning problem and, for comparison and benchmarking purposes, we have also implemented the popular auxiliary BF [9], which has an algorithmic structure (resampling, importance sampling, and state estimation) very

Parameters. For all i,
$\lambda = 0.95$; $q = 1, 2$; $\delta = 0.01$; $\beta = 2$; $M = 50$; $\sigma_0^{2,(i)} = 10$.
Initialization. For $i = 1, \ldots, M$,
$\mathbf{x}_0^{(i)} \sim \mathcal{U}(-8, +8)$,
$\mathcal{C}_0^{(i)} = 0$.
Recursive update. For $i = 1, \ldots, M$,
$\mathcal{R}_{t+1}^{(i)} = \lambda \mathcal{C}_t^{(i)} + \|\mathbf{y}_{t+1} - f_y(\mathbf{G}_x \mathbf{x}_t^{(i)} + \mathbf{G}_v \mathbf{v}_{t+1})\|^q$.

Multinomial selection (resampling).

$$\text{pmf} : \hat{\pi}_{t+1}^{(i)} = \begin{cases} \dfrac{(\mathcal{R}_{t+1}^{(i)})^{-1}}{\sum_{l=1}^{M} (\mathcal{R}_{t+1}^{(l)})^{-1}} & \text{(function } \mu_1\text{)}, \\ \dfrac{(\mathcal{R}_{t+1}^{(i)} - \min_{j \in \{1,\ldots,M\}} \mathcal{R}_{t+1}^{(j)} + \delta)^{-\beta}}{\sum_{l=1}^{M} (\mathcal{R}_{t+1}^{(l)} - \min_{j \in \{1,\ldots,M\}} \mathcal{R}_{t+1}^{(j)} + \delta)^{-\beta}} & \text{(function } \mu_2\text{)}. \end{cases}$$

Selection. $(\hat{\mathbf{x}}_t^{(i)}, \hat{\mathcal{C}}_t^{(i)}) = (\mathbf{x}_t^{(k)}, \mathcal{C}_t^{(k)})$, $\quad k \in \{1, \ldots, M\}$, with probability $\hat{\pi}_{t+1}^{(k)}$.

Variance update.
$t \leq 10$: $\sigma_{t+1}^{2,(i)} = \sigma_t^{2,(i)}$,
$t > 10$: $\sigma_t^{2,(i)} = \dfrac{t-1}{t} \sigma_{t-1}^{2,(i)} + \dfrac{\|\mathbf{x}_t^{(i)} - f_x(\hat{\mathbf{x}}_{t-1}^{(i)})\|^2}{t L_x}$.
Let $\mathbf{x}_{t+1}^{(i)} \sim p_{t+1}(\mathbf{x}_{t+1} | \hat{\mathbf{x}}_t^{(i)})$, where
$E_{p_{t+1}(\mathbf{x}_{t+1} | \hat{\mathbf{x}}_t^{(i)})}[\mathbf{x}_{t+1}] = f_x(\hat{\mathbf{x}}_t^{(i)})$,
$\text{Cov}_{p_{t+1}(\mathbf{x}_{t+1} | \hat{\mathbf{x}}_t^{(i)})}[\mathbf{x}_{t+1}] = \sigma_{t+1}^{2,(i)} \mathbf{I}_2$,
$\mathbf{x}_{0:t+1}^{(i)} = \{\hat{\mathbf{x}}_{0:t}^{(i)}, \mathbf{x}_{t+1}^{(i)}\}$,
$\mathcal{C}_{t+1}^{(i)} = \lambda \hat{\mathcal{C}}_t^{(i)} + \|\mathbf{y}_{t+1} - f_y(\mathbf{x}_{t+1}^{(i)})\|^q$.
State estimation.
$\pi_t^{(i)} \propto \mu_1(\mathcal{C}_t^{(i)})$ or $\pi_t^{(i)} \propto \mu_2(\mathcal{C}_t^{(i)})$,
$\bar{\mathbf{x}}_t^{\text{mean}} = \sum_{i=1}^{M} \pi_t^{(i)} x_t^{(i)}$.

Algorithm 1: CRPF algorithm with multinomial resampling for the 2-dimensional positioning problem.

similar to the proposed CRPF family. Algorithm 1 summarizes the details of the CRPF algorithm with multinomial selection, including the alternatives in the choice of function μ. The selection step can be substituted by the local resampling procedure shown in Algorithm 2. A pseudocode for the auxiliary BF is also provided in Algorithm 3.

In the following subsections, we describe different computer experiments that were carried out using synthetic data generated according to model (50)-(51). Two types of plots are presented, both for CRPF and BF algorithms. Vehicle trajectories in the 2-dimensional space, resulting from a single simulation of the dynamic system, are shown to illustrate the ability of the algorithms to remain locked to the state trajectory. We chose the mean absolute deviation as a performance figure of merit. It was measured between the true vehicle trajectory in $\mathbb{R}^2$ and the trajectory estimated by the particle filters and its unit was meter. All mean-deviation plots were obtained by averaging 50 independent simulations. Both the BF and the CRPF type of algorithms were run with $M = 50$ particles.

5.1. Mixture Gaussian noise processes

In the first experiment, we modeled the system and observation noise processes $\mathbf{u}_t$ and $\mathbf{w}_t$, respectively, as independent and temporally white, with the mixture Gaussian pdfs:

$$\begin{aligned} \mathbf{u}_t &\sim 0.3\mathcal{N}(\mathbf{0}, \sqrt{0.2}\mathbf{I}_2) \\ &\quad + 0.4\mathcal{N}(\mathbf{0}, \mathbf{I}_2) + 0.3\mathcal{N}(\mathbf{0}, \sqrt{10}\mathbf{I}_2), \\ w_{l,t} &\sim 0.3\mathcal{N}(0, 0.2) + 0.4\mathcal{N}(0, 1) \\ &\quad + 0.3\mathcal{N}(0, 10), \quad l = 1, 2, 3. \end{aligned} \tag{52}$$

In Figure 2, we compare the auxiliary BF with perfect knowledge of the noise distributions, and several CRPF algorithms that use the cost and risk functions proposed in Section 2.2 (see (22)–(24)). For all CRPF methods, the forgetting factor was $\lambda = 0.95$, but we ran algorithms with different values of q, $q = 1, 2$, and functions μ_1 and μ_2 (see (28) and (29)). For the latter function μ_2, we set $\delta = 0.01$ and $\beta = 2$. The propagation mechanism for the CRPF methods consisted of the sequence of Gaussian densities given by (30) and (31), with initial value $\sigma_0^{2,(i)} = 10$ for all i.

Figure 2a shows the system trajectory in a single run and the estimates corresponding to the BF and CRPF algorithms. The trajectory started in an unknown position close to $(0, 0)$ and evolved for one hour, with sampling period $T_s = 2$ seconds. It is apparent that all the algorithms remained locked to the vehicle position during the whole simulation interval.

Local selection (resampling) at the ith PE.

$$\text{For } k = i-1, i, i+1,\ \hat{\pi}_{t+1}^{(k)} = \begin{cases} \dfrac{\left(\mathcal{R}_{t+1}^{(k)}\right)^{-1}}{\sum_{l=i-1}^{i+1}\left(\mathcal{R}_{t+1}^{(l)}\right)^{-1}} & (\mu_1), \\ \dfrac{\left(\mathcal{R}_{t+1}^{(k)} - \min_{j\in\{i-1,i,i+1\}}\mathcal{R}_{t+1}^{(j)} + \delta\right)^{-\beta}}{\sum_{l=i-1}^{i+1}\left(\mathcal{R}_{t+1}^{(l)} - \min_{j\in\{i-1,i,i+1\}}\mathcal{R}_{t+1}^{(j)} + \delta\right)^{-\beta}} & (\mu_2). \end{cases}$$

Selection. $(\hat{\mathbf{x}}_t^{(i)}, \hat{\mathcal{C}}_t^{(i)}) = (\mathbf{x}_t^{(l)}, \mathcal{C}_t^{(l)})$, $l \in \{i-1, i, i+1\}$, with probability $\hat{\pi}_{t+1}^{(l)}$.

Algorithm 2: Local resampling for the CRPF algorithm.

Initialization. For $i = 1, \ldots, M$,

$\mathbf{x}_0^{(i)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_2)$,

$w_0^{(i)} = \dfrac{1}{M}$.

Recursive update.

For $t = 1, \ldots, K$,

For $i = 1, \ldots, M$,

$\hat{\mathbf{x}}_t^{(i)} = f_x(\mathbf{x}_{t-1}^{(i)})$,

$\kappa_i = k$, with probability $p(\mathbf{y}_t|\hat{\mathbf{x}}_t^{(k)})w_{t-1}^{(k)}$,

$\mathbf{x}_t^{(i)} \sim p[\mathbf{x}_t|\mathbf{x}_{t-1}^{(\kappa_i)}]$.

Weight update. $\tilde{w}_t^{(i)} = \dfrac{p(\mathbf{y}_t|\mathbf{x}_t^{(i)})}{p(\mathbf{y}_t|\hat{\mathbf{x}}_t^{(\kappa_i)})}$.

Weight normalization. $w_t^{(i)} = \dfrac{\tilde{w}_t^{(i)}}{\sum_{k=1}^{M}\tilde{w}_t^{(k)}}$.

Algorithm 3: Auxiliary BF for the 2-dimensional positioning problem.

The latter observation is confirmed by the mean absolute deviation plot in Figure 2b. The deviation signal was computed as

$$e_t = \frac{1}{50}\frac{1}{2}\sum_{j=1}^{50}\left|x_{1,t,j} - x_{1,t,j}^{\text{est}}\right| + \left|x_{2,t,j} - x_{2,t,j}^{\text{est}}\right|, \tag{53}$$

where j is the simulation number, $\mathbf{x}_{t,j} = [x_{1,t,j}, x_{2,t,j}]^{\mathrm{T}}$ is the true position at time t, and $\mathbf{x}_{t,j}^{\text{est}} = [x_{1,t,j}^{\text{est}}, x_{2,t,j}^{\text{est}}]^{\mathrm{T}}$ is the corresponding estimate obtained with the particle filter. We observe that the CRPF algorithms with μ_2 attained the lowest deviation and outperformed the auxiliary BF. Although it is not shown here, the auxiliary BF improved its performance as the sampling period was decreased,[3] and achieved a lower deviation than the CRPFs for $T_s \leq 0.5$ second. The reason is that, as T_s decreases, the correlation of the states increases due to the variation of $\mathbf{G}_u$, and the BF exploits this statistical information better. Therefore, we can conclude that the BF can be more accurate when strong statistical information is available, and that the proposed CRPFs are more robust and steadily attain a good performance for a wider range of scenarios. This conclusion is easily confirmed with the remaining experiments presented in this section.

Figures 3 and 4 show the trajectories and the mean absolute deviations for the BF and CRPF algorithms when the sampling period was increased to $T_s = 5$ seconds and $T_s = 10$ seconds, respectively. Note that increasing T_s also increases the speed measurement error. As before, the CRPF techniques with μ_2 outperformed the BF in the long term.

Because of its better performance, we also checked the behavior of the CRPF method that uses μ_2 for different values of parameter δ. Figure 5a shows the true position and the estimates obtained using three different values of δ, namely, 0.1, 0.01, and 0.001, with fixed $\beta = 2$. All the algorithms appear to perform similarly for the considered range of values. This is confirmed with the results presented in Figure 5b in terms of the mean absolute deviation. They also illustrate the robustness and stability of the method.

In the following, unless it is stated differently, the CRPF algorithm was always implemented with μ_2 and parameters $q = 2$, $\delta = 0.01$, and $\beta = 2$. The sampling period was also fixed and was $T_s = 5$ seconds.

5.2. Mixture Gaussian system and observation noise—Gaussian BF

Figure 6 shows the results (trajectory and mean deviation) obtained with the same system and observation noise distributions as in Section 5.1 when the auxiliary BF (labeled as BF (Gaussian)) is mismatched with the dynamical system and models the noise processes with Gaussian densities:

$$\begin{aligned} p[\mathbf{u}_t] &= \mathcal{N}(\mathbf{0}, \sqrt{0.2}\mathbf{I}_2), \\ p[w_{l,t}] &= \mathcal{N}(0, 0.2), \quad l = 1, 2, 3. \end{aligned} \tag{54}$$

It is apparent that the use of the correct statistical information is critical for the bootstrap algorithm (in the figure, we also plotted the result obtained when the BF used the true mixture Gaussian density—labeled as BF (M-Gaussian)). Note that the CRPF algorithm also drew the state particles from a Gaussian sequence of densities (see Section 2.2), but it attained a superior performance compared to the BF.

5.3. Local versus multinomial resampling

We have verified the performance of the CRPF that uses the new resampling scheme proposed in Section 4. The results

[3] Obviously, the BF will also deliver a better performance as the number of particles M grows. In fact, it can be shown [4] that the estimate of the posteriori pdf and its moments obtained from the BF converge uniformly to the true density and the true values of the moments. This means that, as $M \to \infty$, the state estimates given by the BF become optimal (in the mean square error sense) and that for large M, the BF will outperform the CRPF algorithm.

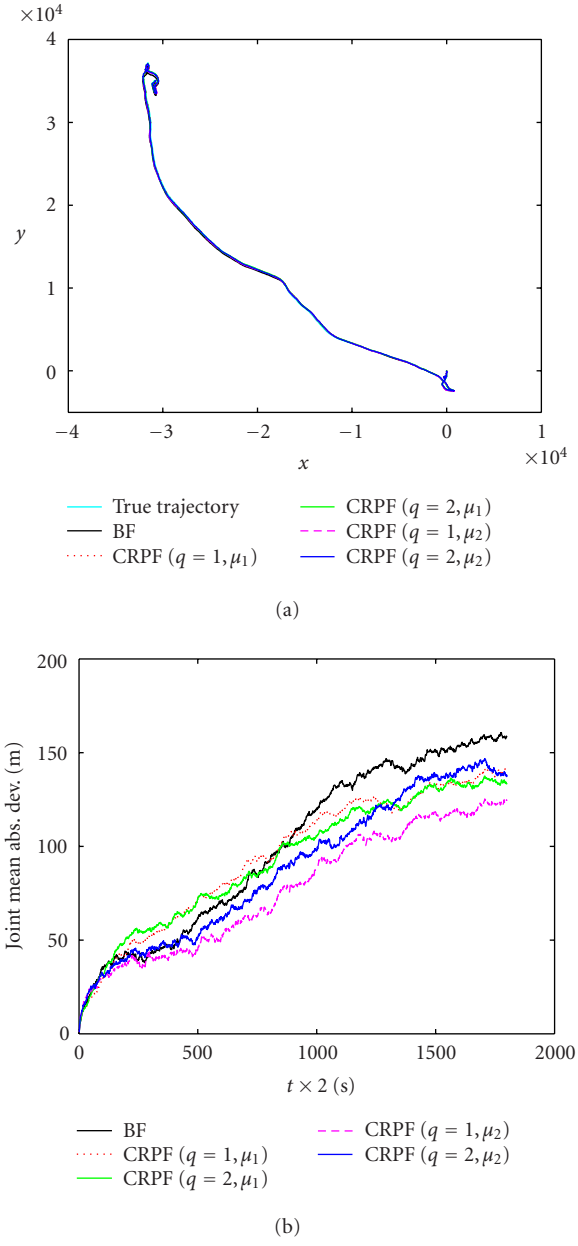

FIGURE 2: Mixture Gaussian noise processes. $T_s = 2$ seconds. (a) Trajectory. (b) Mean absolute deviation.

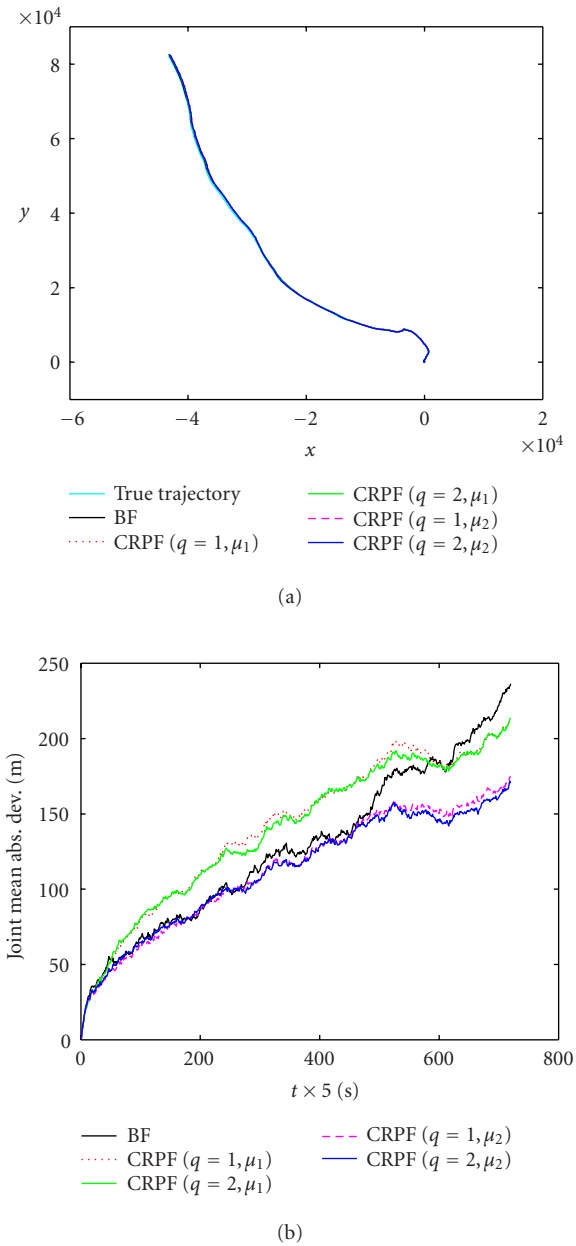

FIGURE 3: Mixture Gaussian noise processes. $T_s = 5$ seconds. (a) Trajectory. (b) Mean absolute deviation.

can be observed in Figure 7. The CRPF with local resampling shows approximately the same performance as the BF with perfect knowledge of the noise statistics. Although it presents a slight degradation with respect to the CRPF with multinomial resampling, the feasibility of a simple parallel implementation makes the local resampling method extremely appealing.

5.4. Different estimation criteria

Figure 8 compares the trajectory and mean deviation of two CRPF algorithms that used different criteria to obtain the estimates of the state: the minimum cost estimate $\tilde{\mathbf{x}}_t^{\min}$ (see (20)) and the mean cost estimate $\tilde{\mathbf{x}}_t^{\text{mean}}$ (see (21)). It is clear that both algorithms performed similarly and outperformed the BF in the long term.

5.5. Laplacian noise

Finally, we have repeated our experiment by modeling the noises using Laplacian distributions, that is,

$$p[\mathbf{u}_t] = \mathcal{L}(\mathbf{0}, \sqrt{0.5}\mathbf{I}_2) = \frac{1}{0.5} e^{-|\mathbf{u}_t|/0.5},$$
$$p[w_{l,t}] = 0.3\mathcal{L}(0, 0.5) = \frac{1}{0.5} e^{-|w_{l,t}|/0.5}, \quad l = 1, 2, 3. \tag{55}$$

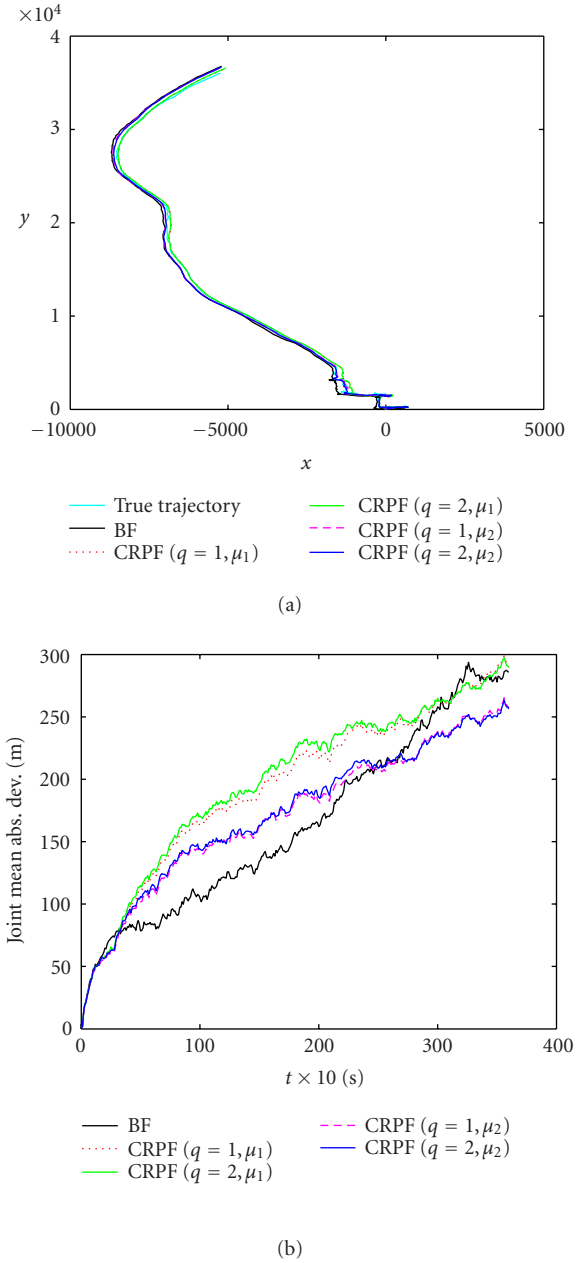

FIGURE 4: Mixture Gaussian noise processes. $T_s = 10$ seconds. (a) Trajectory. (b) Mean absolute deviation.

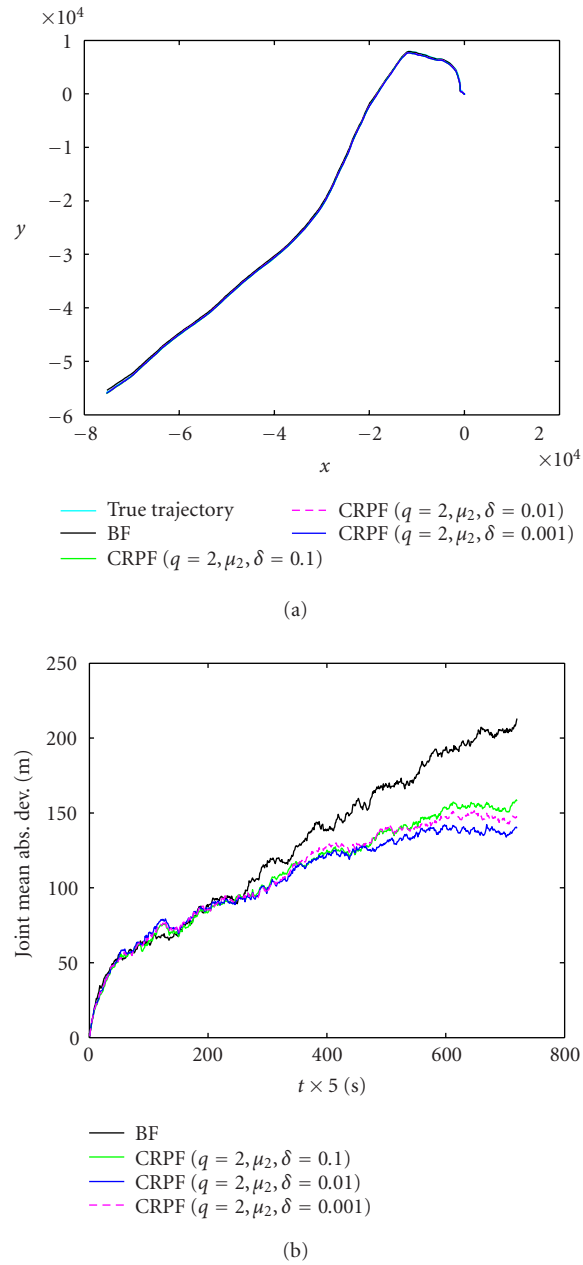

FIGURE 5: Different δ values. $T_s = 5$ seconds. (a) Trajectory. (b) Mean absolute deviation.

Figure 9 depicts the results obtained for the BF with perfect knowledge of the probability distribution of the noise and the CRPF algorithm. Again, the proposed method attained better performance in terms of mean absolute deviation.

6. CONCLUSIONS

Particle filters provide optimal numerical solutions in problems that amount to estimation of unobserved time-varying states of dynamic systems. Such methods rely on the knowledge of prior probability distributions of the initial state and noise processes that affect the system, and require the ability to evaluate likelihood functions and the state transition densities. Under these assumptions, different methods have been proposed that recursively estimate posterior densities by generating a collection of samples and associated importance weights. In this paper, we introduced a new class of particle filtering methods that aim at the estimation of system states from available observations *without a priori knowledge of any probability density functions*. The proposed method is based on cost functions that measure the quality of the state signal estimates given the available observations.

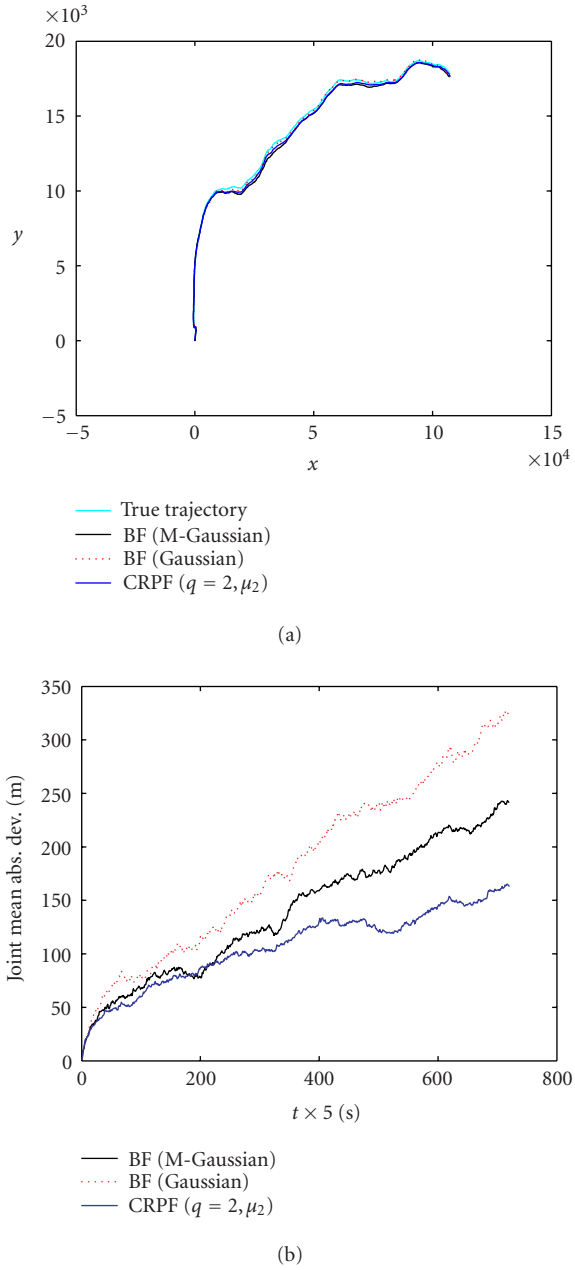

FIGURE 6: Mixture Gaussian system and observation noise—Gaussian BF. $T_s = 5$ seconds. (a) Trajectory. (b) Mean absolute deviation.

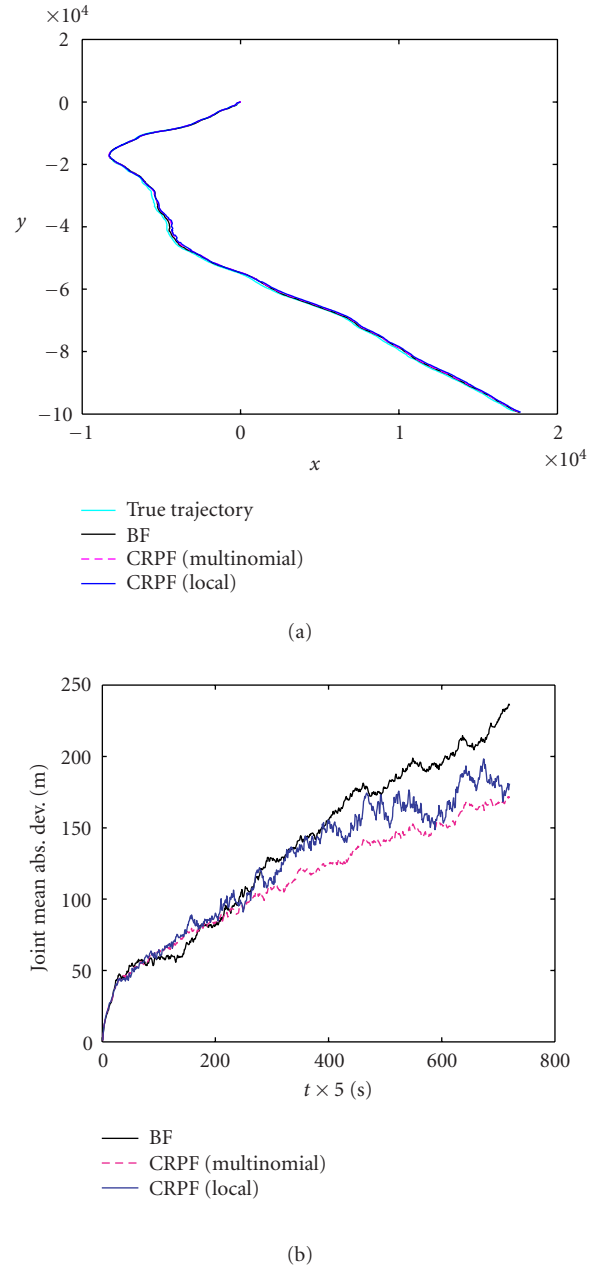

FIGURE 7: Local versus multinomial resampling. $T_s = 5$ seconds. (a) Trajectory. (b) Mean absolute deviation.

Since they do not assume explicit probabilistic models for the dynamic system, the proposed techniques, which have been termed CRPFs, are more robust than standard particle filters in problems where there is uncertainty (or a mismatch with physical phenomena) in the probabilistic model of the dynamic system. The basic concepts related to the formulation and design of these new algorithms, as well as theoretical results concerning their convergence, were provided. We also proposed a *local resampling* scheme that allows for simple implementations of the CRPF techniques with parallel VLSI hardware. Computer simulation results illustrate the robustness and the excellent performance of the proposed algorithms when compared to the popular auxiliary BF.

APPENDICES

A. CRPF AND STOCHASTIC APPROXIMATION

It is interesting to compare the CRPF method with the SA algorithm. The subject of SA can be traced back to the 1951 paper of Robbins and Monro [19], and a recent tutorial review can be found in [14].

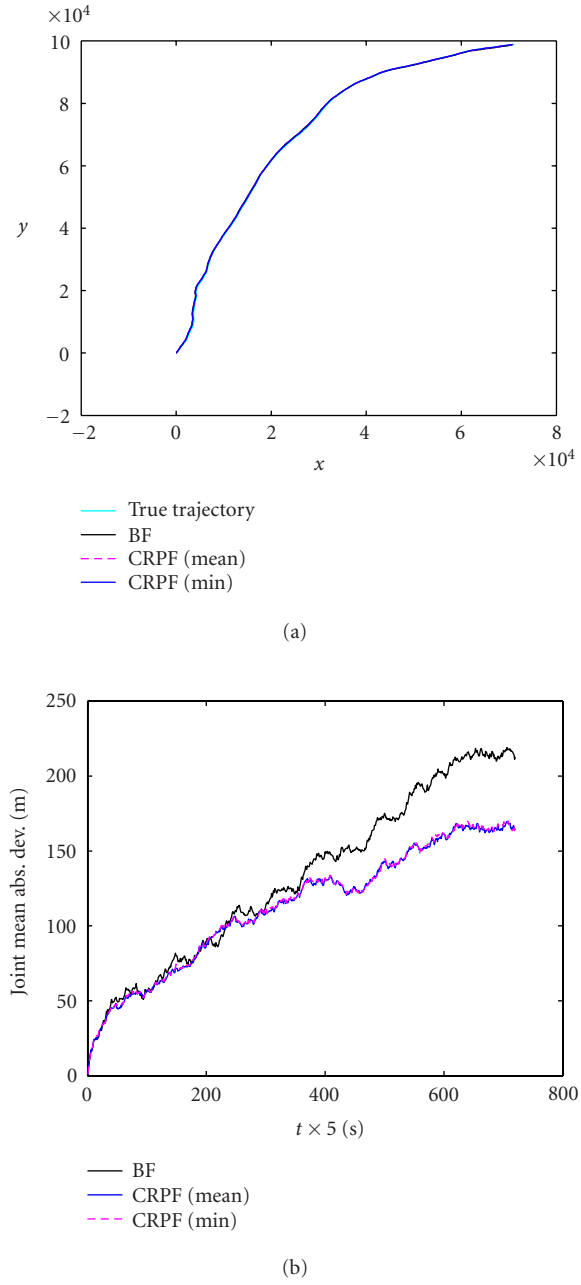

Figure 8: Different estimation criteria. $T_s = 5$ seconds. (a) Trajectory. (b) Mean absolute deviation.

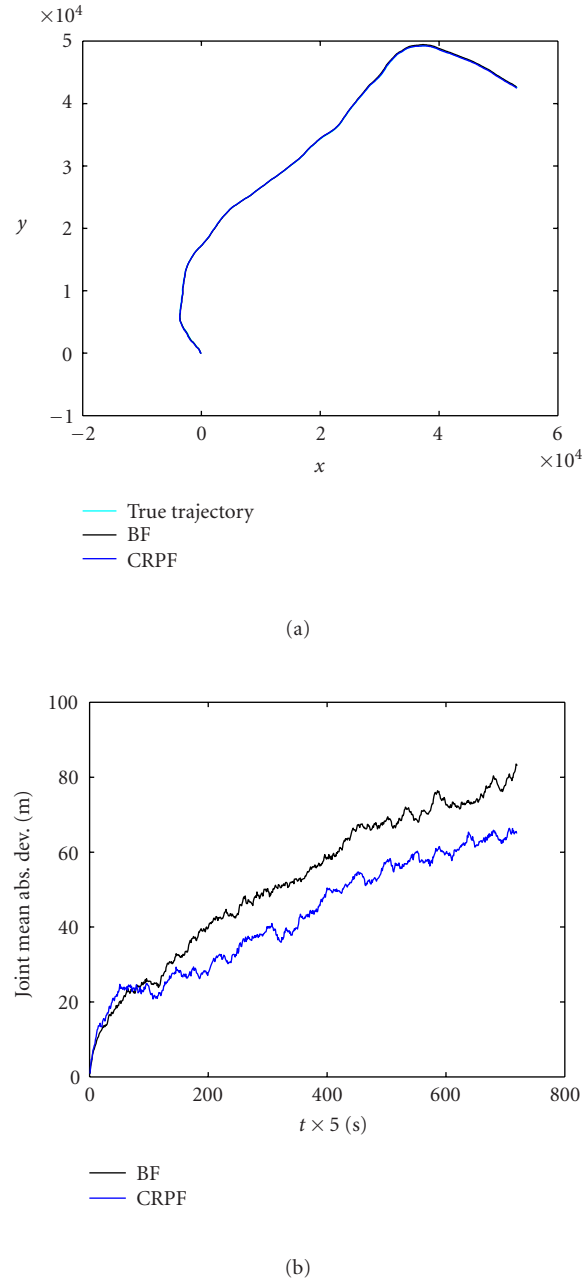

Figure 9: Laplacian. $T_s = 5$ s. (a) Trajectory. (b) Mean absolute deviation.

In a typical problem addressed by SA, an objective function that has to be minimized involves expectations, for example, the minimization of $E(Q(x, \psi_t))$, where $Q(\cdot)$ is a function of the unknown x and random variables ψ_t. The problem is that the distributions of the random variables are unknown and the expectation of the function cannot be analytically found. To make the problem tractable, one approximates the expectation by simply dropping the expectation operator, and proceeding as if $E(Q(x, \psi_t)) = Q(x, \psi_t)$. Robbins and Monro proposed the following scheme that solves for x_t:

$$\hat{x}_t = \hat{x}_{t-1} + \gamma_t Q(\hat{x}_{t-1}, \psi_t), \tag{A.1}$$

where γ_t is a sequence of positive scalars that have to satisfy the conditions $\sum_t \gamma_t = \infty$, $\sum_t \gamma_t^2 < \infty$. In the signal processing literature, the best known SA method is the LMS algorithm.

The CRPF method also attempts to estimate the unknown x_t without probabilistic assumptions. In doing so, it actually aims at *inverting* the dynamic model and, therefore,

it performs SA similarly to RM though by other means.[4] In CRPF, the dynamics of the state are taken into account both through the propagation step and by recursively solving the optimization problem (25). Further research in CRPF from the perspective of SA can probably yield new and deeper insight of this new class of algorithms.

B. PROOF OF LEMMA 1

The proof is carried out in two steps. First, we prove the implication

$$\frac{1-\delta}{\delta}\mathbf{S}_{\text{out}} \lim_{M\to\infty} \frac{E_{p_t^{M'}} n_M}{\mu_t(S^M(\mathbf{x}_t^{\text{opt}},\varepsilon))} = 0 \tag{B.1}$$

$$\Downarrow$$

$$\lim_{M\to\infty} \Pr\left[1 - \frac{\mu_t(S^M(\mathbf{x}_t^{\text{opt}},\varepsilon))}{\mu_t(\{\mathbf{x}_t^{(i)}\}_{i=1}^M)} \geq \delta\right] = 0 \tag{B.2}$$

for any $\varepsilon, \delta > 0$. Then, we only need to show that (B.1) holds true under conditions (38)–(40) in order to complete the proof.

Straightforward manipulation of the inequality in (43) leads to the following equivalence chain that holds true for any $\varepsilon, \delta > 0$:

$$\lim_{M\to\infty} \Pr\left[1 - \frac{\mu_t(S^M(\mathbf{x}_t^{\text{opt}},\varepsilon))}{\mu_t(\{\mathbf{x}_t^{(i)}\}_{i=1}^M)} \geq \delta\right] = 0 \tag{B.3}$$

$$\Updownarrow$$

$$\lim_{M\to\infty} \Pr\left[\mu_t(S^M(\mathbf{x}_t^{\text{opt}},\varepsilon)) \leq (1-\delta)\mu_t(\{\mathbf{x}_t^{(i)}\}_{i=1}^M)\right] = 0 \tag{B.4}$$

$$\Updownarrow$$

$$\lim_{M\to\infty} \Pr\left[\mu_t(S^M(\mathbf{x}_t^{\text{opt}},\varepsilon)) \leq \frac{1-\delta}{\delta}\mu_t(\{\mathbf{x}_t^{(i)}\}_{i=1}^M \setminus S^M(\mathbf{x}_t^{\text{opt}},\varepsilon))\right] = 0, \tag{B.5}$$

where we have exploited that

$$\mu_t(\{\mathbf{x}_t^{(i)}\}_{i=1}^M) = \mu_t(S^M(\mathbf{x}_t^{\text{opt}},\varepsilon)) + \mu_t\left(\{\mathbf{x}_t^{(i)}\}_{i=1}^M \setminus S^M(\mathbf{x}_t^{\text{opt}},\varepsilon)\right). \tag{B.6}$$

Using the notation

$$1_A(x) = \begin{cases} 1 & \text{if } x \in A, \\ 0 & \text{otherwise,} \end{cases} \tag{B.7}$$

for the indicator function, we can write

$$\begin{aligned} &\mu_t(\{\mathbf{x}_t^{(i)}\}_{i=1}^M \setminus S^M(\mathbf{x}_t^{\text{opt}},\varepsilon)) \\ &\quad= \sum_{i=1}^M \mu(\triangle\mathcal{C}(\mathbf{x}_t^{(i)}|\mathbf{y}_t)) 1_{\{\mathbf{x}:\|\mathbf{x}-\mathbf{x}_t^{\text{opt}}\|\geq\varepsilon\}}(\mathbf{x}_t^{(i)}) \\ &\quad\leq \mathbf{S}_{\text{out}} \sum_{i=1}^M 1_{\{\mathbf{x}:|\mathbf{x}_t^{(i)}-\mathbf{x}_t^{\text{opt}}|\geq\varepsilon\}}(\mathbf{x}_t^{(i)}) = \mathbf{S}_{\text{out}} n_M, \end{aligned} \tag{B.8}$$

where n_M is the cardinality of the discrete set $\{\mathbf{x}_t^{(i)}\}_{i=1}^M \setminus S^M(\mathbf{x}_t^{\text{opt}},\varepsilon)$. Therefore, using (B.8) and the equivalence between (B.3) and (B.5), we arrive at the implication

$$\begin{aligned} &\lim_{M\to\infty} \Pr\left[1 - \frac{\mu_t(S^M(\mathbf{x}_t^{\text{opt}},\varepsilon))}{\mu_t(\{\mathbf{x}_t^{(i)}\}_{i=1}^M)} \geq \delta\right] = 0 \\ &\qquad\Uparrow \\ &\lim_{M\to\infty} \Pr\left[\mu_t(S^M(\mathbf{x}_t^{\text{opt}},\varepsilon)) \leq \frac{1-\delta}{\delta}\mathbf{S}_{\text{out}} n_M\right] = 0. \end{aligned} \tag{B.9}$$

Since both $\mu(\triangle\mathcal{C}(\mathbf{x}_t|\mathbf{y}_t)) \geq 0$ and $(1-\delta)\mathbf{S}_{\text{out}} n_M/\delta > 0$, we can use the relationship [16, equation 4.4-5] to obtain

$$\begin{aligned} &\Pr\left[\mu_t(S^M(\mathbf{x}_t^{opt},\varepsilon)) \leq \frac{1-\delta}{\delta}\mathbf{S}_{\text{out}} n_M\right] \\ &\qquad\leq \frac{1-\delta}{\delta}\mathbf{S}_{\text{out}} \frac{E_{\Pr[n_M]}[n_M]}{\mu_t(S^M(\mathbf{x}_t^{opt},\varepsilon))}, \end{aligned} \tag{B.10}$$

where we have used the fact that the supremum $\mathbf{S}_{\text{out}}$ does not depend on M or n_M. When jointly considered, (B.9) and (B.10) yield the implication (B.1)⇒(B.2) and we only have to show that (B.1) holds true in order to complete the proof.

The expectation on the left-hand side of (B.1) can be computed by resorting to assumption (38), which yields, after straightforward manipulations,

$$\lim_{M\to\infty} E_{\Pr[n_M]}[n_M] = (1-\gamma) \lim_{M\to\infty} M \quad \text{(i.p.)}. \tag{B.11}$$

Substituting (B.11) into (B.1) yields

$$\begin{aligned} &\frac{1-\delta}{\delta}\mathbf{S}_{\text{out}} \lim_{M\to\infty} \frac{E_{\Pr[n_M]}[n_M]}{\mu_t(S^M(\mathbf{x}_t^{\text{opt}},\varepsilon))} \\ &\quad= \frac{1-\delta}{\delta}\mathbf{S}_{\text{out}}(1-\gamma) \lim_{M\to\infty} \frac{M}{\mu_t(S^M(\mathbf{x}_t^{\text{opt}},\varepsilon))} \\ &\quad\leq \frac{1-\delta}{\delta}\mathbf{S}_{\text{out}}(1-\gamma) \lim_{M\to\infty} \frac{M}{\mathbf{S}_{\text{in}}} = 0, \end{aligned} \tag{B.12}$$

where the last equality is obtained from assumptions (39) and (40). The proof is complete by going back to implication (B.1)⇒(B.2).

[4] Note, however, that the notion of inversion must be understood in a broad sense, since f_y may not necessarily be invertible and, even if f_y^{-1} exists, it may happen that $\mathbf{y}_t$ does not belong to its domain.

C. PROOF OF THEOREM 1

Using Lemma 1, we obtain that the set $S^M(\mathbf{x}_t^{\text{opt}}, \varepsilon)$ has (asymptotically) a unit probability mass after the propagation step, that is,

$$\lim_{M\to\infty} \sum_{i:\mathbf{x}_t^{(i)}\in S^M\left(\mathbf{x}_t^{\text{opt}},\varepsilon\right)} \omega_t^{(i)} = 1 \quad \text{(i.p.)}\ \forall \varepsilon > 0. \tag{C.1}$$

Therefore,

$$\begin{aligned}
&\lim_{M\to\infty} \left| \triangle\mathcal{C}\left(\mathbf{x}_t^{\text{opt}} \mid \mathbf{y}_t\right) - \overline{\triangle\mathcal{C}_t} \right| \\
&\quad = \lim_{M\to\infty} \left| \triangle\mathcal{C}\left(\mathbf{x}_t^{\text{opt}} \mid \mathbf{y}_t\right) \right. \\
&\qquad \left. - \sum_{i:\mathbf{x}_t^{(i)}\in S^M\left(\mathbf{x}_t^{\text{opt}},\varepsilon\right)} \omega_t^{(i)} \triangle\mathcal{C}\left(\mathbf{x}_t^{(i)} \mid \mathbf{y}_t\right) \right| \quad \text{(i.p.)},
\end{aligned} \tag{C.2}$$

$$\begin{aligned}
&\lim_{M\to\infty} \left| \triangle\mathcal{C}\left(\mathbf{x}_t^{\text{opt}} \mid \mathbf{y}_t\right) - \sum_{i:\mathbf{x}_t^{(i)}\in S^M\left(\mathbf{x}_t^{\text{opt}},\varepsilon\right)} \omega_t^{(i)} \triangle\mathcal{C}\left(\mathbf{x}_t^{(i)} \mid \mathbf{y}_t\right) \right| \\
&\quad \le \lim_{M\to\infty} \left| \triangle\mathcal{C}\left(\mathbf{x}_t^{\text{opt}} \mid \mathbf{y}_t\right) \right. \\
&\qquad \left. - \sup_{\mathbf{x}_t\in S^M\left(\mathbf{x}_t^{\text{opt}},\varepsilon\right)} \left\{\triangle\mathcal{C}\left(\mathbf{x}_t \mid \mathbf{y}_t\right)\right\} \right| \quad \text{(i.p.)}
\end{aligned} \tag{C.3}$$

for all $\varepsilon > 0$.

We write the upper bound on the right-hand side of (C.3) as a function of the radius ε:

$$B(\varepsilon) = \lim_{M\to\infty} \left| \triangle\mathcal{C}\left(\mathbf{x}_t^{\text{opt}} \mid \mathbf{y}_t\right) - \sup_{\mathbf{x}_t\in S^M\left(\mathbf{x}_t^{\text{opt}},\varepsilon\right)} \left\{\triangle\mathcal{C}\left(\mathbf{x}_t \mid \mathbf{y}_t\right)\right\} \right|. \tag{C.4}$$

It can be easily proved that $\lim_{M\to\infty} \#S^M(\mathbf{x}_t^{\text{opt}},\ \varepsilon = 1/\sqrt{M}) = \infty$, where # denotes the number of elements in a discrete set. Since $\lim_{M\to\infty} 1/\sqrt{M} = 0$ and it is assumed that $\triangle\mathcal{C}(\cdot\,|\mathbf{y}_t)$ is continuous and bounded in $S(\mathbf{x}_t^{\text{opt}}, \varepsilon)$ for all $\varepsilon > 0$, it follows that

$$B\left(\varepsilon = \frac{1}{\sqrt{M}}\right) = 0 \quad \text{(i.p.)} \tag{C.5}$$

and, by exploiting the fact that the left-hand side of (C.2) does not depend on ε, we can readily use (C.5) to obtain

$$\begin{aligned}
&\lim_{M\to\infty} \left| \triangle\mathcal{C}\left(\mathbf{x}_t^{\text{opt}} \mid \mathbf{y}_t\right) - \overline{\triangle\mathcal{C}_t} \right| \\
&\qquad \le B\left(\varepsilon = \frac{1}{\sqrt{M}}\right) = 0 \quad \text{(i.p.)},
\end{aligned} \tag{C.6}$$

which concludes the proof.

D. PROOF OF COROLLARY 1

When $\lambda = 0$, $\omega_t^{(i)} = \pi_t^{(i)}$ and, according to Lemma 1,

$$\lim_{M\to\infty} \sum_{i:\mathbf{x}_t^{(i)}\in S^M(\mathbf{x}_t^{\text{opt}},\varepsilon)} \pi_t^{(i)} = 1 \quad \text{(i.p.)} \tag{D.1}$$

for all $\varepsilon > 0$. Hence, we can write the mean state estimate (in the limit $M \to \infty$) as

$$\lim_{M\to\infty} \tilde{\mathbf{x}}_t^{\text{mean}} = \lim_{M\to\infty} \sum_{\mathbf{x}_t^{(i)}\in S^M(\mathbf{x}_t^{\text{opt}},\varepsilon)} \pi_t^{(i)} \mathbf{x}_t^{(i)} \tag{D.2}$$

and, therefore, the incremental cost of the mean state estimate can be upper bounded as

$$\lim_{M\to\infty} \triangle\mathcal{C}\left(\tilde{\mathbf{x}}_t^{\text{mean}} \mid \mathbf{y}_t\right) \le \lim_{M\to\infty} \sup_{\mathbf{x}_t\in S^M(\mathbf{x}_t^{\text{opt}},\varepsilon)} \triangle\mathcal{C}\left(\mathbf{x}_t|\mathbf{y}_t\right). \tag{D.3}$$

Using inequality (D.3) and the obvious fact that $\triangle\mathcal{C}(\mathbf{x}_t^{\text{opt}}|\mathbf{y}_t)$ is minimal by definition, we find that

$$\begin{aligned}
&\lim_{M\to\infty} \left| \triangle\mathcal{C}\left(\tilde{\mathbf{x}}_t^{\text{mean}}|\mathbf{y}_t\right) - \triangle\mathcal{C}\left(\mathbf{x}_t^{\text{opt}}|\mathbf{y}_t\right) \right| \\
&\quad \le \lim_{M\to\infty} \left| \sup_{\mathbf{x}_t\in S^M(\mathbf{x}_t^{\text{opt}},\varepsilon)} \triangle\mathcal{C}\left(\mathbf{x}_t|\mathbf{y}_t\right) - \triangle\mathcal{C}\left(\mathbf{x}_t^{\text{opt}}\mid\mathbf{y}_t\right) \right| \\
&\quad = B(\varepsilon),
\end{aligned} \tag{D.4}$$

where $B(\varepsilon)$ is the same as defined in (C.4). Therefore, we can apply the same technique as in the proof of Theorem 1 and, taking $\varepsilon = 1/\sqrt{M}$, we obtain

$$\begin{aligned}
&\lim_{M\to\infty} \left| \triangle\mathcal{C}\left(\tilde{\mathbf{x}}_t^{\text{mean}}\mid\mathbf{y}_t\right) - \triangle\mathcal{C}_t\left(\mathbf{x}_t^{\text{opt}}\mid\mathbf{y}_t\right) \right| \\
&\qquad \le B\left(\varepsilon = \frac{1}{\sqrt{M}}\right) = 0 \quad \text{(i.p.)},
\end{aligned} \tag{D.5}$$

which concludes the proof of the corollary.

ACKNOWLEDGMENTS

This work has been supported by Ministerio de Ciencia y Tecnología of Spain and Xunta de Galicia (Project TIC2001-0751-C04-01), and the National Science Foundation (Award CCR-0082607). The authors wish to thank the anonymous reviewers of this paper for their very constructive comments that assisted us in improving the original draft. J. Míguez would also like to thank Professor Ricardo Cao for intellectually stimulating discussions which certainly contributed to the final form of this work.

REFERENCES

[1] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME—Journal of Basic Engineering*, vol. 82, pp. 35–45, March 1960.

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

[3] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Trans. on Automatic Control*, vol. 45, no. 3, pp. 477–482, 2000.

[4] D. Crisan and A. Doucet, "A survey of convergence results on particle filtering methods for practitioners," *IEEE Trans. Signal Processing*, vol. 50, no. 3, pp. 736–746, 2002.

[5] W. J. Fitzgerald, "Markov chain Monte Carlo methods with applications to signal processing," *Signal Processing*, vol. 81, no. 1, pp. 3–18, 2001.

[6] J. S. Liu and R. Chen, "Sequential Monte Carlo methods for dynamic systems," *Journal of the American Statistical Association*, vol. 93, no. 443, pp. 1032–1044, 1998.

[7] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.

[8] N. Gordon, D. Salmond, and A. F. M. Smith, "Novel approach to nonlinear and non-Gaussian Bayesian state estimation," *IEE Proceedings Part F: Radar and Signal Processing*, vol. 140, no. 2, pp. 107–113, 1993.

[9] M. K. Pitt and N. Shephard, "Auxiliary variable based particle filters," in *Sequential Monte Carlo Methods in Practice*, A. Doucet, N. de Freitas, and N. Gordon, Eds., pp. 273–293, Springer, New York, NY, USA, 2001.

[10] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*, Springer, New York, NY, USA, 2001.

[11] M. L. Honig, U. Madhow, and S. Verdú, "Blind adaptive multiuser detection," *IEEE Transactions on Information Theory*, vol. 41, no. 4, pp. 944–960, 1995.

[12] J. R. Treichler, M. G. Larimore, and J. C. Harp, "Practical blind demodulators for high-order QAM signals," *Proceedings of the IEEE*, vol. 86, no. 10, pp. 1907–1926, 1998.

[13] J. Míguez and L. Castedo, "A linearly constrained constant modulus approach to blind adaptive multiuser interference suppression," *IEEE Communications Letters*, vol. 2, no. 8, pp. 217–219, 1998.

[14] T. L. Lai, "Stochastic approximation," *The Annals of Statistics*, vol. 31, no. 2, pp. 391–406, 2003.

[15] M. Bolić, P. M. Djurić, and S. Hong, "New resampling algorithms for particle filters," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03)*, vol. 2, pp. 589–592, Hong Kong, April 2003.

[16] H. Stark and J. W. Woods, *Probability and Random Processes with Applications to Signal Processing*, Prentice-Hall, Upper Saddle River, NJ, USA, 3rd edition, 2002.

[17] M. Bolić, P. M. Djurić, and S. Hong, "Resampling algorithms for particle filters suitable for parallel VLSI implementation," in *Proc. 37th Annual Conference on Information Sciences and Systems (CISS '03)*, Baltimore, Md, March 2003.

[18] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund, "Particle filters for positioning, navigation, and tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 425–437, 2002.

[19] H. Robbins and S. Monro, "A stochastic approximation method," *Annals of Mathematical Statistics*, vol. 22, pp. 400–407, 1951.

Joaquín Míguez was born in Ferrol, Galicia, Spain, in 1974. He obtained the Licenciado en Informática (M.S.) and Doctor en Informática (Ph.D.) degrees from Universidade da Coruña, Spain, in 1997 and 2000, respectively. Late in 2000, he joined Departamento de Electrónica e Sistemas, Universidade da Coruña, where he became an Associate Professor in July 2003. From April to December 2001, he was a Visiting Scholar in the Department of Electrical and Computer Engineering, Stony Brook University. His research interests are in the field of statistical signal processing, with emphasis on the topics of Bayesian analysis, sequential Monte Carlo methods, adaptive filtering, stochastic optimization, and their applications to multiuser communications, smart antenna systems, sensor networks, target tracking, and vehicle positioning and navigation.

Mónica F. Bugallo received the Ph.D. degree in computer engineering from Universidade da Coruña, Spain, in 2001. From 1998 to 2000, she was with the Departamento de Electrónica e Sistemas, the Universidade da Coruña, where she worked in interference cancellation applied to multiuser communication systems. In 2001, she joined the Department of Electrical and Computer Engineering, Stony Brook University, where she is currently an Assistant Professor, and teaches courses in digital communications and information theory. Her research interests lie in the area of statistical signal processing and its applications to different disciplines including communications and biology.

Petar M. Djurić received his B.S. and M.S. degrees in electrical engineering from the University of Belgrade in 1981 and 1986, respectively, and his Ph.D. degree in electrical engineering from the University of Rhode Island in 1990. From 1981 to 1986, he was a Research Associate with the Institute of Nuclear Sciences, Vinča, Belgrade. Since 1990, he has been with Stony Brook University, where he is a Professor in the Department of Electrical and Computer Engineering. He works in the area of statistical signal processing, and his primary interests are in the theory of modeling, detection, estimation, and time series analysis and its application to a wide variety of disciplines including wireless communications and biomedicine.