

BOX-PARTICLE INTENSITY FILTER

M. Schikora^{1,3}, A. Gning², L. Mihaylova², D. Cremers³, W. Koch¹, and R. Streit⁴

¹Fraunhofer FKIE, Dept. Sensor Data and Information Fusion, Wachtberg, Germany*

²University of Lancaster, School of Computing and Communications, United Kingdom

³Technical University of Munich, Department of Computer Science, , Germany

⁴Metron Inc., Reston VA, USA

Keywords: Multi-Target Tracking, Box Particle Filters, Poisson Point Processes, Intensity Filter, Interval Measurements

Abstract

This paper develops a novel approach for multi-target tracking, called box-particle intensity filter (box-iFilter). The approach is able to cope with unknown clutter, false alarms and estimates the unknown number of targets. Furthermore, it is capable of dealing with three sources of uncertainty: stochastic, set-theoretic and data association uncertainty. The box-iFilter reduces the number of particles significantly, which improves the runtime considerably. The low particle number enables this approach to be used for distributed computing. A box-particle is a random sample that occupies a small and controllable rectangular region of non-zero volume. Manipulation of boxes utilizes the methods from the field of interval analysis. Our studies suggest that the box-iFilter reaches an accuracy similar to a sequential Monte Carlo (SMC) iFilter but with much less computational costs.

1 Introduction

Multi-target tracking is a common problem with many applications. In most of these the expected target number is not known a priori, so that it has to be estimated from the measured data. In general, multi-target tracking involves the joint estimation of states and number of targets from a sequence of observations in the presence of detection uncertainty, association uncertainty and clutter [2]. Recently, the intensity filter (iFilter) [16, 17] has been presented, which is similar to the probability density hypothesis (PHD) filter [9]. Both filters give estimates to multi-target and multi-measurement states along with the estimation of the number of targets. While the PHD filter was originally derived using finite set statistics, the iFilter was derived through Poisson point processes (PPPs). An SMC implementation and analysis of the iFilter was published in [13].

The traditional measurement noise expresses uncertainty due to randomness, often referred to as statistical uncertainty. In many practical applications, however, the standard measurement model is not adequate. Complex distributed surveillance systems, for example, are often op-

erating under unknown synchronization biases and/or unknown system delays. The resulting measurements are affected by bounded errors of typically unknown distribution and biases, and can be expressed rather by intervals than by point values. An interval measurement expresses a type of uncertainty which is referred to as the set-theoretic uncertainty [3, 10], vagueness [8] or imprecision [15]. The concept of box-particle filtering in the context of tracking was introduced in [1]. In [5] it was shown that box particles can be seen as supports of uniform probability density functions (PDF), leading to Bayesian understanding of box particle filters. In [6] a single target box particle Bernoulli filter with box measurements is presented.

The main contribution of this work is a general derivation of box particle methods in the context of multi-target tracking with an unknown number of targets, clutter and false alarms. We present here a box particle version of the multi-target intensity filter (iFilter). In addition, a comparison of the box-iFilter with a standard sequential Monte Carlo iFilter is performed. The optimum subpattern assignment (OSPA) metric [14] is used for performance measure, together with the criteria for measuring the *inclusion* of the true state and the *volume* of the posterior PDF [6].

2 Poisson Point Processes (PPPs)

Every PPP is defined on a general set \mathcal{S} . In most of our applications this space will be considered as Euclidean, $\mathcal{S} \subseteq \mathbb{R}^d$, with $d \geq 1$ denoting the dimension, e.g. $d = 2$ for targets located in the (x, y) -plane. Other more complicated spaces are also possible. Realizations of PPPs on $\mathcal{R} \subset \mathcal{S}$ consist of $n > 0$ points $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n \in \mathcal{R}$. We denote a realization as ordered pair $\xi = (n, \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\})$. If n is equal to zero we set $\xi = (0, \emptyset)$, with \emptyset the empty set. Through this notation we emphasize that the ordering of $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ is irrelevant, but not that the points are necessarily distinct. The event space of a PPP is defined as

$$\mathcal{E}(\mathcal{S}) := \{(0, \emptyset)\} \cup \{(n, \{\mathbf{x}_1, \dots, \mathbf{x}_n\}) : \mathbf{x}_i \in \mathcal{S}, i = 1, \dots, n\}_{n=1}^{\infty}. \quad (1)$$

Every PPP is parameterized by only one function $g : \mathcal{S} \rightarrow \mathbb{R}, s \mapsto g(s), s \in \mathcal{S}$, called the intensity. We call $g(s)$ the intensity at point s . For all $s \in \mathcal{S}$, if $g(s) = c, c \geq 0$, where c is constant, the PPP is called homogeneous; otherwise it is non-homogeneous. It is assumed that $0 \leq \int_{\mathcal{R}} g(s) ds < \infty$ holds for all bounded subsets \mathcal{R} of \mathcal{S} , $\mathcal{R} \subset \mathcal{S}$. One realization of the PPP with intensity $g(s)$ comprises the number

Corresponding author: M. Schikora, Email: marek.schikora@fkie.fraunhofer.de

and the locations of points in \mathcal{R} .

In multi-target tracking applications two sequences of PPPs are usually used: one which corresponds to the multi-target state $\mathcal{X}_0, \mathcal{X}_1, \dots, \mathcal{X}_k$ and one that corresponds to measurements $\mathcal{Z}_1, \mathcal{Z}_2, \dots, \mathcal{Z}_k$. Both are bound to discrete time steps t_0, t_1, \dots, t_k , with $t_{j-1} < t_j$ for $j = 1, \dots, k$. Measurements are assumed to be only available for time steps $j > 0$. An important but subtle point is hidden in this language. The multi-target process is not assumed to be a PPP, but it is approximated at every time step by a PPP. These PPP approximations are the \mathcal{X}_k . Similarly, measurement sets are not assumed to be PPPs. However, under the approximate PPP target models, the measurements are realizations of PPPs. These are the \mathcal{Z}_k .

We define now $\mathcal{S} \subseteq \mathbb{R}^{n_x}$, with $n_x \geq 1$, an n_x dimensional bounded single target state space. The multi-target state space is then an augmented space $\mathcal{S}^+ = \mathcal{S} \cup \mathcal{S}_\phi$, where \mathcal{S}_ϕ represents space of the ‘‘target absent’’ hypothesis ϕ . \mathcal{S}^+ is a discrete-continuous space. The main concepts of PPPs can be adapted to this space, but some modifications are needed. Here, $g(s)$ is a intensity defined for all $s \in \mathcal{S}^+$. Integrals of $g(s)$ over bounded subsets \mathcal{R} of \mathcal{S}^+ must be finite, giving a discrete–continuous integral:

$$0 \leq \int_{\mathcal{R}^+} g(s) ds \equiv g(\phi) + \int_{\mathcal{R}} g(\mathbf{x}) d\mathbf{x} < \infty, \quad (2)$$

with $\mathcal{R}^+ \subset \mathcal{S}^+$, $\mathcal{R} \subset \mathcal{S}$, $\phi \in \mathcal{S}_\phi$ and $g(\phi)$ being a dimensionless intensity on \mathcal{S}_ϕ . The number of copies of ϕ , or ‘‘clutter targets’’, in a realization is Poisson distributed with mean $g(\phi)$. In other words, $g(\phi)$ is the expected number of targets in \mathcal{S}_ϕ . The augmented state space enables estimates of both target birth and measurement clutter. The integral $\int_{\mathcal{R}} g(\mathbf{x}) d\mathbf{x}$ is the expected number of targets in \mathcal{S} . The measurement sequence is defined on the measurement space $\mathcal{Z} \subset \mathbb{R}^{n_z}$, with $n_z \geq 1$ being the dimension of the individual measurement. Further discussion of PPPs defined on discrete-continuous and other spaces is given in [16].

3 General Overview – iFilter

The iFilter operates on the augmented space \mathcal{S}^+ . In the same way as a standard single target filter consists of two main steps (Prediction and Update) also the iFilter predicts the intensity over \mathcal{S}^+ and then updates this intensity every time when new measurements arrive.

In the following sections an index $a|b$ in the intensity function $f_{a|b}(\cdot)$ denotes that the intensity was updated in time step t_a with all the measurements up to time step t_b .

The intensity of the PPP \mathcal{X}_k is $f_{k|k}(s)$, $s \in \mathcal{S}^+$. We split the intensity $f_{k|k}(s)$ over \mathcal{S}^+ into two intensities $f_{k|k}(\mathbf{x})$ and $f_{k|k}(\phi)$. In general we can write

$$f_{k|k}(s) = \begin{cases} f_{k|k}(\mathbf{x}), & s = \mathbf{x} \in \mathcal{S} \\ f_{k|k}(\phi), & s = \phi \in \mathcal{S}_\phi \end{cases} \quad (3)$$

with $f_{k|k}(\mathbf{x})$ being the intensity over \mathcal{S} and $f_{k|k}(\phi)$ the intensity for \mathcal{S}_ϕ . In order to describe the iFilter the following

probabilities and PDFs have to be defined:

$$\psi_k(\mathbf{x} | \phi) \quad \text{transition probability for new targets} \quad (4)$$

$$\psi_k(\mathbf{x} | \mathbf{y}) \quad \text{target transition probability} \quad (5)$$

$$\psi_k(\phi | \phi) \quad \text{transition probability in } \mathcal{S}_\phi \quad (6)$$

$$\psi_k(\phi | \mathbf{x}) \quad \text{transition probability for target death} \quad (7)$$

$$p_k(\mathbf{z} | \mathbf{x}) \quad \text{measurement likelihood} \quad (8)$$

$$p_k(\mathbf{z} | \phi) \quad \text{likelihood for measurement from } \phi \quad (9)$$

$$p_k^D(\mathbf{x}) \quad \text{detection probability for } \mathbf{x} \quad (10)$$

$$p_k^D(\phi) \quad \text{detection probability for } \phi \quad (11)$$

with $\mathbf{x}, \mathbf{y} \in \mathcal{S}$ and $\mathbf{z} \in \mathcal{Z}$. Let us assume that we have the intensities $f_{k-1|k-1}(\mathbf{x})$ and $f_{k-1|k-1}(\phi)$, from the previous time step t_{k-1} . Similarly to most stochastic filtering techniques, the iFilter admits the Markovian assumption that the current state is only dependent of the last state. In every time step t_k the likelihood (9) is set, according to:

$$p_k(\mathbf{z} | \phi) = \frac{(f_{k-1|k-1}(\phi))^{m_k}}{m_k!} e^{(-f_{k-1|k-1}(\phi))}, \quad (12)$$

with m_k the number of measurements in time step t_k .

The main drawback of the iFilter equations is that in general the involved integrals cannot be solved analytically. Therefore an appropriate numerical solution is needed. Previously, we showed a SMC-iFilter in which the intensity $f_{k|k}(\mathbf{x})$ was be approximated by particles (delta peaks) drawn from this intensity [13]. Actually the particles approximate the involved integrals and not the intensities. Another name for this kind of technique is particle based filtering [11]. In this work, however, we present a representation with box particles, which can be seen as supports of uniform probability density functions (PDF).

4 Interval Analysis

This section gives a short introduction to the field of interval analysis, which will be used in this article. For more informations see [7]. The original idea of interval analysis was to deal with intervals instead of real numbers for exact computation in the presence of rounding errors. However, this field has strongly increased its potential applications. We will use the main concepts to represent particles not as delta peaks but as boxes in the state space. An interval $[x] = [\underline{x}, \bar{x}] \in \mathbb{IR}$ is a closed and connected subset of the real numbers \mathbb{R} , with $\underline{x} \in \mathbb{R}$ representing its lower bound and $\bar{x} \in \mathbb{R}$ its upper bound. In multiple dimensions d this interval becomes a box $[\mathbf{x}] \in \mathbb{IR}^d$ defined as a Cartesian product of d intervals: $[\mathbf{x}] = [x_1] \times \dots \times [x_d]$. Here the operator $[\cdot]$ will be used as the size of a box $[\mathbf{x}]$. The function $\text{mid}([\mathbf{x}])$ returns the center of a box. Elementary arithmetic operations, basic functions and operations between sets have been naturally extended to the interval analysis context.

For general functions the concept of *inclusion functions* has been developed. An *inclusion function* of a given function g is defined such that the image of a box $[\mathbf{x}]$ is a box $[g]([\mathbf{x}])$ containing $g([\mathbf{x}])$. Of course, the goal is to use only inclusion functions, which are minimal in the sense that the

size of the box $[g](\mathbf{x})$ is minimal but still covers the whole image of a box $[\mathbf{x}]$. An important class in the context of tracking are the *natural inclusion functions*.

Theorem 1. Assume $g: \mathbb{R}^d \rightarrow \mathbb{R}$, $(x_1, \dots, x_d) \mapsto g(x_1, \dots, x_d)$ is a function expressed as a finite composition of the operators $+$, $-$, $*$, $/$ and elementary functions (\sin , \cos , \exp , ...). A **natural inclusion function** is obtained by replacing each real variable and each operator or function by its interval counterpart.

In general natural inclusion functions are not minimal, but many functions can be modified in order to satisfy the conditions in the following theorem and then their natural inclusion functions are minimal. Proofs can be found in [7].

Definition 1. An inclusion function $[g]$ for g is convergent if, for any sequence of boxes $[\mathbf{x}](k)$,

$$\lim_{k \rightarrow \infty} |[\mathbf{x}](k)| = 0 \Rightarrow \lim_{k \rightarrow \infty} [g](\mathbf{x}(k)) = 0. \quad (13)$$

Theorem 2. If g involves only continuous operators and continuous elementary functions then $[g]$ is convergent. If, furthermore, each of the variables x_1, \dots, x_d occurs at most once in the formal expression of g , then $[g]$ is minimal.

The next needed concept is *contraction*, which will be used in the definition of likelihood functions and the update step of the proposed filters. A Constraint Satisfaction Problem (CSP), often denoted by \mathcal{H} , can be written as:

$$\mathcal{H} = (\mathbf{g}(\mathbf{x}) = \mathbf{0}, \mathbf{x} \in [\mathbf{x}]). \quad (14)$$

A common interpretation of (14) is: find the optimal box enclosure of the set of vector \mathbf{x} belonging to a given prior domain $[\mathbf{x}]$ satisfying a set of m constraints $\mathbf{g} = (g_1, \dots, g_m)^T$, with g_i a real valued function. The solution consists of all \mathbf{x} , that satisfy $\mathbf{g}(\mathbf{x}) = \mathbf{0}$ or written as a set:

$$\mathbb{S} = \{\mathbf{x} \in [\mathbf{x}] \mid \mathbf{g}(\mathbf{x}) = \mathbf{0}\}. \quad (15)$$

A *contraction* of \mathcal{H} means replacing $[\mathbf{x}]$ by a smaller box $[\mathbf{x}]'$ under the constraint $\mathbb{S} \subseteq [\mathbf{x}]' \subseteq [\mathbf{x}]$. There are several methods to build a contractor for \mathcal{H} , e.g. by the Gauss elimination, Gauss-Seidel algorithm, linear programming. In this work, however, we will use *Constraint Propagation* (CP) for its good suitability in the context of tracking problems.

5 From Particles to Boxes

A popular class of methods for implementation of Bayes-like filters are particle filters [11]. Applying these methods to the iFilter leads to a particle approximation of the intensity $f_{k|k}(\mathbf{x})$ with a set of N_k weighted random samples $\{(\mathbf{x}_i, w_i)\}_{i=1}^{N_k}$. The approximation can be written as:

$$f_{k|k}(\mathbf{x}) \approx \sum_{i=1}^{N_k} w_i \delta_{\mathbf{x}_i}(\mathbf{x}), \quad (16)$$

where $\delta_{\mathbf{x}_i}(\mathbf{x})$ is the Dirac delta function concentrated at \mathbf{x}_i . The sum (16) converges to $f_{k|k}(\mathbf{x})$, with $N_k \rightarrow \infty$ [4]. The number of particles used is a key issue to the overall

filter performance. In general, the higher the number of particles, the better the approximation and with it the performance. However, a high number leads often to a computational demanding scenario. In [1] the authors presented a natural way to deal with the decrease of N_k by using boxes instead of point particles and combining particle filter techniques with interval analysis methods. Moreover, in [5] the authors propose to interpret box particles as supports of uniform PDFs, so that (16) changes to:

$$f_{k|k}(\mathbf{x}) \approx \sum_{i=1}^{N_k} w_i U_{[\mathbf{x}_i]}(\mathbf{x}), \quad (17)$$

with $U_{[\mathbf{x}_i]}(\mathbf{x})$ denoting the uniform PDF over the box $[\mathbf{x}_i]$. Bayes-like filters (also the SMC-iFilter) require the knowledge of the measurement likelihood function $p(\mathbf{z}|\mathbf{x})$. A likelihood returns values in the interval $[0, 1]$. The returned value depends on the probability that this measurement \mathbf{z} was produced by the state \mathbf{x} . In the context of this article we assume also box measurements $[\mathbf{z}]$. We do not need to model the statistical sensor error with some error density (that in practice is mostly unknown) and we do not need to model systematic errors directly. With this assumption the only information needed from a sensor is its error range. We define the measurement likelihood for box measurements and box particles as

$$p([\mathbf{z}] \mid [\mathbf{x}]) := \frac{|[h_{\text{CP}}](\mathbf{x}, [\mathbf{z}])|}{|[\mathbf{x}]|}. \quad (18)$$

The function $[h_{\text{CP}}](\mathbf{x}, [\mathbf{z}])$ returns a contracted version of $[\mathbf{x}]$ under the constraints given by the measurement function $\mathbf{h}(\mathbf{x}) = \mathbf{z}$. An example is given in [6].

6 The Box-iFilter

Inspired by the works of Vo et al. [18] and Ristic et al. [12] on efficient sequential Monte Carlo methods for the PHD filter we present here a sequential Monte Carlo method for the iFilter with box particles. A general implementation of the iFilter with punctual particles was previously published in [13]. Here we use an improved version of this work as basis to build a box particle implementation.

The box-SMC-iFilter can be summarized in eight steps, which will be presented in the following. Here the box particle set represents the target intensity of the PPP, which corresponds to the multi-target state \mathcal{X}_k . By analogy to the PHD-filter, the integral over this intensity (or sum, if using particles) is the estimated expected number of targets and it is not necessary equal to one. Given from the previous time step we have the particle set $\{([\mathbf{x}_i], w_i)\}_{i=1}^{N_k}$, with $[\mathbf{x}_i] \in \mathbb{I}\mathbb{R}^{n_x}$, w_i the corresponding weight and N_k denoting the number of particles, estimated at time step t_{k-1} . This set represents the target intensity. In addition we have the intensity of the space \mathcal{S}_ϕ denoted by $f_{k-1|k-1}(\phi)$. For the sake of simplicity, we will assume in the following uniformly distributed clutter. With this assumption the intensity $f_{k-1|k-1}(\phi)$ can be represented by a single number, called the number of ϕ hypotheses.

The implementation details using a box particle representation are presented in the following. Steps 1 and 2 correspond to the prediction phase, steps 3-7 to the correction phase and step 8 to the resampling phase of a sequential Monte Carlo algorithm.

- 1. Predict target intensity:** The resampled particle set gained from the previous step is denoted by $\{[\mathbf{x}_i], w_i\}_{i=1}^{N_k}$, where N_k was estimated in time step t_{k-1} , c.f. Step 8. The prediction of the whole set can be modeled by applying a transition model to every particle and adding some noise to it. The weights are unchanged. In practical implementations this has the same effect as predicting the intensity distribution over S with a closed formula.

The iFilter models the birth process by itself, so that the particle number has to be increased in order to represent newly born targets correctly. Then

$$N_{k,new} = \left\lceil \frac{N_k}{\eta_{k-1}} \cdot (1 - \psi_k(\phi | \phi)) \cdot f_{k-1|k-1}(\phi) \right\rceil \quad (19)$$

denotes the additional number of particles. In order to avoid a high number of additional particles, we will sample new born particles according to the measurements from the previous time step Z_{k-1} . Let m_{k-1} denote the number of measurements in time step t_{k-1} , then for each of these we sample

$$N_{k,new}^j = \lceil N_{k,new} / m_{k-1} \rceil, \quad j = 1, \dots, m_{k-1} \quad (20)$$

many particles $[\tilde{\mathbf{x}}_i]$ drawn from the distribution $\mathcal{N}(\text{mid}([z_j^{k-1}]), \Sigma)$, centered around an old measurement $[z_j^{k-1}]$ with a covariance matrix Σ . The weights of the new born particles are set to

$$w_i = \frac{\psi_k(\mathbf{x}_i | \phi) \cdot f_{k-1|k-1}(\phi)}{N_{k,new}}, \quad i = 1, \dots, N_{k,new}. \quad (21)$$

This sampling is an approximation of the transition model $\psi_k([\mathbf{x}] | \phi)$, which has proven to be very stable in experiments. We define $\{[\tilde{\mathbf{x}}_i], w_i\}_{i=1}^{N_k + N_{k,new}}$ as the predicted particle set containing the newly created and the shifted particles.

- 2. Predict hypothesis intensity** The predicted number of ϕ hypotheses is then the sum of a predicted number of persistently absent and newly absent targets:

$$f_{k|k-1}(\phi) = \psi_k(\phi | \phi) \cdot f_{k-1|k-1}(\phi) + \sum_{i=1}^{N_k} \psi_k(\phi | [\tilde{\mathbf{x}}_i]) \cdot w_i. \quad (22)$$

- 3. Predict measurement intensity**

For all new measurements $[z_j]$, with $j = 1, \dots, m_k$ compute, the partition functions evaluated at $[z_j]$ for the state space and ϕ :

$$\nu_k([z_j]) = \sum_{i=1}^{N_k + N_{k,new}} p_k([z_j] | [\tilde{\mathbf{x}}_i]) p_k^D([\tilde{\mathbf{x}}_i]) w_i \quad (23)$$

$$\tilde{\lambda}_k([z_j]) = p_k([z_j] | \phi) p_k^D(\phi) f_{k|k-1}(\phi). \quad (24)$$

The sum of both is the measurement intensity for $[z_j]$

$$\lambda_{k|k-1}([z_j]) = \tilde{\lambda}_k([z_j]) + \nu_k([z_j]) \quad (25)$$

- 4. Estimate target states** To avoid a clustering step the methodology presented in [12] is used and adopted to the iFilter. First, compute the following weights for all new measurements $[z_j], j = 1, \dots, m_k$ and all persistent particles, i.e. not the new born, $[\mathbf{x}_i], i = 1, \dots, N_k$.

$$w_{j,i} = \frac{p_k([z_j] | [\tilde{\mathbf{x}}_i]) p_k^D([\tilde{\mathbf{x}}_i])}{\lambda_{k|k-1}([z_j])} \cdot w_i \quad (26)$$

Then compute the following sum $W_j = \sum_{i=1}^{N_k} w_{j,i}$, which can be seen as a probability of existence for target j , similarly to the multi-target multi-Bernoulli filter. For further analysis only those j are considered for which W_j is above a specified threshold τ , i.e.

$$\mathcal{J} = \{j | W_j > \tau, j = 1, \dots, m_k\} \quad (27)$$

For all $j \in \mathcal{J}$ the estimated point states are then:

$$\hat{\mathbf{y}}_j = \frac{1}{W_j} \sum_{i=1}^{N_k} \text{mid}([\tilde{\mathbf{x}}_i]) \cdot w_{j,i}. \quad (28)$$

For all $j \in \mathcal{J}$ the estimated box states are then:

$$[\hat{\mathbf{y}}_j] = \frac{1}{W_j} \sum_{i=1}^{N_k} [\tilde{\mathbf{x}}_i] \cdot w_{j,i}. \quad (29)$$

In Equation (28) and (29) we added, in contrast to [12], the normalization term $\frac{1}{W_j}$ to receive more accurate state estimates when W_j is not practically one. Note that only targets that have been detected at time step t_k can be reported as present. In experiments τ is usually set as $\tau = 0.75$.

- 5. Update target intensity** Given m_k new measurements the update of the state intensity is realized through a correction of the individual particle weights. For every particle $([\mathbf{x}_i], w_i)$, with $i = 1, \dots, N_k + N_{k,new}$ set:

$$\hat{w}_i = \left[(1 - p_k^D([\tilde{\mathbf{x}}_i])) + \sum_{j=1}^{m_k} \frac{p_k([z_j] | [\tilde{\mathbf{x}}_i]) p_k^D([\tilde{\mathbf{x}}_i])}{\lambda_{k|k-1}([z_j])} \right] \cdot w_i \quad (30)$$

- 6. Update hypothesis intensity** Adjust also the number of ϕ hypotheses:

$$f_{k|k}(\phi) = \left[(1 - p_k^D(\phi)) + \sum_{j=1}^{m_k} \frac{p_k([z_j] | \phi) p_k^D(\phi)}{\lambda_{k|k-1}([z_j])} \right] \cdot f_{k|k-1}(\phi) \quad (31)$$

- 7. Contract particles** In order to improve the accuracy of the filter we contract every box particle $\mathbf{x}_i, i = 1, \dots, N_k + N_{k,new}$ with its corresponding measurement. The corresponding measurement is defined through:

$$[\mathbf{z}] = \arg \max_{w_{j,i}} \{ [z_j], w_{j,i} > 0 \}. \quad (32)$$

If no $[\mathbf{z}]$ is found, this particle is not contracted, else the particle i is set to

$$([\hat{\mathbf{x}}_i], \hat{w}_i), \text{ with } [\hat{\mathbf{x}}_i] = [\mathbf{h}_{\text{CP}}]([\tilde{\mathbf{x}}_i], [\mathbf{z}]). \quad (33)$$

8. **Resampling** The number of particles in the state space may and should vary over time in order to represent the current situation better, e.g. more targets need more particles, so that the particle approximation accuracy is still sufficient. To estimate the correct number of particles resampled for the next time step compute first the estimated expected number of targets

$$\eta_k = \sum_{i=1}^{N_k + N_{k,\text{new}}} \hat{w}_i. \quad (34)$$

Then compute the probability: $p_S = \eta_k / (\eta_k + f_{k|k}(\phi))$. The number of resampled particles N_{k+1} is then the expectation of a binomial distribution with the probability p_S and samples equal to $N_k + N_{k,\text{new}}$, i.e.

$$N_{k+1} = (N_k + N_{k,\text{new}}) \cdot p_S. \quad (35)$$

The estimation of N_{k+1} at every time step prevails the particle number from growing against infinity. Given N_{k+1} any standard resampling technique for particle filtering can be used. However, instead of duplicating box particles (sampled more than once in the resampling step), subdivide a box particle into a disjunct set of new smaller boxes. The number of division is the number of times a box particle has been chosen for resampling. Several strategies of subdivision can be used. In this paper we randomly pick a dimension of the selected box particle and divide equally, obtaining a set of new box particles. Rescale the weights by η_k to get a new particle set $\{\mathbf{x}_i, \eta_k / N_{k+1}\}_{i=1}^{N_{k+1}}$.

7 Numerical Studies

This section gives numerical studies for the proposed box-iFilter algorithm. For comparison with traditional particle filter techniques we use a point particle SMC-iFilter. As performance measure the optimum subpattern assignment (OSPA) metric [14] is used for performance measure, together with the criteria for measuring the *inclusion* of the true state and the *volume* of the posterior PDF. The later two were introduced in [6].

We analyze the behavior of both filters in a demanding linear scenario. Herein six inertial moved targets are placed in an area $A = [-500, 500] \times [-500, 500]$. The state space is $\mathcal{S} \subset \mathbb{R}^4$, where the first two components correspond to the x and y coordinates and the third and fourth their velocities. The measurement space consists of $[x]$ and $[y]$ measurements, so $\mathcal{Z} \subset \mathbb{R}^2$. The measurement noise is white gaussian noise with a standard deviation $\sigma_x = \sigma_y = 15$. The probability of detection is set equal for all states to $p_k^D([\mathbf{x}]) = 0.95$. Targets 1 – 3 are present for all time steps. Target 4 is presented between time step 15 and 90. Target 5 and 6 are present between time step 30 and 75. The whole scenario has a length of 100 time steps. The number

of clutter measurements is estimated following a Poisson distribution with the mean value $|A| \cdot \rho_A$:

$$p(n_c) = \frac{1}{n_c!} (A \cdot \rho_A)^{n_c} \exp(-|A| \cdot \rho_A), \quad (36)$$

with $|A|$ denoting the volume of a observed area and ρ_A a parameter describing the clutter rate. For this scenario we used $\rho_A = 4 \cdot 10^{-6}$. Clutter measurements are generated by a i.i.d. process. The transition probabilities where set equal for both filter to $\psi_k(\mathbf{x} | \phi) = 0.2$, $\psi_k(\phi | \phi) = 0.01$ and $\psi_k(\phi | \mathbf{x}) = 0.1$ and the probabilities of detection where set to $p_k^D(\mathbf{x}) = 0.95$ and $p_k^D(\phi) = 0.3$. To initialize the particle cloud at time step $t_k = 0$, $N_0 \in \mathbb{N}^+$ particles are distributed uniformly across the state space \mathcal{S} , e.g. $N_0 = 1000$. The weights are set to $w_i = 1/N_0$ and $f_{0|0}(\phi)$ is set to a initial number, e.g. 2. We assume a constant velocity model for the prediction of persistent particles. In addition we add a 3σ interval of some white process noise, defined by a covariance matrix Σ . Hidden here are inclusion functions for the individual dimension of the state space. A close look reveals that every variable only appears once (for each dimension) and that all operations are continuous, so these natural inclusion functions are minimal and the propagated boxes have minimal size. This fact holds for constant velocity models with arbitrary dimensions.

In this experiment we investigate the accuracy achieved with the box-iFilter in comparison with the SMC-iFilter. To do so we will use the linear scenario described earlier. Figure 1 visualizes the mean OSPA values achieved with both filters on the given scenario. We can observe that the OSPA values are in general very low. This means that the SMC-iFilter and the box-iFilter behave very good in this scenario. However, we can also observe that the box-iFilter has slightly higher OSPA values than the SMC-iFilter. The authors of [6] already noticed that point estimates gained from box particles can have a slight bias. Therefore they introduced two new measurements criteria *inclusion* and *volume*. The mean results for 1000 Monte Carlo trials and all targets are shown in Figures 2 and 3, respectively. It can be easily seen that the inclusion and volume values react to target appearance and target disappearance. In general we can say that the box-iFilter has a higher volume than the SMC-iFilter. This can be seen as a drawback of the box particle technique. However, a closer look on the inclusion values reveals that the higher volume leads to better values for the inclusion criteria. So, we can state that the SMC-iFilter converges too fast and therefore it can happen sometimes that the true target state is not in the support of any covariance matrix \mathbf{P}_j . From an engineering point of view both filters reach similar results in this scenario. The estimated mean number of states is are practically identical for both filters. Nevertheless, the number of particles need for the box-iFilter is much smaller in comparison with the SMC-iFilter, which yields in a better runtime. The SMC-iFilter needs a mean runtime for processing one time step 7.46 msec. The box-iFilter needs only 1.47 msec. The mean speedup factor for the box-iFilter is 5.2. The mean number of particles in this scenario was 928.2 for the SMC-iFilter and 83.7 for the box-iFilter.

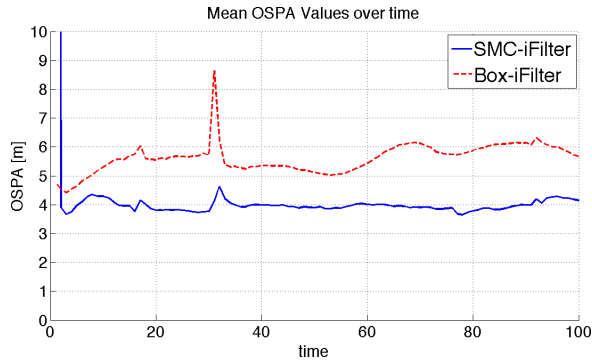


Figure 1: Mean OSPA values for 1000 Monte Carlo trials on linear scenario for both filters.

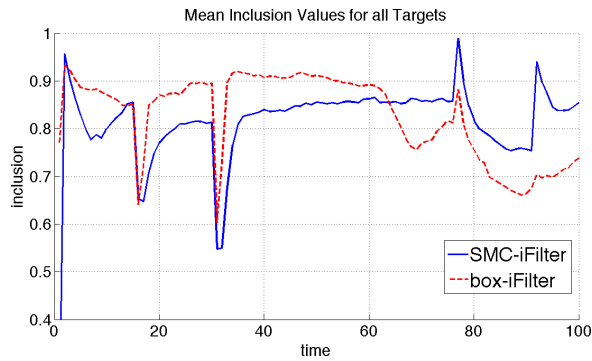


Figure 2: Mean inclusion values for 1000 Monte Carlo trials and all targets on linear scenario for both filters.

8 Conclusions

In this paper we presented a novel technique for non-linear multi-target tracking with a box particle based filter, called the box-iFilter. The theoretical backbone of this are Poisson point processes, which can be used to derive the general intensity filter equations. For the implementation, however, methods from interval analysis are used additionally to get a box particle representation of the iFilter. This representation allows a decrement of the number of particles needed. In our experiments we could reduce the number of particles by a factor of approximately eleven and reduce the compu-

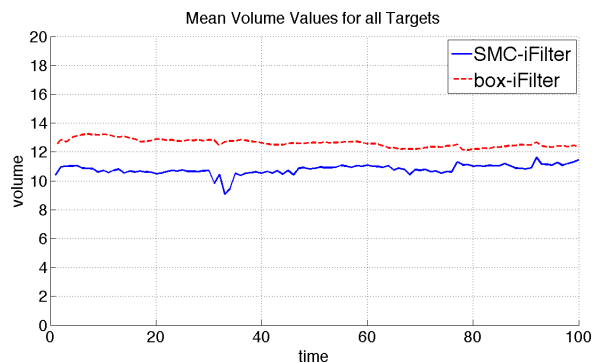


Figure 3: Mean volume values for 1000 Monte Carlo trials and all targets on linear scenario for both filters.

tation time by a factor of approximately five. On the other hand, the accuracy of the filter was preserved in comparison to a point particle implementation.

Acknowledgment

We acknowledge the support of the European Community's Seventh Framework Program [FP7/2007-2013] under grant agreement No 238710 (Monte Carlo based Innovative Management and Processing for an Unrivalled Leap in Sensor Exploitation).

References

- [1] F. Abdallah, A. Gning, and P. Bonnifait. Box particle filtering for nonlinear state estimation using interval analysis. *Automatica*, 44:807–815, 2008.
- [2] Y. Bar-Shalom and T.E. Fortmann. *Tracking and Data Association*. San Diego, CA: Academic, 1988.
- [3] P.L. Combettes. Foundations of set theoretic estimation. *Proc. IEEE*, 81(2):182–208, 1993.
- [4] D. Crisan and A. Doucet. A survey of convergence results on particle filtering methods for practitioners. *IEEE Trans. Signal Process.*, 50(3):736–746, 2002.
- [5] A. Gning, L. Mihaylova, and F. Abdallah. Mixture of uniform probability density functions for non linear state estimation using interval analysis. In *FUSION*, Edinburgh, UK, July 2010.
- [6] A. Gning, B. Ristic, and L. Mihaylova. A box particle filter for stochastic set-theoretic measurements with association uncertainty. In *FUSION*, Chicago, IL, USA, July 2011.
- [7] L. Jaulin, M. Kieffer, O. Didrit, and É. Walter. *Applied Interval Analysis*. Springer, 2001.
- [8] R. Kruse, E. Schwecke, and J. Heinsohn. *Uncertainty and vagueness in knowledge based systems*. Springer-Verlag, 1991.
- [9] R.P.S. Mahler. Multitarget Bayes filtering via first-order multitargets moments. *IEEE Trans. Aerosp. Electron. Syst.*, 39(4):1152–1178, 2003.
- [10] M. Milanese and A. Vicino. Optimal estimation theory for dynamic systems with set membership uncertainty: An overview. *Automatica*, 27(6):997–1009, 1991.
- [11] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman filter: Particle filters for tracking applications*. Artech House, 2004.
- [12] B. Ristic, D. Clark, and B.-N. Vo. Improved SMC implementation of the PHD filter. In *FUSION*, Edinburgh, UK, July 2010.
- [13] M. Schikora, W. Koch, R.L. Streit, and D. Cremers. Sequential Monte Carlo method for the iFilter. In *FUSION*, Chicago, IL, USA, July 2011.
- [14] D. Schumacher, B.-T. Vo, and B.-N. Vo. A consistent metric for performance evaluation of multi-object filters. *IEEE Trans. Signal Process.*, 56(8):3447–3457, 2008.
- [15] P. Smets. Imperfect information: Imprecision and uncertainty. *Uncertainty Management in Information Systems*, pages 225–254, 1996.
- [16] R.L. Streit. *Poisson Point Processes: Imaging, Tracking, and Sensing*. Springer, 2010.
- [17] R.L. Streit and L.D. Stone. Bayes derivation of multitarget intensity filters. In *11th International Conference on Information Fusion*, 2008.
- [18] B.-N. Vo, S. Singh, and A. Doucet. Sequential Monte Carlo methods for multi-target filtering with random finite sets. *IEEE Trans. Aerosp. Electron. Syst.*, 41(4):1224–1245, 2005.