Aalto University School of Science

# **Tracking of Multiple Interacting Targets Using Rao-Blackwellized Particle Filters**

Joona Karjalainen

Mat-2.4108 – Independent Research Project in Applied Mathematics October 12, 2015

Supervisor: Prof. Kai Virtanen Department of Mathematics and Systems Analysis Aalto University

Instructor: M.Sc. Juho Kokkala Department of Neuroscience and Biomedical Engineering / Department of Computer Science Aalto University

The document can be stored and made available to the public on the open internet pages of Aalto University. All other rights are reserved.

# Contents

1	Introduction		
2	Tracking Methods2.1Kalman Filtering and Smoothing2.2Rao-Blackwellized Monte Carlo Data Association	2 . 2 . 4	
3	Extension of RBMCDA for Interacting Targets	6	
4	Discretization of Dynamic Models	7	
5	Models of Target Dynamics5.1Mean-Reverting Model for Independent Targets (I-MR)5.2Mean-Reverting Model with a Common Center5.3Independent Targets with White Noise Accelerations (I-WNA)5.4Follower Model	8 . 8 . 10 . 11 . 11	
6	Parameter Estimation Using Particle Gibbs	12	
7	Optimal Subpattern Assignment for Tracks	14	
8	Simulations8.1Mean-Reverting Targets with a Common Center in Clutter8.2Group Tracking in Clutter	<b>15</b> . 16 . 22	
9	Conclusions	28	

## **1** Introduction

Multiple target tracking methods aim to estimate the states of targets using noisy measurements. These methods have been used in diverse application areas of multiple object systems research, such as traffic control [Blackman, 1986], oceanography [Lane et al., 1998], computer vision [Cham and Rehg, 1999] and estimation of wild animal populations [Abbas, 2011] [Kokkala and Särkkä, 2015].

Multiple target tracking is inherently more complicated than tracking single targets. Whereas the main challenges in single target scenarios lie in defining a dynamic model of the target and detecting clutter measurements, in multiple target scenarios one has to be able to solve the problem of data association, i.e., assigning the measurements to the different targets. The problem of data association is complicated by that the number of targets is generally not known, but has to be estimated from the measurements.

Different approaches to multiple target tracking exist, ranging from heuristics (such as nearest-neighbour association [Blackman, 1986]) to methods based on probabilistic models. The latter includes methods such as Joint Probabilistic Data Association (JPDA) [Fortmann et al., 1980], Probability Hypothesis Density filters (PHD) [Mahler, 2003] and Rao-Blackwellized Monte Carlo Data Association (RBMCDA) [Särkkä et al., 2004]. These methods differ, e.g., in the way they approximate the probability distributions of the quantities of interest. The main focus of this independent research project is RBMCDA, which uses Monte Carlo sampling to approximate the distribution of the data associations.

The targets are often assumed to behave independently of each other. Such models are widely applicable, especially in situations where there is no sufficient prior information about the behaviour of the targets. However, in some applications it is natural to assume that there are interactions between the target movements, e.g., when the targets are expected to move in groups, or if the targets are expected to avoid or approach certain locations. The term interaction is used here to refer to any probabilistic dependencies between the target movements, regardless of whether the targets are truly aware of each other. When these interactions are modeled, one should expect improvements in the state estimates, and more accurate data associations and estimates of the number of targets.

Several approaches to modeling target interactions are found in the literature. Khan et al. [2004] use Markov random fields to learn the nature of the interactions between the targets in real-time. Pang et al. [2011] propose group models for tracking of multiple interacting targets that require simpler computational procedures than Markov random fields. They consider both repulsive and attractive forces between the targets, and present how the computations can be carried out efficiently with these models.

Models of the target dynamics generally contain unknown parameters, such as variances related to the target accelerations, and strengths of the attractive forces between the targets. These parameters can be tuned through experiments, or chosen based on prior information in the model construction phase. Another approach is to estimate these parameters from the data. For example, Kokkala and Särkkä [2015] showed how Particle Markov Chain Monte Carlo (PMCMC) methods can be used with RBMCDA to compute posterior distributions of the parameters.

The objective of this research project is to extend the Rao-Blackwellized Monte Carlo Data Association method to tracking of multiple interacting targets. The approach taken to modeling interactions is to define them in the dynamic models of the targets. The parameters of the models are estimated using the Particle Gibbs method. For independent targets, this is done using the approach introduced in [Kokkala and Särkkä, 2015]. A slight modification is made to apply this approach to interacting targets. Models with and without interactions are compared using simulations.

This report is organized as follows. Kalman filtering, smoothing and RBMCDA are briefly reviewed in Section 2. The extension of RBMCDA for interacting targets is presented in Section 3. The procedure for discretizing the dynamic models of the targets is described in Section 4. The models of the target dynamics are developed in Section 5. The parameter estimation method is described in Section 6. The performance measure used in the simulations, the OSPA-T metric, is summarized in Section 7. The models are compared using simulated data in Section 8. Conclusions and directions for future work are presented in Section 9.

## 2 Tracking Methods

The necessary preliminaries, namely Kalman filtering, smoothing, and RBMCDA are reviewed in this section. Some background in estimation and Bayesian statistics is assumed. For detailed introductions, see, e.g, [Grewal and Andrews, 2008], [Särkkä, 2013] and [Särkkä et al., 2007].

#### 2.1 Kalman Filtering and Smoothing

The Kalman filter [Kalman, 1960] is an optimal estimator for the state of a linear dynamic system. It is used in multiple target tracking methods to solve the subproblem of state estimation conditional on the data associations and the measurements. The models of the target dynamics are expressed as linear time-invariant dynamic systems in discrete time, defined by

$$\mathbf{x}_t = \mathbf{F}_t \mathbf{x}_{t-1} + \mathbf{w}_t \tag{1}$$

$$\mathbf{y}_t = \mathbf{H}_t \mathbf{x}_t + \mathbf{v}_t, \tag{2}$$

where

- $\mathbf{x}_t$  is the state vector of the targets at time step t.
- $\mathbf{y}_t$  is the measurement at time step t.
- $\mathbf{w}_t$  is a white noise process that represents deviations from the dynamic model.
- $\mathbf{v}_t$  is a white noise process that represents the measurement errors.
- $\mathbf{F}_t$  is the transition matrix that describes the linear dynamics of the system at time step *t*.
- **H**<sub>t</sub> is the measurement model that describes the relationship between the measurements and the state at time step t.

It is often assumed that  $\mathbf{w}_t$  and  $\mathbf{v}_t$  are normally distributed with zero mean, i.e.,

$$\mathbf{w}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_t) \qquad \qquad \mathbf{v}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_t), \tag{3}$$

where  $\mathbf{Q}_t$  is the process noise covariance matrix,  $\mathbf{R}_t$  is the measurement error covariance matrix, and  $\mathbf{0}$  is the zero vector. Once the matrices  $\mathbf{F}_t$ ,  $\mathbf{H}_t$ ,  $\mathbf{Q}_t$ , and  $\mathbf{R}_t$  are defined, the Kalman filter can be used to compute the optimal estimate of  $\mathbf{x}_t$  based on the measurements.

The Bayesian interpretation of the Kalman filter is that it is an exact method to combine the measurements and the prior information about the states to yield the posterior distribution. The linear dynamic system with normally distributed random terms can be formulated as

$$p(\mathbf{x}_0) = \mathcal{N}(\mathbf{m}_0, \mathbf{P}_0) \tag{4}$$

$$p(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{F}_t \mathbf{x}_{t-1}, \mathbf{Q}_t)$$
(5)

$$p(\mathbf{y}_t | \mathbf{x}_t) = \mathcal{N}(\mathbf{H}_t \mathbf{x}_t, \mathbf{R}_t), \tag{6}$$

where  $\mathbf{x}_0$  is the initial state,  $\mathbf{m}_0$  is the prior mean, and  $\mathbf{P}_0$  is the prior covariance matrix. The Kalman filter is then understood as a method that solves the conditional mean and conditional covariance terms  $\mathbf{m}_t$ ,  $\mathbf{P}_t$ ,  $\mathbf{m}_t^-$ , and  $\mathbf{P}_t^-$ , defined by

$$p(\mathbf{x}_t | \mathbf{y}_1 \dots \mathbf{y}_{t-1}, \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{m}_t^-, \mathbf{P}_t^-)$$
(7)

$$p(\mathbf{x}_t | \mathbf{y}_1 \dots \mathbf{y}_t) = \mathcal{N}(\mathbf{m}_t, \mathbf{P}_t), \tag{8}$$

where  $\mathbf{m}_t$  is referred to as the filtered mean of  $\mathbf{x}_t$ . Solving the means and covariances in equations (7) and (8) is commonly referred to as the prediction step and the update step, respectively.

As new measurements from time steps  $t + 1 \dots T$  are obtained, the states  $\mathbf{x}_t$  can be updated using all the measurements. This is referred to as smoothing, and the

corresponding distribution can be expressed as

$$p(\mathbf{x}_t | \mathbf{y}_1 \dots \mathbf{y}_T) = \mathcal{N}(\mathbf{m}_t^s, \mathbf{P}_t^s), \tag{9}$$

where  $\mathbf{m}_t^s$  is referred to as the smoothed mean of  $\mathbf{x}_t$ . The means and covariances  $\mathbf{m}_t^s$  and  $\mathbf{P}_t^s$  can be solved exactly using the Rauch-Tung-Striebel smoother [Rauch et al., 1965]. Smoothing can be combined with multiple target tracking methods to estimate the history of the target states, e.g., to estimate the starting points of the targets.

The Kalman filtering and smoothing equations are omitted here, and can be found in many textbooks, e.g., [Grewal and Andrews, 2008].

#### 2.2 Rao-Blackwellized Monte Carlo Data Association

Rao-Blackwellized Monte Carlo Data Association (RBMCDA) [Särkkä et al., 2007] is a multiple target tracking method that allows for unknown and varying numbers of targets. Clutter measurements are treated in a probabilistic way.

In RBMCDA, the targets are assumed to have linear dynamics with Gaussian process noise, as well as a linear measurement model (as in Equations (1) and (2)). Each measurement is assumed to either originate from exactly one target, or to be a clutter measurement. The data association history is contained in the vector c, defined by

$$c_t = \begin{cases} k & \text{if the measurement } y_t \text{ is assigned to target } k \\ 0 & \text{if } y_t \text{ is considered a clutter measurement.} \end{cases}$$
(10)

RBMCDA uses a method called *particle filtering* [Doucet et al., 2000] to assign the measurements to the targets, and Kalman filtering for estimating the states of the targets. The output of RBMCDA is a set of weighted Monte Carlo samples from the posterior distribution

$$p(c_{1...T}|\mathbf{y}_{1...T}),$$
 (11)

where T is the time step when the last measurement is obtained.

At each time step t, RBMCDA approximates the posterior distribution  $p(c_{1...t}|\mathbf{y}_{1...t})$  by a finite number of point masses (or *particles*). Each particle contains a data association history and a particle weight  $w_t$  that represents the probability mass of the particle. Using large numbers of particles results in more accurate inference, but increases the computational expense of the method.

RBMCDA uses Kalman filtering to solve  $p(\mathbf{x}_t | \mathbf{y}_t, c_{1...t})$ , i.e., the exact posterior distributions of the target states conditional on the measurements and the data association history. This procedure is referred to as Rao-Blackwellization [Casella and Robert, 1996]. It reduces the variance of the particle weights, resulting in better efficiency of the method than would be obtained with Monte Carlo based methods. If the model of the target dynamics is non-linear or the error terms are not normally distributed, the Kalman filter does not yield the exact posterior distributions, and other estimators may be considered.

As RBMCDA is used to compute Monte Carlo samples, it does not directly offer a way to find the most probable data associations. When a point estimate is needed, one may use the data association history with the largest weight, as was done in [Särkkä et al., 2007]. The weight of a data association history is here defined as the combined weight of the particles that contain that particular history. Once the data associations are fixed, estimating the target states is reduced to Kalman filtering and smoothing.

The following are defined before applying RBMCDA:

- The dynamic model of the targets (Equations (1)-(2)).
- The prior distribution of the initial state  $\mathcal{N}(\mathbf{m}_0, \mathbf{P}_0)$ . If there is no prior information about the locations of the targets, one may use large variances in  $\mathbf{P}_0$ .
- The conditional prior distribution of the data associations  $p(c_t = j | c_{1..t-1})$ . This can be defined uniformly so that the prior probability of choosing any of the already seen targets is the same.
- The prior distribution of the clutter measurements  $p(\mathbf{y}_t | c_t = 0)$ . If there is no prior information about the locations of the clutter measurements, one may use a uniform distribution.
- The number of particles *N*.

The data associations are then obtained by proceeding through the following steps for all measurements, starting from t = 1:

- 1. When a measurement  $\mathbf{y}_t$  is received, for all particles  $i \in \{1...N\}$ :
  - (a) Use the Kalman filter prediction step for each target. Use the Kalman filter update step separately for each target  $(\mathbf{m}_{k}^{(i)}, \mathbf{P}_{k}^{(i)})$  to obtain the likelihoods  $p(\mathbf{y}_{t}|c_{t} = k, c_{1..t-1}^{(i)}, \mathbf{y}_{1...t-1})$  for data associations  $k \in \{1...M_{t}^{(i)}\}$ , where  $M_{t}^{(i)}$  is the number of targets already seen, i.e.,  $\max_{k=1...t-1} c_{k}^{(i)}$ . Use the Kalman update step to obtain the likelihood for a new target,

$$p(\mathbf{y}_t | c_t = M_t^{(i)} + 1, c_{1..t-1}^{(i)}, \mathbf{y}_{1...t-1}).$$

(b) Draw  $c_t^{(i)}$ , i.e., the target associated with the newest observation  $y_t$ , from the discrete distribution defined by

$$p(c_t = j) = \frac{p(c_t = j | c_{1...t-1}^{(i)}) p(\mathbf{y}_t | c_{1...t-1}^{(i)}, c_t = j, \mathbf{y}_{1...t-1})}{\sum_{j=0}^{M_t^{(i)}+1} p(c_t = j | c_{1...t-1}^{(i)}) p(\mathbf{y}_t | c_{1..t-1}^{(i)}, c_t = j, \mathbf{y}_{1...t-1})}, \quad (12)$$

where  $j = 0...M_t^{(i)} + 1$ . The probabilities  $p(c_t = j|c_{1...t-1}^{(i)})$  are obtained from the prior distribution of the data associations, and  $p(\mathbf{y}_t|c_{1...t}^{(i)}, \mathbf{y}_{1...t-1})$  is obtained using the Kalman filter in step (a).

(c) Update the particle weight:

$$w^{(i)} \leftarrow w^{(i)} \sum_{j=0}^{M_t^{(i)}+1} p(c_t = j | c_{1...t-1}^{(i)}) p(\mathbf{y}_t | c_{1..t-1}^{(i)}, c_t = j, \mathbf{y}_{1...t-1}).$$

- 2. Normalize the particle weights to sum to 1.
- 3. Resample: Replace the particles with *N* samples from the discrete distribution defined by the weights  $p(c_{1...t}^{(i)}) = w^{(i)}$ . Set all weights to 1/N.

After the last measurement, the particles represent a discrete approximation of the posterior distribution of the data associations. As the Kalman filter is a recursive method, the means and covariances of the target states can be stored with the particles for further processing to avoid repeating the same computations in Step 1(a). In practice, the updated mean and covariance are stored for the target chosen in Step 1(b), and the predicted means and covariances are stored for the other targets. In [Särkkä et al., 2007], target death processing was also used in RBMCDA. For simplicity, it is not used in this research project. However, if no new measurements are obtained from a target, the posterior probability of associating it with any measurement decreases. Eventually particles containing such a target is expected to be removed in the resampling step (3).

## **3** Extension of RBMCDA for Interacting Targets

RBMCDA does not directly offer a way to model interactions, as the means and covariance matrices of the states are processed separately for each target. For this reason, RBMCDA is here extended by considering the joint state of all targets instead of the per-target states. When the dynamic model of the targets does not include correlations between different targets, this method yields the same estimates as RBMCDA without the extension.

Let the states of the targets  $1 \dots K$  be  $\mathbf{x}_1 \dots \mathbf{x}_K$ , each state being of size  $h \times 1$ . Let  $\mathbf{x}_c$  be a column vector of states that are not associated with any particular target, but with the whole system (e.g., the common center of the targets). Define the joint state  $\mathbf{x}$  of the targets  $1 \dots K$  as

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_c \\ \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_K \end{bmatrix}.$$
(13)

Let **m** and **P** be the estimated mean and covariance matrix of the joint state. Let the prior distribution of the initial state of a target be defined by  $\mathbf{m}_0$  and  $\mathbf{P}_0$ . As the number of targets varies, the transition matrix  $\mathbf{F}_t$  (in Equation (1)) must be defined for each number of targets, which is denoted by  $\mathbf{F}_t^{(K)}$  for *K* targets.

In this extension, one starts with **m** and **P** being equal to the prior mean and covariance of  $\mathbf{x}_c$ . Step 1 (a) in the previous section is replaced with the following:

- 1. When a measurement  $\mathbf{y}_t$  is received, for all particles  $i \in \{1...N\}$ :
  - (a) Use the Kalman filter prediction step for the joint state using  $\mathbf{F}_{t}^{M_{t}^{(i)}}$ . Use the Kalman filter update step for the joint state to obtain the the likelihoods  $p(\mathbf{y}_{t}|c_{t} = k, c_{1..t-1}^{(i)}, \mathbf{y}_{1..t-1})$  for data associations  $k \in \{1...M_{t}^{(i)}\}$ . Only the measurement model  $\mathbf{H}_{t}$  (Equation (2)) is different for each target. Use the Kalman update step for the *extended mean and covariance* to obtain the likelihood for a new target using  $\mathbf{F}_{t}^{M_{t}^{(i)}+1}$ . The extended mean  $\mathbf{m}_{e}$  and covariance  $\mathbf{P}_{e}$  are defined by

$$\mathbf{m}_e = \begin{bmatrix} \mathbf{m} \\ \mathbf{m}_0 \end{bmatrix} \tag{14}$$

and

$$\mathbf{P}_e = \begin{bmatrix} \mathbf{P} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_0 \end{bmatrix}. \tag{15}$$

This yields the likelihood  $p(\mathbf{y}_t | c_t = M_t^{(i)} + 1, c_{1..t-1}^{(i)}, \mathbf{y}_{1...t-1})$ . Steps from 1(b) to 3 are as in the previous section.

The updated joint mean and covariance for the chosen data association  $c_t$  can be stored with the particles to make use of the recursive nature of the Kalman filter. Unlike in the original RMBCDA, the output of the Kalman filter prediction step is not stored.

The definitions of  $\mathbf{m}_e$  and  $\mathbf{P}_e$  imply that the new target is independent of the other targets before the first measurement is associated to it. This means that the prior distribution of the new target does not change over time, which seems like a reasonable assumption for some applications. The interactions between the new target and the other targets are taken into account at all time steps after the first association.

As the original RBMCDA, this extended method is also a Rao-Blackwellized particle filter, since it uses Kalman filtering to compute the likelihoods, and particle filtering to sample from the posterior distribution of the data associations.

## **4** Discretization of Dynamic Models

RBMCDA requires the discrete-time transition matrix and process noise covariance matrix of the model. However, the target dynamics are modeled in continuous time. The motivation for this is that continuous-time models are straightforward to define and can be discretized as needed, whereas starting with a discrete model can be difficult. In particular, the discrete-time covariance matrices are not, in general, of any simple form due to correlations between the target states. The discretization is not an approximation of the continuous-time model, but it yields the exact transition matrix and the process noise covariance matrix for a given time interval. If the continuous-time model is linear with Gaussian noise, then so is the discretized model.

The continuous-time models of the target dynamics used in this research project are of the form

$$\dot{\mathbf{x}}(t) = \mathbf{F}_c \mathbf{x}(t) + \mathbf{L}_c \mathbf{w}(t)$$
(16)

where  $\mathbf{F}_c$  and  $\mathbf{L}_c$  are analoguous to the discrete case in Equation (1) and  $\mathbf{w}(t)$  is a white-noise process with spectral density  $\mathbf{Q}_c$ .

The discrete-time covariance matrix can be sometimes solved analytically for simple models. However, as the size of the state vector may vary, such a discretization would have to be solved for an arbitrary number of targets. In practice, the discrete-time process covariance matrix is evaluated numerically. This is done using the matrix fraction decomposition [Grewal and Andrews, 2008]:

$$\begin{bmatrix} \mathbf{C} \\ \mathbf{D} \end{bmatrix} = exp\left( \begin{bmatrix} \mathbf{F}_c & \mathbf{L}_c \mathbf{Q}_c \mathbf{L}_c^T \\ 0 & -\mathbf{F}_c \end{bmatrix} \Delta t \right) \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix},$$
(17)

where exp denotes the matrix exponential. Both the zero matrix **0** and the identity matrix **I** are the same size as  $\mathbf{F}_c$ . The discrete-time process noise covariance is then given by

$$\mathbf{Q} = \mathbf{C}\mathbf{D}^{-1}.\tag{18}$$

The discrete-time transition matrix is given by

$$\mathbf{F}_t = exp(\mathbf{F}_c \Delta t). \tag{19}$$

## 5 Models of Target Dynamics

In this section, two models for independent targets and two models for interacting targets are described. The models are first defined in continuous time, and then discretized to make use of the Kalman filter for state estimation. The models with independent targets are described in Sections 5.1 and 5.3. These models are used for comparisons in the simulations. The interaction models developed for the purposes of this research project are described in Sections 5.2 and 5.4.

#### 5.1 Mean-Reverting Model for Independent Targets (I-MR)

In this model, each target is allowed to approach a different fixed location. These locations are unknown constants that are estimated from the data. The targets do not

move in straight lines, but random noise is added to the target locations. The state vector contains the locations of the targets  $(x_j, y_j)$  and the locations they approach  $(x_{j,cp}, y_{j,cp})$ . For each target *j*, this model can be written in continuous time as

$$\dot{\mathbf{x}}_j = \lambda (\mathbf{x}_{j,cp} - \mathbf{x}_j) + \sqrt{q} \mathbf{w}_j(t), \qquad (20)$$

where  $\lambda$  is a rate parameter, q is the spectral density of the process noise and  $\mathbf{w}_j(t)$  is a two-dimensional white noise process. This model can be viewed as a generalization of the Ornstein-Uhlenbeck model (see, e.g., [Papoulis, 1984]).

By comparing Equations (20) and (16), it is seen that the continuous-time model is

This continuous-time model can be discretized, e.g., using the matrix fraction decomposition, which yields the block diagonal matrices

$$\mathbf{F}_{t} = \begin{bmatrix} e^{-\lambda\Delta t} & 0 & 1 - e^{-\lambda\Delta t} & 0 \\ 0 & e^{-\lambda\Delta t} & 0 & 1 - e^{-\lambda\Delta t} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ & & & \ddots \end{bmatrix}$$

In practice, one may use the  $4 \times 4$  blocks of  $\mathbf{F}_t$  and  $\mathbf{Q}_t$  instead of the full matrices, and apply them separately to each target to speed up computations. The measurement model for a measurement from target *k* is

#### 5.2 Mean-Reverting Model with a Common Center

This model is similar to the mean-reverting model for independent targets, with the difference that in this model all targets approach a fixed common center. The transition matrices and the discrete-time process noise covariance matrices are no longer of block diagonal form. The state vector consists of the common center point  $(x_{cc}, y_{cc})$  and the target locations  $(x_j, y_j)$ . The form of Equation (16) is obtained by defining

$$\mathbf{x} = \begin{bmatrix} x_{cc} \\ y_{cc} \\ x_1 \\ y_1 \\ x_2 \\ y_2 \\ \vdots \end{bmatrix} \qquad \mathbf{F}_c = \begin{bmatrix} 0 & 0 & & & & \\ 0 & 0 & & & & \\ \lambda & 0 & -\lambda & 0 & & \\ 0 & \lambda & 0 & -\lambda & & \\ \lambda & 0 & & & -\lambda & 0 \\ 0 & \lambda & & 0 & -\lambda & \\ \vdots & \vdots & & & \ddots \end{bmatrix}$$

$$\mathbf{Q}_{c} = q\mathbf{I} \qquad \mathbf{L}_{c} = \begin{bmatrix} 0 & 0 & & & & \\ 0 & 0 & & & & \\ & 1 & 0 & & & \\ & 0 & 1 & & & \\ & & & 1 & 0 & \\ & & & 0 & 1 & \\ & & & & & \ddots \end{bmatrix}.$$

The corresponding discrete-time transition matrix  $\mathbf{F}_t$  is similar to that in the meanreverting model for independent targets. The discrete-time covariance matrix is not of simple form, and is computed numerically.

#### 5.3 Independent Targets with White Noise Accelerations (I-WNA)

The states of the independent targets are described by their locations  $(x_j, y_j)$  and velocities  $(\dot{x}_j, \dot{y}_j)$ . Accelerations are modelled as white noise processes affecting the velocities of the targets. Following the notation in Equation (16), this model is defined by

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \dot{x}_1 \\ \vdots \\ y_J \\ \dot{y}_J \end{bmatrix} \qquad \mathbf{F}_c = \begin{bmatrix} 0 & 1 & & & \\ 0 & 0 & & & \\ & \ddots & & \\ & 0 & 1 & & \\ & 0 & 0 & & \\ \end{bmatrix}$$
$$\mathbf{Q}_c = q\mathbf{I} \qquad \mathbf{L}_c = \begin{bmatrix} 0 & 0 & & & & \\ 0 & 1 & & & & \\ & 0 & 0 & & & \\ & & 0 & 1 & & \\ & & 0 & 0 & & \\ & & 0 & 1 & \\ & & 0 & 1 & \\ \end{bmatrix},$$

where q is a spectral density parameter, and the matrices  $\mathbf{F}_c$  and  $\mathbf{L}_c$  are block-diagonal matrices. This model can be discretized analytically [Bar-Shalom et al., 2004]. Following the notation in Equation (1), this yields

$$\mathbf{F}_{t} = \begin{bmatrix} 1 & \Delta t & & \\ 0 & 1 & & \\ & \ddots & & \\ & & 1 & \Delta t \\ & & 0 & 1 \end{bmatrix} \qquad \mathbf{Q}_{t} = q \begin{bmatrix} \frac{(\Delta t)^{3}}{3} & \frac{(\Delta t)^{2}}{2} & & \\ \frac{(\Delta t)^{2}}{2} & \Delta t & & \\ & & \ddots & & \\ & & \frac{(\Delta t)^{3}}{3} & \frac{(\Delta t)^{2}}{2} \\ & & \frac{(\Delta t)^{2}}{2} & \Delta t \end{bmatrix}.$$

#### 5.4 Follower Model

In the follower model the observed targets are assumed to approach an unobserved moving target. The dynamics of the unobserved target are modeled with white noise accelerations, as in Section 5.3. The observed targets are mean-reverting targets with the location of the unobserved target,  $(x_{ut}, y_{ut})$ , as the common center.

The velocities of the observed targets are not included in the model. Instead, changes in the velocities of the observed targets are interpreted to be a consequence of the changes in the velocity of the unobserved target. The spectral density of the process noise q is common for all the observed targets, and is in general different from the spectral density of the process noise of the unobserved target,  $q_{ut}$ .

In continuous time, this model is expressed in the form of Equation (16) as

$$\mathbf{x} = \begin{bmatrix} x_{ut} \\ \dot{x}_{ut} \\ y_{ut} \\ \dot{y}_{ut} \\ \dot{y}_{ut} \\ \dot{x}_{1} \\ x_{1} \\ x_{2} \\ y_{2} \\ \vdots \\ x_{J} \\ y_{J} \end{bmatrix}$$

$$\mathbf{F}_{c} = \begin{bmatrix} 0 & 1 & & & & \\ 0 & 0 & & & & \\ 0 & 0 & 1 & & & & \\ 0 & 0 & 1 & & & & \\ 0 & 0 & \lambda & 0 & 0 & -\lambda & & \\ \lambda & 0 & 0 & 0 & -\lambda & & \\ \lambda & 0 & 0 & 0 & -\lambda & & \\ \lambda & 0 & 0 & 0 & -\lambda & & \\ \vdots & \vdots & & & \ddots & \\ \lambda & 0 & 0 & 0 & & & -\lambda & 0 \\ 0 & 0 & \lambda & 0 & & & 0 & -\lambda \end{bmatrix}$$

$$\mathbf{Q}_{c} = \mathbf{I} \qquad \mathbf{L}_{c} = \begin{bmatrix} 0 & 0 & & & & & \\ 0 & \sqrt{q_{ut}} & & & & & \\ & 0 & \sqrt{q_{ut}} & & & & \\ & & 0 & \sqrt{q} & 0 & & \\ & & & 0 & \sqrt{q} & & \\ & & & & 0 & \sqrt{q} & \\ & & & & & \ddots & \\ & & & & & \sqrt{q} & 0 \\ & & & & & 0 & \sqrt{q} \end{bmatrix}$$

The discrete-time matrices are computed numerically.

## 6 Parameter Estimation Using Particle Gibbs

Particle Gibbs [Andrieu et al., 2010] is a method for computing posterior distributions of parameters in a variety of statistical models, where particle filters are applicable. In particular, it can be used in multiple target tracking with the models of target dynamics introduced in Section 5. While the Kalman filter yields the exact posterior distributions

of the states of the targets in linear models with Gaussian noise, it cannot be used, e.g., to estimate multiplicative constants. Particle Gibbs offers a way to estimate such parameters, as well as the spectral density of the process noise. As Particle Gibbs is a Bayesian method, a prior distribution is defined for the parameters to be estimated. In this research project, the main motivation for estimating the parameters is to improve the accuracy of the data associations and the state estimates.

Kokkala and Särkkä [2015] combined Particle Gibbs with RBMCDA to estimate the parameters in the dynamic models of independent targets. The resulting method is based on running RBMCDA sequentially with different parameter values  $\theta^1 \dots \theta^J$ . The parameter values that are a priori likely (according to the prior distribution  $p(\theta)$ ) and fit the data well (as measured by the likelihood  $p(\mathbf{y}|\theta, c)$ ) are likely to be accepted, and the parameter values that are a priori unlikely and fit the data poorly are more likely to be rejected. The data associations of one particle are always inherited from the previous RBMCDA run. This introduces correlation between successive RBMCDA runs and decreases the effect of the random number generation in RBMCDA for any single run.

Denote

- $\theta$ , the parameters to be estimated.
- $\theta^1 \dots \hat{\theta}^J$ , the samples generated from the posterior distribution of the parameters.
- $\theta^*$ , the proposed values of the parameters.
- $p(\theta)$ , the prior distribution of the parameters .
- $q(\theta^*|\theta^{j-1})$ , the proposal distribution. This distribution defines how new parameter values are proposed based on the previous accepted value  $\theta^{j-1}$ .

The initial values of the parameters are chosen or randomly generated, and the following steps are repeated for j = 1...J:

- 1. Propose a new  $\theta^* \sim q(\theta^* | \theta^{j-1})$
- 2. Generate a random number  $r \sim \text{Uniform}([0,1])$  and compute the acceptance probability

$$\alpha = \min\left(\frac{q(\theta^{j-1}|\theta^*)p(\mathbf{y}_{1:T}|\theta^*, c_{1:T}^{j-1})p(\theta^*)}{q(\theta^*|\theta^{j-1})p(\mathbf{y}_{1:T}|\theta^{j-1}, c_{1:T}^{j-1})p(\theta^{j-1})}, 1\right),$$
(21)

where the likelihoods are evaluated using the Kalman filter. If  $r \le \alpha$ , set  $\theta^j = \theta^*$ , otherwise set  $\theta^j = \theta^{j-1}$ .

- 3. Generate a set of N particles  $(c_{1:T}^{j,1:N}, w_{1:T}^{j,(1:N)})$  by running RBMCDA with parameters  $\theta^{j}$ , while keeping the data associations in the first particle as  $c_{1:T}^{j-1}$ .
- 4. Draw new data associations  $c_{1:T}^{j}$  from  $c_{1:T}^{j,1:N}$  with probabilities  $w_T^{j,(1:N)}$ .

This method is applied to tracking of interacting targets by replacing RBMCDA in Step 3 with the extension described in Section 3.

The length J of the sequence (or *chain*) of parameter values is chosen sufficiently large, so that the effect of the initial value is negligible. The method can be run several times to produce different chains, which can then be compared to each other to assess whether the chains have converged to the same distribution.

# 7 Optimal Subpattern Assignment for Tracks

In this section, the Optimal Subpattern Assignment for Tracks (OSPA-T) [Ristic et al., 2010], a performance measure for multiple target tracking methods, is reviewed. While a large number of measures could be used for performance evaluation (for an overview, see [Gorji et al., 2011]), using a single performance measure allows for more easily interpretable numerical results.

OSPA-T takes into account the accuracy of the state estimates and the estimate of the number of targets. In addition, it considers the state estimates as a whole by adding a penalty if an estimated target corresponds to different true targets at different times.

The true track of a target is defined as the set of locations at different time steps. Similarly, the estimated tracks are sets of estimated locations of the targets at different time steps. OSPA-T assigns the estimated tracks to the true tracks so that they are as close to each other as possible. The differences between the true and estimated tracks are then measured using the OSPA metric [Schuhmacher et al., 2008]. Missing and false tracks increase the OSPA-T metric, whereas accurate state estimates decrease it.

Denote

- $\mathbf{x}_{t}^{l}$ , the true location of the target in track l at measurement t.
- *L*, the total true number of targets.
- $L_t$ , the true number of targets at measurement t.
- $\mathbf{z}_{t}^{l}$ , the estimated location of target *l* at time *t*.
- *R*, the total estimated number of targets.
- $R_t$ , the estimated number of targets at measurement t.
- $\Pi_L^R$ , the set of permutations of length L with elements taken from  $\{1 \dots R\}$ .
- ||||, the Euclidean norm.
- $e_l^t$ , is 1, if the target *l* exists at time *t*, otherwise zero.
- $\alpha$ , the penalty resulting from assigning the target to a wrong track.
- $\Delta$ , the penalty resulting from a missing or a false track in the track assignment step (step 1 in the following).
- $\beta$ , the penalty resulting from a missing or a false track in the OSPA step (step 2). Generally  $\alpha < \beta$ , meaning that it is preferred to assign a target to a wrong track than miss the target.
- p and p', parameters that describe the sensitivity of the OSPA-T value with respect to outliers, i.e., how much OSPA-T is affected by a small number of state estimates with large errors.

The OSPA-T metric is computed in two steps as follows:

1. Find  $\pi^* \in \Pi_L^R$ , the optimal track assignment of the estimated tracks to the true tracks. Here  $\pi^*(i) = j$  means that the estimated track *l* is assigned to the target *j*. If  $L \leq R$ ,

$$\pi^* = \arg \min_{\pi \in \Pi_L^R} \sum_{l=1}^L \sum_{t=1}^T \left( e_t^{\pi(l)} e_t^l \min(\|x_t^l - z_t^{\pi(l)}\|, \Delta) + (1 - e_t^{\pi(l)}) e_t^l \Delta + e_t^{\pi(l)} (1 - e_t^l) \Delta \right).$$
(22)

If L > R, the roles of the true tracks and the estimated tracks are reversed. When the number of targets is low, the minimization problem can be solved by exhaustive search through all the permutations in  $\Pi_{L}^{R}$ .

2. The OSPA-T metric is given by the OSPA metric for  $\pi^*$ , at each measurement t = 1...T, computed as follows. Define  $\tilde{z}_t^{(i)} = z_t^{\pi^*(i)}$ . If  $L_t \leq R_t$ ,

$$OSPA-T(t) = \left(\frac{1}{R_t} \min_{\pi \in \Pi_{L_t}^{R_t}} \left( (R_t - L_t)\beta^p + \sum_{i=1}^{L_t} \min(d(x_t^i, \tilde{z}_t^{\pi(i)}), \beta)^p \right) \right)^{\frac{1}{p}}, \quad (23)$$

where

$$d(x,\tilde{z}) = \begin{cases} \|x - \tilde{z}\|, & \text{if } i = \pi(i) \\ (\|x - \tilde{z}\|^{p'} + \alpha^{p'})^{\frac{1}{p'}} & \text{otherwise.} \end{cases}$$
(24)

If  $L_t > R_t$ , the roles of the true tracks and the estimated tracks are reversed.

OSPA-T does not directly penalize for large variances in the state estimates. This means that in the sense of OSPA-T, posterior uncertainty of the state estimates does not matter at all. Indirectly, large variances may affect the data associations and consequently increase the OSPA-T value. As OSPA-T is computed separately for each time step, it does not directly offer a way to summarize the results with a single number. One may plot the OSPA-T values over time for visual comparisons, or summarize the values with a time-average, as was done with OSPA in [Septier et al., 2011]. The approach taken in this project is to use both visual comparisons and time-averages.

### 8 Simulations

In this section, the models described in Section 5 are compared using simulated data. RBMCDA is used for the models with independent targets, and the extension of RBMCDA described in Section 3 is used for the models with interactions. The models are compared using plots of average OSPA-T values over the simulations, time-averaged OSPA-T values, posterior distributions of the number of targets, and computation times.

#### 8.1 Mean-Reverting Targets with a Common Center in Clutter

This simulation scenario contains five targets approaching a common center point. The two dynamic models used for the targets are the independent targets model with mean-reversion (I-MR), described in Section 5.1, and the common center model, described in Section 5.2. The only difference between these models is that in the former model the targets are allowed to approach different points, whereas in the latter model the targets approach the same point.

The true locations of the targets are generated from Equation (20) with parameters  $\lambda = 0.07$  and  $\sqrt{q} = 9$ . At constant time intervals  $\Delta t = 0.3$ , one measurement is obtained from each target. In addition, 10 clutter measurements, uniformly distributed over the square  $(0,0) \times (500,500)$ , are obtained at each time step. Thus, only 1/3 of the measurements represent true targets. The measurement errors are normally distributed with a standard deviation of 4 in each coordinate. The prior distributions and parameter values used for tracking are the following:

- The prior distribution for the target locations is the normal distribution with zero mean and variance  $10^7$  in each coordinate. This represents vague prior information: about 97.7% of the probability mass is contained within the square  $(-8000, -8000) \times (8000, 8000)$
- The prior distribution for the centers in both models is the normal distribution with zero mean and variance  $10^6$  in each coordinate.
- The prior distribution for the data associations is constructed as follows:
  - Each already seen target is a priori equally likely to be assigned to a new measurement.
  - The prior probability of clutter is 0.7. This represents a good estimate of the true proportion of the clutter measurements.
  - The prior probability of assigning a measurement to a new target is 0.06.
- The parameter values  $\lambda$  and  $\sqrt{q}$  in the dynamic models are set equal to the true values used to generate the data.
- The prior density of clutter is set to the constant value  $1450^{-2}$  everywhere.
- The number of particles *N* is 1000.

The following example illustrates the differences between the models. Figure 1(a) shows one realization of the true locations of the targets. The measurements from the targets and the clutter are shown in Figure 1(b).

The data associations obtained using the two models are shown in Figure 2. These data associations are the ones with the largest weight. The model with independent targets yields a false target near the point (150, 150). This type of mistake would be unlikely when using the model with a common center, as this false target moves away from the other targets. Some delay in detecting the targets is seen with both models, i.e., the first measurements from the targets are treated as clutter.



Figure 1: a) The true target locations generated from the model. b) The generated measurements with additive measurement noise.



Figure 2: The data associations obtained using a) independent targets (I-MR) and b) a common center for all targets. Each measurement assigned to a target is marked with a  $\circ$ . The colours are assigned according to the order in which the targets are detected. The measurements associated with clutter are marked with  $\times$ .

Figure 3 shows the estimates of the target locations obtained using Kalman filtering and the previous data associations. As these estimates are obtained by conditioning on the previous measurements, they represent a possible result if the targets were tracked in real time. The number of estimates is greater than the number of measurements, as the estimates are also updated when no new measurements are associated with the targets. When using the common center model, each measurement associated to a target affects the estimate of the center, which further affects the estimates of all the targets.

The false target in 3(a) is estimated to approach the location (558.2, -1182.9). The estimated center in the case shown in Figure 3(b) is (251.8, 273.6), not far from the true center at (250, 250).



Figure 3: Estimates of the locations of the targets obtained with Kalman filtering using the model with a) independent targets (I-MR) and b) a common center for all targets.

Finally, the smoothed estimates are shown in Figure 4. Only slight corrections to the estimates are made by the smoother, as the measurement errors are small and the estimates given by the Kalman filter are already quite consistent with the dynamic model.



Figure 4: Estimates of the locations of the targets obtained with a Rauch-Tung-Striebel smoother using the model with a) independent targets (I-MR) and b) with a common center for all targets.



Figure 5: OSPA-T values of the smoothed estimates in the example case. Each time step corresponds to 5 target measurements and 10 clutter measurements.

Figure 5 shows the OSPA-T values for the smoothed estimates in the example case with parameters  $\beta = \Delta = 10$ ,  $\alpha = 5$ , p = 1, and p' = 1. The common center model has a slightly larger OSPA-T value for the first two time steps. This seems to be caused by the delay in detecting the targets. The independent targets model is worse according

to this metric for most of the time steps. The OSPA-T values are heavily correlated between the models, suggesting that the measurement errors and the clutter affect the results similarly with both models.

Figure 6 shows the OSPA-T values averaged over 40 Monte Carlo simulations. Both the true target locations and the measurement noise are randomized separately for each simulation, but the same data is used with both models. The OSPA-T values are computed for the smoothed estimates using the data associations with the largest weight.

It is seen that the models give similar results in the first 7 time steps. After this, the common center model is superior. An increase in OSPA-T is seen at the last time steps. This can be explained by that there is not enough data to rule out target births at the last time steps. This problem could be addressed by modifying the target birth and clutter prior probabilities. However, this might result in larger delays in target detection and larger OSPA-T values in the first time steps. Another cause for the increase in the OSPA-T value is that there are no future measurements to use for smoothing the last state estimates.



Figure 6: OSPA-T values of the smoothed estimated averaged over 40 Monte Carlo simulations using the two dynamic models.

The Monte Carlo simulations are summarized in Table 1. The average OSPA-T is the time-average of the values shown in Figure 6. The computation time is longer when using the independent targets model, compared to the common center model. This is most likely related to that the independent targets model tends to overestimate the number of targets, which results in larger numbers of possible data associations and more unique particles to update.

Model	Time-averaged OSPA-T	Computation time (s)
Common center	6.3497	450.4
Independent targets (I-MR)	6.8491	714.3

Table 1: A summary of the Monte Carlo simulations in the first simulation scenario.

The computation times for the I-WNA model could be reduced by requiring more evidence before a measurement is assigned to a new target. This could be done by changing the clutter prior or the birth prior probabilities. However, as noted, this might result in larger delays in target detection and larger OSPA-T values in the first measurements.

Figure 7 shows the posterior distributions of the number of targets with the two models. The number of particles is 10000 for each model. As the OSPA-T values in Figure 6 suggest that the number of targets is overestimated at the last time steps, the number of targets is estimated at time step 20 using the particles obtained by computing the data associations for the whole data. It is seen that the number of targets is overestimated when using the I-MR model, and even the number of targets 9 is given a positive probability. With the common center model, all the posterior mass is given to numbers 5 and 6. The number of targets is not underestimated with either model.



Figure 7: Posterior distributions of the number of targets at time step 20 using a) the I-MR model and b) the common center model.

#### 8.2 Group Tracking in Clutter

In the second simulation scenario a group of five targets moves in a V-formation, making two turns of 180 degrees. One measurement from each target and 10 clutter measurements are received at constant time intervals  $\Delta t = 1$ . Figure 8(a) shows the true target locations, starting from the lower left corner at t = 0, and Figure 8(b) shows an example of a simulated data set. The measurements are random, whereas the true target locations are not. The standard deviation of the measurement noise is 6.



Figure 8: a) The true target locations and b) measurements with additive measurement noise.

The I-WNA model (Section 5.3), and the follower model (Section 5.4) are used to track the targets. The following choices for the parameters and the prior distributions are used:

- The prior variance for the target locations is  $10^6$  in each coordinate.
- The prior variances for the location and velocity of the unobserved target in the follower model are  $10^5$  and  $10^3$  in each coordinate, respectively.
- The prior standard deviations of the target velocities in the I-WNA model are 50 in each coordinate. The true velocities in this scenario vary roughly between 50 and 100.
- The target birth prior is that used in [Kokkala and Särkkä, 2015], with the difference that clutter measurements are taken into account. The prior probability of clutter is 0.7. In this prior distribution the probability of a target birth decreases as new measurements are assigned to targets.

- The square roots of the process noise spectral densities of the observed targets are 15 (follower model) and 35 (I-WNA). The spectral density parameter for the unobserved target in the follower model is 90.
- The parameter value  $\lambda$  is 0.25.
- The prior density of clutter,  $p(\mathbf{y}_k | c_k = 0)$ , is set to the constant value 1750<sup>-2</sup> everywhere.
- The number of particles N is 1000.

Figure 9 shows the data associations from a simulation where both models yield good results. Figure 9(a) shows that when using the I-WNA model, two observations that lie far from the group are associated with a target in the last time step. This is consistent with the model, as the targets are not forced to be close to each other. According to the data associations, the targets switch places with each other. For example, the red target is in the middle at the beginning of the tracking, and at the end it is in the leftmost position. This phenomenon is far less pronounced with the follower model in Figure 9(b).



Figure 9: The data associations obtained using the two models.

Figures 10 and 11 show the filtered and smoothed estimates of the targets, respectively. The differences between the filtered and the smoothed estimates are small. This is an expected result, considering that the measurement errors are small. The largest differences are seen at the beginning of the tracking with the follower model, whereas with the I-WNA model the largest differences are seen about halfway into the tracking.



Figure 10: Estimates of the locations of the targets obtained with Kalman filtering.



Figure 11: Estimates of the locations of the targets obtained with a Rauch-Tung-Striebel smoother.

The estimated locations of the unobserved target in the follower model are shown in Figure 12. When the targets change directions, an overshooting phenomenon is seen, i.e., the turns made by the unobserved target are greater than 180 degrees. This is not an error, because the unobserved target is only used in the model to introduce correlations between the observed targets, and consequently, no ground truth exists for these locations.



Figure 12: Smoothed estimates of the unobserved target in the follower model. True locations of the targets are drawn for reference.

Figure 13 shows the prior and posterior distributions of the square roots of the spectral densities in the I-WNA model and the follower model, obtained using the Particle Gibbs algorithm described in Section 6. The data used for parameter estimation is that in the example case shown in Figure 8(b). The number of chains used in Particle Gibbs is 20, and the length of each chain is 1000. The first 500 samples of the chains are discarded as warm-up samples to decrease the effect of the initial values of the parameters. The number of particles is 1000 for the first sample of each chain, and 5 for the remaining parts. The prior distributions are Gamma distributions with scale parameter 2 and means 35 and 90, respectively. Point estimates for the parameters are obtained as posterior means (38.4 and 73.4).

The OSPA-T values averaged over 40 Monte Carlo simulations are shown in Figure 14. The measurements are generated separately for each simulation. The models are first used before parameter estimation, and then using the point estimates obtained above. The parameters of OSPA-T are  $\beta = \Delta = 200$ ,  $\alpha = 50$ , p = 1, and p' = 1. These values are larger than in the first simulation scenario in Section 8.1, because the errors in the target locations are expected to be higher due to mislabeling of the targets, as seen in Figure 9(a). Since these parameters are different, these OSPA-T values are not comparable to those in the first simulation scenario.

It is seen that the follower model performs better than the I-WNA model according to the OSPA-T values. An improvement is seen in the performance of both models when the estimated parameter values are used. The differences between the models are small in the first time steps, and the effect of parameter estimation seems to vanish in the last steps for the I-WNA model. Unlike in the first simulation scenario, no significant increase is seen at the last time steps. This can be attributed to the different target birth prior distribution, which was chosen so that new targets are unlikely to be born after the first time steps.



Figure 13: Histograms of the posterior distributions of the parameters. The prior distributions are shown as solid lines.



Figure 14: OSPA-T values of the smoothed estimates averaged over 40 Monte Carlo simulations using the I-WNA model and the follower model with and without parameter estimation (PE).

The Monte Carlo simulations are summarized in Table 2. The computation times are sensitive to the parameter values. Using the estimated parameters decreases the computation time with the follower model, whereas the opposite happens with the I-WNA model. This can be attributed to that without parameter estimation, the number of targets was often overestimated with the follower model, and underestimated with the I-WNA model. When the estimated parameter values are used, the follower model is faster. The computation times are longer than in the first simulation scenario. This is natural, as the measurements are close to each other, which means that a large number of plausible data association histories can be found. This results in a large number of unique particles.

Model	Time-averaged OSPA-T	Computation time (s)
Follower model	73.9578	1593.4
Follower model (PE)	65.5909	1138.8
I-WNA	88.6114	1226.6
I-WNA (PE)	82.9237	1448.8

Table 2: A summary of the Monte Carlo simulations in the second simulation scenario. (PE) denotes parameter estimation.

Figure 15 shows the posterior distributions of the number of targets using the I-WNA model and the follower model with the estimated parameters. According to preliminary simulations, a large number of particles would be needed to compute the posterior distributions with the whole data to avoid large variation between different runs. For this reason, the posterior distributions were computed using 100000 particles and the data from the first 15 time steps of the example case in Figure 8(b). Most of the posterior mass is given to the correct number of targets with both models. The posterior variance is smaller when using the follower model. Very small, but non-zero probabilities are given to numbers 3 and 7 with the I-WNA model, and 6 and 7 with the follower model. The unobserved target in the follower model is not counted in the number of targets.

According to the Potential Scale Reduction Factor [Gelman and Rubin, 1992], the chains in Particle Gibbs had not converged to the same distribution (PSRF > 1.2), which indicates that the samples obtained may not be representative of the true posterior distributions of the parameters. For this reason, the way Particle Gibbs is used in this experiment is somewhat heuristic. The nonconvergence is likely due to the fact that early data associations were unlikely to change within the chains. Similar problems have been reported in the literature, also outside the context of multiple target tracking, and addressed by modifying the sampling procedure in various ways (see, e.g., [Lindsten et al., 2014], [Andrieu et al., 2010], [Kokkala and Särkkä, 2015]). However, these modifications are outside the scope of this research project.



Figure 15: Posterior distributions of the number of targets using the data from the first 15 time steps with a) the I-WNA model and b) the follower model.

It should be noted that some of the parameters in the follower model were fixed. Thus, it seems likely that the performance of the follower model could be further improved by estimating all the parameters with Particle Gibbs. The I-WNA model does not contain other parameters than the spectral density of the process noise, which was estimated.

## **9** Conclusions

An extension to the Rao-Blackwellized Monte Carlo Data Association (RBMCDA) method was presented to track interacting targets. Two different interaction models were developed: one for targets approaching a common center and one for targets moving in groups. The models were applied to simulated data and compared to models without interactions using the OSPA-T metric, computation times and posterior distributions of the number of targets. Parameter estimation was carried out by applying the Particle Gibbs method.

Two simulation scenarios were used, and 40 data sets were generated in each scenario to obtain reliable estimates of the OSPA-T values and computation times. The simulations show that the extension of RBMCDA with interaction models yields good results: the OSPA-T values obtained with the interaction models are lower than those obtained with models without interactions. In the first simulation scenario, the comparison between the models was justified by that the same parameter values were used in both models. The set-up in the second simulation scenario can be considered to favour the model without interactions, since not all parameters of the interaction model

were estimated. Despite this, the interaction model performed clearly better. However, it should be noted that the performance of the models was found to be sensitive with respect to the parameter values used.

Modeling the interactions decreased the computation times, despite that the interaction models contain larger state vectors, and consequently require computations with larger matrices. In the second simulation scenario the decrease in computation times was seen after the parameters were estimated. The differences in the computation times can be partially attributed to that using models without interactions tends to lead to greater uncertainty about the number of targets. This was seen in the posterior distributions of the number of targets, which were concentrated closer to the true value when the interactions were modeled. The increase in the uncertainty naturally leads to larger numbers of unique particles, increasing the computation times. Furthermore, computations with larger numbers of targets are naturally heavier.

The chains in Particle Gibbs seemed not to have converged to the same distribution, which means that the samples obtained may not be representative of the true posterior distributions of the parameters. Improvements to this method in the context of multiple target tracking are left as a subject for future research. When computational expense is not a limiting factor, longer chains could be used with more particles to improve convergence. Despite the convergence issues, parameter estimation improved the performance of the two models it was applied to, according to the OSPA-T metric.

The approach to modeling interactions could be extended in several directions. Nonlinear dynamic models could be used in the RBMCDA framework by replacing the Kalman filter with, e.g., Extended Kalman filter (EKF) or Unscented Kalman filter (UKF) [Särkkä et al., 2007]. This may open new possibilities for developing target interaction models.

In practical applications, e.g., tracking of people and animal populations, it might be beneficial to consider multiple different types of interactions between the targets. Multiple groups of targets may be present, and the groups may further interact with each other. Further research could be carried out to estimate the number of groups, and to associate each target with one of them. Pang et al. [2011] suggest that the difficulties in implementing such methods has hindered progress in the area of group tracking. However, it seems plausible that such group assignments could be included in the RBMCDA framework, once a suitable distribution for the data associations is defined.

# References

- Abbas, M. Statistical estimation of wild animal population in Finland: a multiple target tracking approach. Master's thesis, Aalto University, 2011.
- Andrieu, C., Doucet, A., and Holenstein, R. Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B*, 72(3):269–342, 2010.
- Bar-Shalom, Y., Li, X., and Kirubarajan, T. *Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software*. Wiley, 2004.
- Blackman, S. S. *Multiple-target tracking with radar applications*. Artech House Radar Library. Artech House, 1986.
- Casella, G. and Robert, C. Rao-Blackwellisation of sampling schemes. *Biometrika*, 83 (1):81–94, 1996.
- Cham, T. and Rehg, J. A multiple hypothesis approach to figure tracking. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition.*, 239–245, 1999.
- Doucet, A., Godsill, S., and Andrieu, C. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and computing*, 10(3):197–208, 2000.
- Fortmann, T., Bar-Shalom, Y., and Scheffe, M. Multi-target tracking using joint probabilistic data association. In 19th IEEE Conference on Decision and Control, 807–812, 1980.
- Gelman, A. and Rubin, D. B. Inference from iterative simulation using multiple sequences. *Statistical science*, 7(4):457–472, 1992.
- Gorji, A., Tharmarasa, R., and Kirubarajan, T. Performance measures for multiple target tracking problems. In *Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on*, 1–8, 2011.
- Grewal, M. and Andrews, A. *Kalman filtering: theory and practice using MATLAB*, *Third edition*. Wiley InterScience Online Books, Electronic and Electrical Engineering Collection. Wiley, 2008.
- Kalman, R. A new approach to linear filtering and prediction problems. *Transactions* of the ASME–Journal of Basic Engineering, 82(Series D):35–45, 1960.
- Khan, Z., Balch, T., and Dellaert, F. An MCMC-based particle filter for tracking multiple interacting targets. In *Computer Vision-ECCV 2004*, 279–290. Springer, 2004.

- Kokkala, J. and Särkkä, S. Combining particle MCMC with Rao-Blackwellized Monte Carlo data association for parameter estimation in multiple target tracking. *In press, Digital Signal Processing*, 2015.
- Lane, D., Chantler, M., and D., D. Robust tracking of multiple objects in sector-scan sonar image sequences using optical flow motion estimation. *IEEE Journal of Oceanic Engineering*, 23(1):31–46, 1998.
- Lindsten, F., Jordan, M. I., and Schön, T. B. Particle Gibbs with ancestor sampling. *The Journal of Machine Learning Research*, 15(1):2145–2184, 2014.
- Mahler, R. Multitarget Bayes filtering via first-order multitarget moments. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1152–1178, 2003.
- Pang, S., Li, J., and Godsill, S. Detection and tracking of coordinated groups. *IEEE Transactions on Aerospace and Electronic Systems*, 47(1):472–502, 2011.
- Papoulis, A. Probability, Random Variables, and Stochastic Processes. Mc-Graw Hill, 1984.
- Rauch, H. E., Striebel, C. T., and Tung, F. Maximum likelihood estimates of linear dynamic systems. *Journal of the American Institute of Aeronautics and Astronautics*, 3(8):1445–1450, 1965.
- Ristic, B., Vo, B.-N., and Clark, D. Performance evaluation of multi-target tracking using the OSPA metric. In *Information Fusion (FUSION), 2010 13th Conference on*, 1–7, 2010.
- Schuhmacher, D., Vo, B.-T., and Vo, B.-N. A consistent metric for performance evaluation of multi-object filters. *IEEE Transactions on Signal Processing*, 56(8): 3447–3457, 2008.
- Septier, F., Cornebise, J., Godsill, S., and Delignon, Y. A comparative study of Monte-Carlo methods for multitarget tracking. In *IEEE Statistical Signal Processing Workshop (SSP)*, 205–208, 2011.
- Särkkä, S. *Bayesian Filtering and Smoothing*. Institute of Mathematical Statistics Textbooks. Cambridge University Press, 2013.
- Särkkä, S., Vehtari, A., and Lampinen, J. Rao-Blackwellized Monte Carlo data association for multiple target tracking. In *Proceedings of the Seventh International Conference on Information Fusion*, 583–590, 2004.
- Särkkä, S., Vehtari, A., and Lampinen, J. Rao-Blackwellized particle filter for multiple target tracking. *Information Fusion*, 8(1):2–15, 2007.