

A COMPARATIVE STUDY OF MONTE-CARLO METHODS FOR MULTITARGET TRACKING

François Septier^{*}, Julien Cornebise[†], Simon Godsill[‡] and Yves Delignon^{*}

^{*} Signal Processing Group, Institut TELECOM / TELECOM Lille1, France

[†] Department of Computer Science, University of British Columbia, Canada

[‡] Signal Processing Laboratory, University of Cambridge, UK

ABSTRACT

In this paper, we address the problem of tracking an unknown and time varying number of targets and their states from noisy observations available at discrete intervals of time. Attention has recently focused on the role of simulation-based approaches, including Monte Carlo methods, in solving multitarget tracking problem, as these methods are able to perform well for nonlinear and non-Gaussian data models. In this paper, we present a comparative study of several Monte-Carlo methods in terms of estimation quality and complexity.

Index Terms— Tracking, Monte-Carlo methods, Particle filter, MCMC

1. INTRODUCTION

A major challenge in statistical signal processing has historically been the tracking of moving targets, let them be aircrafts, submarines, robots, or even, in less military areas, cells in a blood sample observed by microscope and fluorescence imaging, football players on a video, or even cellphones. The purpose of multiple target tracking algorithms is to detect, track and identify targets from sequences of noisy observations of the targets provided by one or more sensors.

Many approaches have been proposed to tackle the problem of multitarget detection and tracking. These range from the Kalman filter and its non-linear extensions to JPDAF trackers [1] and the Probability Hypothesis Density (PHD) filter [2]. The latter has been proposed to estimate the targets' characteristics as well as the number of targets by modeling them as a random set. However its performance degrades significantly in hostile environment - high clutter density and low target detection probability, and target identity is not preserved [3]. With the simultaneous advances in modern computational power and the developments in optimal inference for strongly non-linear models such as particle filters [4] and Markov Chain Monte Carlo (MCMC) [5], it is now possible to solve complex state space models efficiently, potentially achieving significant performance gains.

In this paper, we will compare the performances of several Monte-Carlo methods on the challenging case of tracking an unknown number of independent targets in a hostile environment of heavy level of false alarm (clutter) and very low probability of detection. We evaluate the efficiency as well as the complexity of the sequential importance resampling (SIR) algorithm [4], the resample-move [6] as well as the MCMC-based particle method [7]. Since number of objects to estimate is unknown, the optimal sub-pattern assignment (OSPA), recently proposed in [8] to evaluate the estimation quality of multitarget tracking algorithms, will be used as comparison metric.

The paper is organized as follows. In Section 2, we present the multitarget tracking by formulating the likelihood function for the observations as well as the necessary prior functions for the states of interest in order to construct the posterior distribution function. In Section 4, performances of the Monte-Carlo approaches are assessed through numerical simulations with different environmental conditions. Finally, conclusions are given in Section 5.

2. PROBLEM FORMULATION

In this study, we consider a time-varying number of targets. As a consequence, the target states are variable dimension quantities since targets can appear or disappear from the scene randomly over time. In order to model this birth and death process, we choose to use a set of existence variables \mathbf{e}_k with elements $e_{k,i} \in \{0, 1\}$ model the birth and death process for each individual target. In this formulation, the targets' kinematic vector is thus regarded as fixed dimensional quantity with N_{max} targets, each of which being active or inactive according to its existence variable $e_{k,i}$.

In this application, the aim is thus to compute, at time t_k , the filtering posterior distribution $p(\mathbf{x}_k, \mathbf{e}_k | \mathbf{z}_{0:k})$ where $\mathbf{x}_k = [x_{k,1} \cdots x_{k,N_{max}} \ y_{k,1} \cdots y_{k,N_{max}} \ \dot{x}_{k,1} \cdots \dot{x}_{k,N_{max}} \ \dot{y}_{k,1} \cdots \dot{y}_{k,N_{max}}]^T$, $\mathbf{e}_k = [e_{k,1} \cdots e_{k,N_{max}}]^T$ and $\mathbf{z}_{0:k}$ are respectively the targets' kinematics vector, the existence vector and the observation set from time t_0 to t_k .

In this work, we consider that the targets evolve independently of one another and the existence variable and the targets' kinematics are independent, so the transition probability distribution can be expanded as follows

$$p(\mathbf{x}_k, \mathbf{e}_k | \mathbf{x}_{k-1}, \mathbf{e}_{k-1}) = p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{e}_k, \mathbf{e}_{k-1}) p(\mathbf{e}_k | \mathbf{e}_{k-1}) \\ = \prod_{n=1}^{N_{max}} p(\mathbf{x}_{k,n} | \mathbf{x}_{k-1,n}, e_{k,n}, e_{k-1,n}) p(e_{k,n} | e_{k-1,n}) \quad (1)$$

We now describe the various densities involved in the computation of the filtering posterior distribution.

2.1. The prior distribution of the existence variables

Each target's existence variable will be modeled as a discrete Markov chain [9] which is independent of all other states. In this paper, the birth process is modeled as a Bernoulli like the death process:

$$p(e_{k,n} | e_{k-1,n}) = \delta(e_{k,n} - 1) [(1 - P_D)\delta(e_{k-1,n} - 1) \\ + P_B\delta(e_{k-1,n})] + \delta(e_{k,n}) [(1 - P_B)\delta(e_{k-1,n}) \\ + P_D\delta(e_{k-1,n} - 1)] \quad (2)$$

where P_B and P_D are probability values for a target to become respectively active ("alive") or inactive ("dead").

2.2. The transition probability of the targets

The transition probability of the n^{th} target can be partitioned according to $e_{k,n}$ and $e_{k-1,n}$ as follows:

$$p(\mathbf{x}_{k,n}|\mathbf{x}_{k-1,n}, e_{k-1:n}, e_{k,n}) = \begin{cases} p_b(\mathbf{x}_{k,n}) & \text{if } \{e_{k,n}, e_{k-1,n}\} = \{1, 0\} \\ p_d(\mathbf{x}_{k,n}) & \text{if } e_{k,n} = 0 \\ p_u(\mathbf{x}_{k,n}|\mathbf{x}_{k-1,n}) & \text{if } \{e_{k,n}, e_{k-1,n}\} = \{1, 1\} \end{cases} \quad (3)$$

Target Birth - The target can appear anywhere uniformly in the surveillance area of L_x by L_y . The speed is also uniformly distributed between $-V_{max}$ and V_{max} , i.e.

Target Death - For an inactive target, i.e. $e_{k,i} = 0$, we will keep the target state at some \mathbf{x}_{death} , which is the state where an inactive target is represented, i.e. $p_d(\mathbf{x}_{k,n}) = \delta(\mathbf{x}_{k,n} - \mathbf{x}_{death})$

Target Update - This case corresponds to active targets that will be updated according to the near constant velocity model

$$p_u(\mathbf{x}_{k,n}|\mathbf{x}_{k-1,n}) = \mathcal{N}(\mathbf{x}_{k,n}|\mathbf{A}_{k,n}\mathbf{x}_{k-1,n}, \mathbf{Q}_{k,n}) \quad (4)$$

where the matrices $\mathbf{A}_{k,n}$ and $\mathbf{Q}_{k,n}$ are defined as follows:

$$\mathbf{A}_{k,n} = \begin{bmatrix} \mathbf{I}_2 & \tau_k \mathbf{I}_2 \\ \mathbf{0}_2 & \mathbf{I}_2 \end{bmatrix}, \mathbf{Q}_{k,n} = \sigma_{x,n}^2 \begin{bmatrix} (\tau_k^3/3)\mathbf{I}_2 & (\tau_k^2/2)\mathbf{I}_2 \\ (\tau_k^2/2)\mathbf{I}_2 & \tau_k \mathbf{I}_2 \end{bmatrix} \quad (5)$$

with $\tau_k = t_k - t_{k-1}$.

2.3. Association Free Observation Model

An association free observation is considered in this study. At each time step t_k , a set or frame of sensor measurements $\mathbf{z}_k = \{\mathbf{z}_k^1, \dots, \mathbf{z}_k^{M_k}\}$ is received from a sensor scanning within an observation space, where M_k is the number of measurements (both target and clutter) returned by the sensor. Since any element of \mathbf{z}_k may originate from a true target or from clutter, a Poisson process is adopted to model the observations falling in a specific region. The number of the n^{th} active target measurements is randomly generated from a Poisson distribution having mean Λ_x^n whereas the number of clutter measurements have a mean number Λ_C [10]. Accordingly, the likelihood function of the observations can be expressed as

$$p(\mathbf{z}_k|\mathbf{x}_k, \mathbf{e}_k) = \frac{e^{-\mu_k}}{M_k!} \prod_{m=1}^{M_k} \lambda(\mathbf{z}_k^m|\mathbf{x}_k, \mathbf{e}_k) \quad (6)$$

where $\mu_k = \Lambda_C + \sum_{n \in \mathcal{I}} \Lambda_x^n$ is the expected total number of measurements received at time t_k and

$$\lambda(\mathbf{z}_k^m|\mathbf{x}_k, \mathbf{e}_k) = \sum_{n \in \mathcal{I}} \Lambda_x^n p_x(\mathbf{z}_k^m|\mathbf{x}_{k,n}) + \Lambda_C p_C(\mathbf{z}_k^m) \quad (7)$$

with $p_x(\cdot)$ and $p_C(\cdot)$ being the likelihood functions of target and clutter measurements and \mathcal{I} the index set corresponding to the index of active targets (e.g. $e_{k,n} = 1$) at time t_k . For the measurements issuing from the n^{th} target, $\mathbf{z}_k^m = [z_{k,x}^m, z_{k,y}^m]^T$ is considered to be drawn from $p_x(\mathbf{z}_k^m|\mathbf{x}_{k,n}) = \mathcal{N}(\mathbf{z}_k^m|\mathbf{x}_{k,n}, \Sigma_x)$. The clutter measurements are drawn uniformly in the surveillance region.

3. MONTE-CARLO METHODS FOR TRACKING

In Bayesian framework, we are aimed at computing the posterior distribution $p(\mathbf{s}_{0:k}|\mathbf{z}_{0:k})$, with $\mathbf{s}_k = [\mathbf{x}_k, \mathbf{e}_k]$, recursively by

$$p(\mathbf{s}_{0:k}|\mathbf{z}_{0:k}) \propto p(\mathbf{z}_k|\mathbf{s}_k)p(\mathbf{s}_k|\mathbf{s}_{k-1})p(\mathbf{s}_{0:k-1}|\mathbf{z}_{0:k-1}) \quad (8)$$

Unfortunately in multitarget tracking, like in many other applications, this distribution is analytically intractable. We therefore turn to Monte-Carlo methods to provide an efficient numerical approximation strategy for recursive estimation of the state of interest.

In this work, we focus our study on three Monte-Carlo methods used for Bayesian filtering: the sequential importance resampling (SIR) algorithm [4], the resample-move [6] as well as the MCMC-based particle method [7]. In all these approaches, the posterior distribution is approximated, at each time, by a set of N_p unweighted particles, i.e. at time t_{k-1} :

$$\hat{p}(\mathbf{s}_{0:k-1}|\mathbf{z}_{0:k-1}) = \frac{1}{N_p} \sum_{j=1}^{N_p} \delta(\mathbf{s}_{0:k-1} - \mathbf{s}_{0:k-1}^{(j)}) \quad (9)$$

The difference between these methods relies on the sampling strategy of $\{\mathbf{s}_{0:k}^{(j)}\}_{j=1}^{N_p}$ from the posterior $p(\mathbf{s}_{0:k}|\mathbf{z}_{0:k})$ by using the previous samples $\{\mathbf{s}_{0:k-1}^{(j)}\}_{j=1}^{N_p}$. Let us now describe these algorithms applied to multitarget tracking.

3.1. Sequential Importance Resampling

The SIR was proposed in [4] under the name of ‘‘bootstrap filter’’. This algorithm is derived from the Sequential Importance Sampling (SIS) algorithm by choosing the importance density to be the prior one, Eq. (1), and by performing the resampling step at every time index. Applied to the multitarget tracking problem described in Section 2, this algorithm, summarized in Algorithm 1, has an asymptotic complexity of $\mathcal{O}(M_k N_p N_{max})$ per time index.

Algorithm 1 Sequential Importance Resampling (SIR) at time t_k

- 1: **for** $i = 1, \dots, N_p$ **do**
 - 2: Sample $\{\mathbf{x}_k^{(i)}, \mathbf{e}_k^{(i)}\} \sim p(\mathbf{x}_k, \mathbf{e}_k|\mathbf{x}_{k-1}, \mathbf{e}_{k-1})$
 - 3: Compute the IS weight: $w_k^{(i)} = w_{k-1}^{(i)} p(\mathbf{z}_k|\mathbf{x}_k^{(i)}, \mathbf{e}_k^{(i)})$
 - 4: **end for**
 - 5: Normalization of the weights
 - 6: Resampling Step
-

3.2. Resample-Move

Gilks and Berzuini introduced the Resample-Move algorithm [6], which is one of the first methods to make use of MCMC in a sequential state estimation problem. The algorithm takes the output of a standard particle filter after resampling, and applies a MCMC move to each particle using the posterior distribution $p(\mathbf{s}_{0:k}|\mathbf{z}_{0:k})$ as the stationary distribution. The main idea is that moving each particle after resampling will improve the diversity of the particle population, and this can only improve the representation of the posterior distribution. As mentioned in [11], the Resample-Move algorithm does not require each MCMC chain initiated to have a burn-in period. This is because the output of the particle filter is already a reasonable approximation to the posterior filtering distribution.

For this multitarget tracking problem, the MCMC-move consists in sampling successively each of the individual targets and their associated existence variables by using a Metropolis-Hastings-within-Gibbs step with the prior distribution, $p(\mathbf{x}_{k,n}, e_{k,n}|\mathbf{x}_{k-1,n}, e_{k-1,n})$, as proposal. The asymptotic complexity of this Algorithm 2 is $\mathcal{O}(M_k N_p N_{max} N_{it.})$ per time index.

Algorithm 2 Resample-Move at time t_k

```
1: Steps 1 to 6 of the SIR
2: for  $i = 1, \dots, N_p$  do
3:   for  $k = 1, \dots, N_{it.}$  do
4:     for  $n = 1, \dots, N_{max}$  do
5:       MCMC move of the  $n^{th}$  target using a MH-within Gibbs
6:     end for
7:   end for
8: end for
```

3.3. MCMC-based Particle Algorithm

Markov chain Monte Carlo (MCMC) methods are generally more effective than PFs in high-dimensional spaces. Their traditional formulation, however, allows sampling from probability distributions in a non-sequential fashion. Recently, sequential MCMC schemes were proposed by [12–15]. These approaches are distinct from the Resample-Move scheme [6] in particle filter where the MCMC algorithm is used to rejuvenate degenerate samples following importance sampling resampling. These methods [12–15] use neither resampling nor importance sampling.

In the MCMC-based particle algorithm [15], a MCMC procedure is used to produce samples from the posterior distribution, $p(\mathbf{s}_{0:k} | \mathbf{z}_{0:k})$, by approximating in Eq. (8), $p(\mathbf{s}_{0:k-1} | \mathbf{z}_{0:k-1})$ with the empirical distribution obtained at the previous time index, Eq. (9). The MCMC-move used in this scheme (Step 7 in Algorithm 3) is the same than the one used in the Resample-Move. The asymptotic complexity of this method is $\mathcal{O}(M_k N_p N_{max})$ if the initial burn-in is chosen as $N_{Burn-in} = \alpha N_p$ with $0 < \alpha < 1$.

Algorithm 3 MCMC-Based Particle Algorithm at time t_k

```
1: Sample  $\{\mathbf{x}_{0:k-1}^{(1)}, \mathbf{e}_{0:k-1}^{(1)}\} \sim \hat{p}(\mathbf{x}_{0:k-1}, \mathbf{e}_{0:k-1} | \mathbf{z}_{1:k-1})$  and
 $\{\mathbf{x}_k^{(1)}, \mathbf{e}_k^{(1)}\} \sim p(\mathbf{x}_k, \mathbf{e}_k | \mathbf{x}_{k-1}^{(1)}, \mathbf{e}_{k-1}^{(1)})$ 
2: for  $i = 2, \dots, N_p + N_{Burn-in}$  do
3:   Sample  $\{\mathbf{x}_{0:k-1}^*, \mathbf{e}_{0:k-1}^*\} \sim \hat{p}(\mathbf{x}_{0:k-1}, \mathbf{e}_{0:k-1} | \mathbf{z}_{1:k-1})$ 
4:   Sample  $\{\mathbf{x}_k^*, \mathbf{e}_k^*\} \sim p(\mathbf{x}_k, \mathbf{e}_k | \mathbf{x}_{k-1}^*, \mathbf{e}_{k-1}^*)$ 
5:   Accept or Reject the move  $\{\mathbf{x}_{0:k}^{(i)}, \mathbf{e}_{0:k}^{(i)}\} = \{\mathbf{x}_{0:k}^*, \mathbf{e}_{0:k}^*\}$ 
6:   for  $n = 1, \dots, N_{max}$  do
7:     MCMC move of the  $n^{th}$  target using a MH-within Gibbs
8:   end for
9: end for
```

4. NUMERICAL SIMULATIONS

Consider a time-varying number of targets (max of 5) moving using the near-constant velocity model defined in Eq. (4) as shown in Fig. 1. Note that targets 1-3 are born at time $k = 1$, targets 4-5 are born at time $k = 25$, and target 1 dies at time $k = 50$. The starting and stopping positions for each track are labelled with a circle and triangle respectively. Individual target motions follow the near constant velocity motion model with a sampling period of $\tau_k = 3s$ and process noise variance $\sigma_{x,n}^2 = 0.5$. The observation scene is a square of $5000m \times 5000m$. The parameters involved in the observation model are $\Lambda_C = 20$, $\Sigma_x = 100 \times \mathbf{I}_2$. Numerical simulations have been performed with two different values for the average number of measurements issued from a specific target, i.e. Scenario 1 : $\Lambda_x^n = 1$ (the probability that a target returns at least one measurement is thus $P_{Detect} = 0.63$) and Scenario 2 : $\Lambda_x^n = 1.5$

($P_{Detect} = 0.78$). Concerning the MCMC-based Particle algorithm, we choose $N_{Burn-in} = N_p/10$. To capture the average performance, we run 200 Monte-Carlo (MC) trials for each scenario with the same target tracks but independently generated measurements.

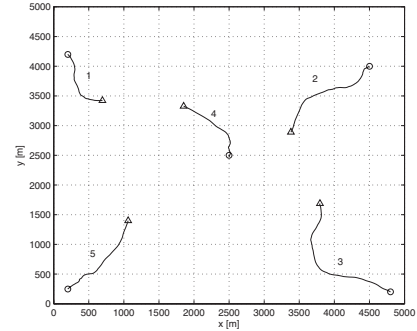


Fig. 1. True tracks in the xy plane. Targets move with near-constant velocity along the paths shown. Start/Stop positions are shown with \bigcirc/\triangle

Figures 2-(a) show the MC average of the estimated cardinality. From this figure, we can see that the various Monte-Carlo algorithms are capable of adequately tracking the varying number of targets in the observation scene. However, the MCMC-based particle algorithm slightly outperforms the other ones in terms of cardinality error. The OSPA distance analysis (see Figs. 2-(b,c)) allows us to compare tracking performances in terms of both localization and cardinality errors [8]. The MCMC-based particle algorithm exhibits better tracking performances than both the SIR and the resample-move. The introduction of the MCMC move after the resampling step of the SIR allows to increase tracking performance. But even with 10 iterations ($N_{it.} = 10$), the Resample-Move still obtains higher OSPA distance than the MCMC-based particle (cf. Fig. 2-(c-1)).

Finally from Figure 3, as expected by the asymptotic complexity of these algorithms (Section 3), the runtime of all these methods grows linearly with the number of particles.

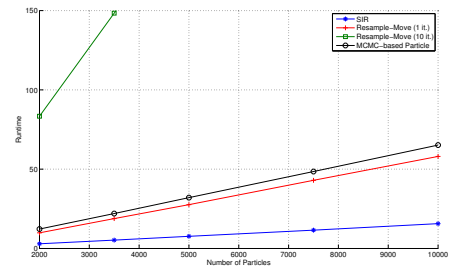
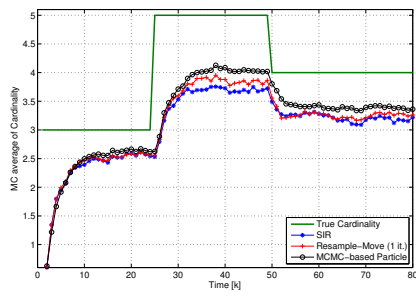


Fig. 3. Runtime of the different MC algorithms

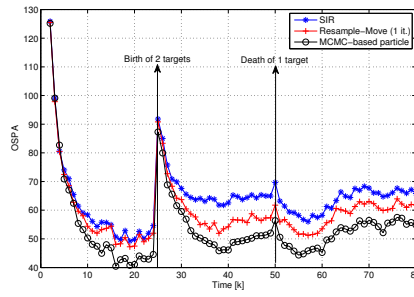
5. CONCLUSION

In this paper, we have compared various Monte-Carlo algorithms that have been developed over the past years for sequentially approximating the filtering distribution in a general state-space model. These methods have been successfully applied to detect, track and identify multiple targets in very low probability of detection and

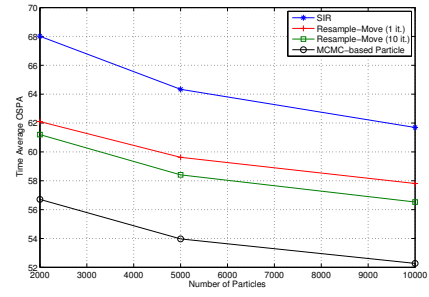
Scenario 1 : $P_{\text{Detect}} = 0.63$



(a-1)

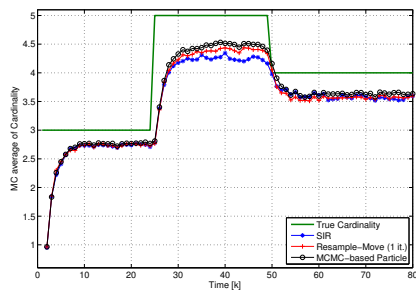


(b-1)

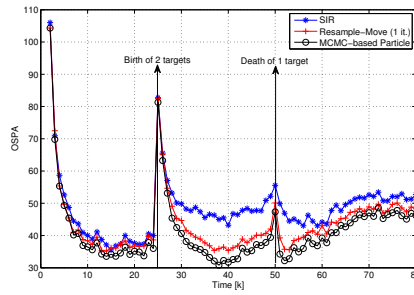


(c-1)

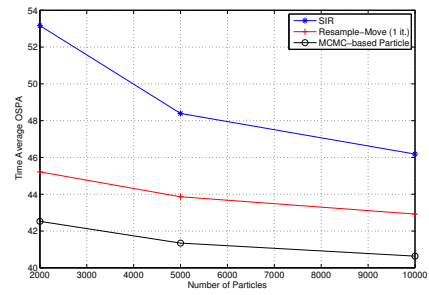
Scenario 2 : $P_{\text{Detect}} = 0.78$



(a-2)



(b-2)



(c-2)

Fig. 2. Tracking performance with $N_p = 5000$. Showing : Monte-Carlo average of the estimated cardinality versus time index (a-1,2). OSPA distance versus time index (b-1,2). Time average OSPA distance versus the number of particles (c-1,2) (OSPA parameters : order $p = 1$ and cut-off $c = 150$).

in hostile environments with heavy clutter. Simulations show that the MCMC-based Particle approach exhibits better tracking performance and thus clearly represents interesting alternatives to SMC methods.

6. REFERENCES

- [1] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*, Artech House, 1999.
- [2] R. Mahler, "A theoretical foundation for the stein-winter probability hypothesis density (phd) multi-target tracking approach," in *Proc. MSS Nat'l Symp. on Sensor and Data Fusion*, San Antonio TX, 2000, vol. I.
- [3] W. Ng, J. Li, S. Godsill, and J. Vermaak, "A hybrid approach for online joint detection and tracking for multiple targets," in *IEEE Aerospace Conference*.
- [4] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation," in *Radar and Signal Processing, IEE Proceedings*, Apr. 1993, vol. 140, pp. 107–113.
- [5] C. P. Robert and G. Casella, *Monte-Carlo Statistical Methods*, New York: Springer, second edition edition, 2004.
- [6] W. R. Gilks and C. Berzuini, "Following a Moving Target-Monte Carlo Inference for Dynamic Bayesian Models," *J. Roy. Statist. Soc. Ser. B*, vol. 63, pp. 127–146, 2001.
- [7] F. Septier, S. K. Pang, A. Carmi, and S. Godsill, "On MCMC-Based Particle Methods for Bayesian Filtering : Application to Multitarget Tracking," in *IEEE CAMSAP*, Aruba, Dutch Antilles, Dec. 2009.
- [8] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A Consistent Metric for Performance Evaluation of Multi-Object Filters," *IEEE Trans. Sig. Proc.*, vol. 56, no. 8, pp. 3447–3457, 2008.
- [9] J. Vermaak, S. Maskell, M. Briens, and P. Perez, "A Unifying Framework for Multi-Target Tracking and Existence," in *8th International Conference on Information Fusion*, Jul. 2005, vol. 1, pp. 25–28.
- [10] K. Gilholm, S.J. Godsill, S. Maskell, and D. Salmond, "Poisson Models for Extended Target and Group Tracking," *Proc. SPIE Conf. Signal and Data Proc. of Small Targets*, Aug. 2005.
- [11] O. Cappé, S.J. Godsill, and E. Moulines, "An overview of existing methods and recent advances in sequential monte carlo," *Proceedings of the IEEE*, vol. 95, no. 5, pp. 899–924, May 2007.
- [12] C. Berzuini, N. G. Best, W. R. Gilks, and C. Larizza, "Dynamic Conditional Independence Models and Markov Chain Monte Carlo Methods," *Journal of the American Statistical Association*, vol. 440, pp. 1403–1412, Dec. 1997.
- [13] Z. Khan, T. Blach, and F. Dellaert, "MCMC-based Particle Filtering for Tracking a Variable Number of Interacting Targets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, pp. 1805–1819, Nov. 2005.
- [14] A. Golightly and D. J. Wilkinson, "Bayesian Sequential Inference for Nonlinear Multivariate Diffusions," *Statistics and Computing*, pp. 323–338, Aug. 2006.
- [15] F. Septier, A. Carmi, S. K. Pang, and S. J. Godsill, "Multiple Object Tracking Using Evolutionary and Hybrid MCMC-Based Particle Algorithms," in *15th IFAC Symposium on System Identification*, Saint-Malo, France, Jul. 2009.