

Systematic approach to IMM mixing for unequal dimension states

Karl Granström, *Member, IEEE*, Peter Willett, *Fellow, IEEE*, and Yaakov Bar-Shalom, *Fellow, IEEE*

Abstract—The IMM estimator outperforms fixed model filters, e.g. the Kalman filter, in scenarios where the targets have periods of disparate behavior. Key to the good performance and low complexity is the mode mixing. Here we propose a systematic approach to mode mixing when the modes have states of different dimensions. The proposed approach is general and encompasses previously suggested solutions. Different mixing approaches are compared, and the proposed methodology is shown to perform very well.

Index Terms—Multiple mode, Interacting Multiple Model, IMM, Jump Markov System, estimator, filter.

I. INTRODUCTION

In many cases it is insufficient to model an estimation problem using a single process model and/or a single measurement model. Multiple models are necessary, e.g. to model motion where one process model is accurate for straight line motion and another model is accurate for curving motion. Each process and measurement model pair is referred to as a mode, and the switches between the modes are modeled as a Markov process.

When the mode switches are unknown, as is the typical case, the complexity of the optimal Bayesian solution to the problem increases exponentially and approximations are necessary. Examples of approximate methods include the first and second order Generalized Pseudo Bayesian (GPB) estimators, see e.g. [3], and the interacting multiple model (IMM) estimator [5].

The IMM filter is known to represent a good compromise between estimation accuracy and computational complexity. The algorithm consists of three main steps:

- 1) mode mixing;
- 2) mode matched prediction update;
- 3) mode matched measurement update.

Typically each mode is represented by a Gaussian estimate and the models are (non)-linear with Gaussian noise, in which case the prediction and measurement updates can be taken from the Kalman filter or one of its non-linear variants.

The key to the remarkable success of the IMM estimator is that it carries out the mixing just before the update of the mode-conditioned estimates, just like the optimal Bayesian estimator [2]. The key to the simplicity of the IMM estimator is the Gaussian mixture approximation, with the number of terms equal to the number of modes, of the exact prediction probability density function (pdf). The exact prediction pdf, in

the case of the optimal Bayesian estimator, has an exponentially increasing number of terms.

In this paper we will consider the first step, mode mixing, for modes with states of unequal dimension. For the sake of simplicity we will assume that there are two modes, however it is straightforward to generalize the presented work to an arbitrary number of modes. Mode mixing involves matching the first two moments, i.e. the Gaussian mean and covariance, however this cannot be performed in the standard way when the state vectors are of unequal dimension. This problem is approached by augmenting the distribution for the smaller state such that the augmented dimension matches the larger state.

A simple approach, see e.g. [3], involves augmenting with zero mean and variance. This leads to a biased estimate [14], and as a remedy one can augment with the mean and variance from the larger state. This approach showed good results when tested with real data from thrusting ballistic projectiles in a scheme involving multiple IMM estimators [14]. A similar approach was used in [8], however the implementation in [8] is performed state-by-state, in contrast to the implementation in [14] (and the standard IMM estimator) which is performed directly on the state vectors.

The approach suggested in this paper is based on the idea that the augmented distribution should reflect the underlying situation when the target switches from one mode to the other, without negatively affecting the estimation of the states that are common to both modes. For example, if the target switches from CV motion to turning motion with some turn rate, then the distribution for the turn rate should correctly reflect the feasible turn rate values.

The paper is organized as follows. The next section presents a problem formulation, followed by the proposed approach to mixing in Section III. Three different motion model pairs are used in a simulation comparison of four different approaches to mode mixing. The simulation setup is presented in Section IV, and the simulation results are given in Section V. The paper is concluded in Section VI.

II. PROBLEM FORMULATION

In this section we briefly review mode mixing, and give a description of the problem considered in the paper.

A. Mode mixing

Let m_k denote the target mode at time step k , and let \mathbf{Z}^k denote all measurements up to and including time step k . For each of the two modes there is a corresponding Gaussian state estimate,

$$p(\mathbf{x}_k | m_k = i, \mathbf{Z}^k) = \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}^{(i)}, P_{k|k}^{(i)}), \quad i \in \{1, 2\} \quad (1)$$

Karl Granström, Yaakov Bar-Shalom and Peter Willett are with the Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT, USA. E-mail: {karl, ybs, willett}@engr.uconn.edu.

Supported by the Naval Postgraduate School, via ONR N00244-14-1-0033, and by ONR directly via N00014-13-1-0231; Y. Bar-Shalom is also supported by ARO W991NF-10-1-0369.

where $\hat{\mathbf{x}}_{k|k}^{(i)}$ and $P_{k|k}^{(i)}$ are the mean and covariance, respectively, and $\hat{\mathbf{x}}_{k|k}^{(i)} \in \mathbb{R}^{n_x^i}$. The backward mixing probabilities

$$\mu_{k-1|k}^{(i,j)} = P(m_{k-1} = i | m_k = j, \mathbf{Z}^{k-1}), \quad i, j \in \{1, 2\} \quad (2)$$

model the probability that the mode was i at time step $k-1$, conditioned on being in mode j at time k . In the mixing step the following Gaussian estimates are computed for $j \in \{1, 2\}$,

$$\hat{\mathbf{x}}^{(0,j)} = \mu^{(1,j)} \hat{\mathbf{x}}^{(1)} + \mu^{(2,j)} \hat{\mathbf{x}}^{(2)} \quad (3a)$$

$$P^{(0,j)} = \mu^{(1,j)} P^{(1)} + \mu^{(2,j)} P^{(2)} + M^{(0,j)} \quad (3b)$$

$$M^{(0,j)} = \mu^{(1,j)} \left(\hat{\mathbf{x}}^{(0,j)} - \hat{\mathbf{x}}^{(1)} \right) \left(\hat{\mathbf{x}}^{(0,j)} - \hat{\mathbf{x}}^{(1)} \right)^T + \mu^{(2,j)} \left(\hat{\mathbf{x}}^{(0,j)} - \hat{\mathbf{x}}^{(2)} \right) \left(\hat{\mathbf{x}}^{(0,j)} - \hat{\mathbf{x}}^{(2)} \right)^T \quad (3c)$$

where, for the sake of brevity, we have dropped the time indexing.

The mode mixing (3) can be derived and motivated with the Kullback Leibler divergence (KL-div) [7]. Defined for two pdfs $p(x)$ and $q(x)$ as

$$\text{KL}(p(x)||q(x)) = \int p(x) \log(p(x)/q(x)) dx \quad (4)$$

the KL-div is a measure of the information lost when the distribution $q(x)$ is used to approximate $p(x)$, see e.g. [4]. When it comes to approximating distributions in a maximum likelihood sense, the KL-div is often considered the optimal difference measure, see e.g. [1], [6], [11]–[13].

The mixing (3) corresponds to the approximation of a weighted sum of Gaussian distributions,

$$\mu^{(1,j)} \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}^{(1)}, P^{(1)}) + \mu^{(2,j)} \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}^{(2)}, P^{(2)}) \quad (5a)$$

with a single Gaussian distribution,

$$\mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}, P) \quad (5b)$$

By minimizing the Kullback Leibler divergence between the weighted sum of Gaussian distributions (5a) and the single Gaussian distribution (5b) it can be shown that the optimal parameter choice $\hat{\mathbf{x}}$ and P are indeed given by the mixed parameters (3). In a maximum likelihood sense, the mean $\hat{\mathbf{x}}^{(0,j)}$ and covariance $P^{(0,j)}$ given by (3) are, given the mixing probabilities $\mu^{(1,j)}$ and $\mu^{(2,j)}$, the best approximations of the two estimates $\hat{\mathbf{x}}^{(1)}$, $P^{(1)}$ and $\hat{\mathbf{x}}^{(2)}$, $P^{(2)}$.

The KL-div is known for its moment matching characteristics for distributions that belong to the exponential family, see e.g. [1], [4], [9]; the mode mixing (3) corresponds to moment matching of the first and second order moments.

B. Unequal state dimensions

Let mode 1 have higher dimension than mode 2, i.e. $n_x^1 > n_x^2$, and assume that the states contained in $\hat{\mathbf{x}}_{k|k}^{(2)}$ are also contained in $\hat{\mathbf{x}}_{k|k}^{(1)}$,

$$\hat{\mathbf{x}}_{k|k}^{(1)} = \begin{bmatrix} \hat{x}_{k|k}^{(1)} \\ \hat{\theta}_{k|k}^{(1)} \end{bmatrix}; \quad P_{k|k}^{(1)} = \begin{bmatrix} P_{k|k}^{(xx,1)} & P_{k|k}^{(x\theta,1)} \\ P_{k|k}^{(\theta x,1)} & P_{k|k}^{(\theta\theta,1)} \end{bmatrix} \quad (6a)$$

$$\hat{\mathbf{x}}_{k|k}^{(2)} = \hat{x}_{k|k}^{(2)}; \quad P_{k|k}^{(2)} = P_{k|k}^{(xx,2)} \quad (6b)$$

The additional state $\hat{\theta}_{k|k}^{(1)} \in \mathbb{R}^{n_1-n_2}$ can, e.g., be model acceleration, turn-rate, steering angle, or thrust.

Since $n_x^1 \neq n_x^2$ the mixing cannot be performed as in (3), because vectors and matrices with unequal dimension cannot be added and multiplied. For mixing involving transition to mode 2 ($\hat{\mathbf{x}}^{(0,2)}$ and $P^{(0,2)}$) this has a simple and straightforward solution: replace $\hat{\mathbf{x}}_{k|k}^{(1)}$ and $P_{k|k}^{(1)}$ with the marginalized mean and covariance, i.e. replace with $\hat{x}_{k|k}^{(1)}$ and $P_{k|k}^{(xx,1)}$.

However, mixing involving transition to mode 1 ($\hat{\mathbf{x}}^{(0,1)}$ and $P^{(0,1)}$) require more elaborate treatment; this is the problem considered in this paper. We will propose a simple and general approach to the problem, show how it encompasses previously suggested solutions, and present results from a simulation study that compares the different approaches for different state vectors.

III. MIXING WITH UNEQUAL STATE DIMENSIONS

In this section, for the sake of brevity, time indexing is omitted. For the case of mixing into mode 1, augment the mode 2 distribution with an arbitrary distribution $q(\theta^{(2)})$,

$$p(x, \theta | m = 2, \mathbf{Z}) = \mathcal{N}(x; \hat{x}^{(2)}, P^{(xx,2)}) q(\theta^{(2)}) \quad (7)$$

Next, analogous to (5), we approximate the sum

$$\mu^{(1,j)} \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}^{(1)}, P^{(1)}) + \mu^{(2,j)} p(x, \theta | m = 2, \mathbf{Z}) \quad (8)$$

with a single Gaussian distribution. By matching the first and second order moments (i.e. minimizing KL-div) the optimal Gaussian parameters can be shown to be

$$\hat{\mathbf{x}}^{(0,1)} = \mu^{(1,1)} \hat{\mathbf{x}}^{(1)} + \mu^{(2,1)} \hat{\mathbf{x}}^{(2,a)} \quad (9a)$$

$$P^{(0,1)} = \mu^{(1,1)} P^{(1)} + \mu^{(2,1)} P^{(2,a)} + M^{(0,1)} \quad (9b)$$

$$M^{(0,1)} = \mu^{(1,1)} \left(\hat{\mathbf{x}}^{(0,1)} - \hat{\mathbf{x}}^{(1)} \right) \left(\hat{\mathbf{x}}^{(0,1)} - \hat{\mathbf{x}}^{(1)} \right)^T + \mu^{(2,1)} \left(\hat{\mathbf{x}}^{(0,1)} - \hat{\mathbf{x}}^{(2,a)} \right) \left(\hat{\mathbf{x}}^{(0,1)} - \hat{\mathbf{x}}^{(2,a)} \right)^T \quad (9c)$$

where the augmented mean $\hat{\mathbf{x}}^{(2,a)}$ and covariance $P^{(2,a)}$ are

$$\hat{\mathbf{x}}^{(2,a)} = \begin{bmatrix} \hat{x}^{(2)} \\ E_q[\theta] \end{bmatrix}; \quad P^{(2,a)} = \begin{bmatrix} P^{(xx,2)} & \mathbf{0} \\ \mathbf{0} & \text{Cov}_q(\theta) \end{bmatrix} \quad (10)$$

Here the expected value and covariance of θ are computed with respect to the distribution q .

The mixing approach (9), (10) is valid for any distribution $q(\cdot)$ for which the first and second moments can be computed, and it encompasses the two following standard approaches to mixing of states of unequal dimension.

- A simple approach, see e.g. [3], is to augment the mode 2 estimate as

$$\hat{\mathbf{x}}^{(2,a)} = \begin{bmatrix} \hat{x}^{(2)} \\ \mathbf{0} \end{bmatrix}; \quad P^{(2,a)} = \begin{bmatrix} P^{(xx,2)} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad (11)$$

This corresponds to the choice

$$q(\theta) = \delta(\theta) \quad (12)$$

- An unbiased approach was suggested in [14], where the mode 2 estimate is augmented as

$$\hat{\mathbf{x}}^{(2,a)} = \begin{bmatrix} \hat{x}^{(2)} \\ \hat{\theta}^{(1)} \end{bmatrix}; \quad P^{(2,a)} = \begin{bmatrix} P^{(xx,2)} & \mathbf{0} \\ \mathbf{0} & P^{(\theta\theta,1)} \end{bmatrix} \quad (13)$$

This corresponds to the choice

$$q(\theta) = \mathcal{N}\left(\theta; \hat{\theta}^{(1)}, P^{(\theta\theta,1)}\right) \quad (14)$$

It is also possible to take an approach to mixing of states of unequal dimensions that does not correspond to a choice of $q(\theta)$. For example (9b) can be replaced by

$$P^{(0,1)} = \left(\mu^{(1,1)}\Lambda^{(1)} + \mu^{(2,1)}\Lambda^{(2,a)}\right)^{-1} + M^{(0,1)} \quad (15)$$

where $\Lambda^{(i)} = (P^{(i)})^{-1}$ is the information matrix. In this case $\Lambda^{(2)}$ is augmented with zero information and the augmented information matrix is

$$\Lambda^{(2,a)} = \begin{bmatrix} \Lambda^{(2)} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad (16)$$

This may seem like an attractive approach, as it captures that for mode 2 we do not have any information about the state θ . However, we have tried this approach to mixing and advise against using it, because we found that it gives results that are numerically unstable. In short, problems occur when $\mu^{(2,j)}$ is close to unity, because then the weighted sum is almost equal to $\Lambda^{(2,a)}$, i.e. the weighted sum is nearly a singular matrix. Note also that the inverse of a weighted sum of two information matrices is not algebraically equivalent to the weighted sum of the corresponding covariance matrices.

In practical scenario the distribution $q(\theta)$ should be chosen such that it correctly represents both practical reality and theoretical properties. In terms of theoretical properties, one may wish to have an unbiased estimate. In [14] the unbiased approach (13) is shown to outperform the standard approach (11). Practical considerations include using any prior available information about the tracking scenario when selecting the distribution $q(\cdot)$.

In the simulation study presented below we compare using a uniform mixing distribution to three other choices, including the simple and unbiased approaches. Comments on how the uniform distribution can be chosen are also given.

IV. SIMULATION SETUP

We conducted a simulation study that compares different approaches to mixing for three different motion model combinations. The first combines two linear models, the second combines a linear and a non-linear model, and the third combines two non-linear models.

Here we also present how the true trajectories were generated, and we present the four different approaches to mode mixing that we compare. In the last subsection we give performance evaluation metrics that are used to compare the IMM estimators' estimation results.

A. Motion and measurement models

1) *Second and third order kinematics in 2D*: The first mode models third order kinematics, also known as nearly constant acceleration (CA) or Wiener process acceleration (WPA). The second mode models second order kinematics, also known

as nearly constant velocity (CV) or white noise acceleration (WNA). The state vectors for the two modes are

$$\mathbf{x}^{(1)} = \begin{bmatrix} \mathbf{p}^{(1)} \\ \mathbf{v}^{(1)} \\ \mathbf{a}^{(1)} \end{bmatrix} \quad \mathbf{x}^{(2)} = \begin{bmatrix} \mathbf{p}^{(2)} \\ \mathbf{v}^{(2)} \end{bmatrix} \quad (17)$$

where \mathbf{p} is position, \mathbf{v} is velocity, and the additional state $\theta^{(1)} = \mathbf{a}^{(1)}$ models the acceleration. The transition density is

$$p\left(\mathbf{x}_{k+1}^{(i)} \mid \mathbf{x}_k^{(i)}\right) = \mathcal{N}\left(\mathbf{x}_{k+1}^{(i)}; \mathbf{F}^{(i)}\mathbf{x}_k^{(i)}, \mathbf{G}^{(i)}\mathbf{Q}\mathbf{G}^{(i)\top}\right) \quad (18)$$

where the models $\mathbf{F}^{(i)}$ and $\mathbf{G}^{(i)}$ and process noise covariance \mathbf{Q} are

$$\mathbf{F}^{(1)} = \begin{bmatrix} \mathbf{I}_2 & T_s\mathbf{I}_2 & \frac{T_s^2}{2}\mathbf{I}_2 \\ \mathbf{0}_{2\times 2} & \mathbf{I}_2 & T_s\mathbf{I}_2 \\ \mathbf{0}_{2\times 2} & \mathbf{0}_{2\times 2} & \mathbf{I}_2 \end{bmatrix}; \quad \mathbf{G}^{(1)} = \begin{bmatrix} \frac{T_s^2}{2}\mathbf{I}_2 \\ T_s\mathbf{I}_2 \\ \mathbf{I}_2 \end{bmatrix} \quad (19a)$$

$$\mathbf{F}^{(2)} = \begin{bmatrix} \mathbf{I}_2 & T_s\mathbf{I}_2 \\ \mathbf{0}_{2\times 2} & \mathbf{I}_2 \end{bmatrix}; \quad \mathbf{G}^{(2)} = \begin{bmatrix} \frac{T_s^2}{2}\mathbf{I}_2 \\ T_s\mathbf{I}_2 \end{bmatrix} \quad (19b)$$

$$\mathbf{Q}^{(1)} = \sigma_{v,1}^2\mathbf{I}_2; \quad \mathbf{Q}^{(2)} = \sigma_{v,2}^2\mathbf{I}_2 \quad (19c)$$

In the simulation study presented below, the acceleration noise standard deviations are set to

$$\sigma_{v,1} = \gamma\sigma_{v,0}, \quad \sigma_{v,2} = \gamma\sigma_{v,0}, \quad (20)$$

where $\sigma_{v,0} = 1\text{m/s}^2$ and γ is a unitless scaling factor.

2) *Constant velocity and coordinated turn in 2D*: The first mode models coordinated turn (CT) kinematics, and the second mode models constant velocity (CV) kinematics (some prefer to call this ‘‘nearly’’-CV or NCV). The state vectors for the two modes are

$$\mathbf{x}^{(1)} = \begin{bmatrix} \mathbf{p}^{(1)} \\ \mathbf{v}^{(1)} \\ \omega^{(1)} \end{bmatrix} \quad \mathbf{x}^{(2)} = \begin{bmatrix} \mathbf{p}^{(2)} \\ \mathbf{v}^{(2)} \end{bmatrix} \quad (21)$$

where \mathbf{p} is position, \mathbf{v} is velocity, and the additional state $\theta^{(1)} = \omega^{(1)}$ models the turnrate. The transition density is

$$p\left(\mathbf{x}_{k+1}^{(i)} \mid \mathbf{x}_k^{(i)}\right) = \mathcal{N}\left(\mathbf{x}_{k+1}^{(i)}; \mathbf{f}^{(i)}\left(\mathbf{x}_k^{(i)}\right), \mathbf{Q}^{(i)}\right) \quad (22)$$

The motion models $\mathbf{f}^{(i)}(\cdot)$ and process noise covariances $\mathbf{Q}^{(i)}$ are

$$\mathbf{f}^{(1)}(\mathbf{x}) = \begin{bmatrix} 1 & 0 & \frac{\sin(\omega T_s)}{\omega} & \frac{-1+\cos(\omega T_s)}{\omega} & 0 \\ 0 & 1 & \frac{1-\cos(\omega T_s)}{\omega} & \frac{\sin(\omega T_s)}{\omega} & 0 \\ 0 & 0 & \cos(\omega T_s) & -\sin(\omega T_s) & 0 \\ 0 & 0 & \sin(\omega T_s) & \cos(\omega T_s) & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x} \quad (23a)$$

$$\mathbf{Q}^{(1)} = \mathbf{G}^{(1)}\text{diag}\left([\sigma_{v,1}^2, \sigma_{v,1}^2, \sigma_\omega^2]\right)\left(\mathbf{G}^{(1)}\right)^\top \quad (23b)$$

$$\mathbf{G}^{(1)} = \begin{bmatrix} \frac{T_s^2}{2}\mathbf{I}_2 & \mathbf{0}_{2\times 1} \\ T_s\mathbf{I}_2 & \mathbf{0}_{2\times 1} \\ \mathbf{0}_{1\times 2} & 1 \end{bmatrix} \quad (23c)$$

$$\mathbf{f}^{(2)}(\mathbf{x}) = \begin{bmatrix} \mathbf{I}_2 & T_s\mathbf{I}_2 \\ \mathbf{0}_{2\times 2} & \mathbf{I}_2 \end{bmatrix} \mathbf{x} \quad (23d)$$

$$\mathbf{Q}^{(2)} = \mathbf{G}^{(2)}\text{diag}\left([\sigma_{v,2}^2, \sigma_{v,2}^2]\right)\left(\mathbf{G}^{(2)}\right)^\top \quad (23e)$$

$$\mathbf{G}^{(2)} = \begin{bmatrix} \frac{T_s^2}{2}\mathbf{I}_2 \\ T_s\mathbf{I}_2 \end{bmatrix} \quad (23f)$$

In the simulation study presented below the acceleration and turnrate standard deviations are set to

$$\sigma_{v,1} = \gamma\sigma_{v,0} \quad \sigma_{v,2} = \gamma\sigma_{v,0} \quad (24a)$$

$$\sigma_{\omega} = \gamma\sigma_{\omega,0} \quad (24b)$$

where $\sigma_{v,0} = 1 \text{ m/s}^2$, $\sigma_{\omega,0} = 1 \text{ deg/s}$, and γ is a unitless scaling factor.

3) *Thrusting ballistic projectile models in 3D*: In this comparison we use models for the motion of projectiles, such as a rocket. When launched it has a thrust and this is modeled by a thrust mode (TM). After the fuel is burnt the projectile's motion is described by a ballistic mode (BM). The state vectors for the two modes, in continuous time, are

$$\mathbf{x}^{(1)}(t) = [x \ y \ z \ \dot{x} \ \dot{y} \ \dot{z} \ \alpha \ \tau]^T \quad (25a)$$

$$\mathbf{x}^{(2)}(t) = [x \ y \ z \ \dot{x} \ \dot{y} \ \dot{z} \ \alpha]^T \quad (25b)$$

where $[x \ y \ z]^T$ is position, $[\dot{x} \ \dot{y} \ \dot{z}]^T$ is velocity¹, α is the drag coefficient at subsonic speed, and the additional state $\theta^{(1)} = \tau^{(1)}$ models the thrust. The motion models, in continuous time, are

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \left(\frac{\tau}{\sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2}} + \alpha\alpha_m D \right) \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} + \mathbf{w}_1 \quad (26a)$$

$$\dot{\alpha} = \mathbf{w}_2 \quad (26b)$$

$$\dot{\tau} = \mathbf{w}_3 \quad (26c)$$

where \mathbf{w}_i are zero mean Gaussian process noises, α_m is a Mach number-dependent coefficient, D is air density and g is standard acceleration due to gravity. The model is discretized using an second-order Taylor expansion; we refer the reader to [10], [14] for the details too lengthy to be repeated here. The discretized process noise covariance matrix is

$$\mathbf{Q}^{(1)} = \begin{bmatrix} \begin{bmatrix} \frac{T_s^4}{4} \mathbf{I}_3 & \frac{T_s^2}{2} \mathbf{I}_3 \\ \frac{T_s^2}{2} \mathbf{I}_3 & T_s^2 \mathbf{I}_3 \end{bmatrix} \sigma_v^2 & \mathbf{0}_{6 \times 1} & \mathbf{0}_{6 \times 1} \\ \mathbf{0}_{1 \times 6} & \sigma_{\alpha}^2 & 0 \\ \mathbf{0}_{1 \times 6} & 0 & \sigma_{\tau}^2 \end{bmatrix} \quad (27)$$

for TM. The process noise covariance matrix $\mathbf{Q}^{(2)}$ for BM is given by removing the last row and column of $\mathbf{Q}^{(1)}$.

In the simulation study presented below the acceleration, drag-coefficient and thrust process noise standard deviations are set to

$$\sigma_{v,1} = \gamma\sigma_{v,0} \quad \sigma_{v,2} = \gamma\sigma_{v,0} \quad (28a)$$

$$\sigma_{\alpha,1} = \gamma\sigma_{\alpha,0} \quad \sigma_{\alpha,2} = \gamma\sigma_{\alpha,0} \quad (28b)$$

$$\sigma_{\tau} = \gamma\sigma_{\tau,0} \quad (28c)$$

where $\sigma_{v,0} = 10 \text{ m/s}^2$, $\sigma_{\alpha,0} = 5 \times 10^{-3} \text{ m}^2/\text{kg}$, $\sigma_{\tau,0} = 5 \text{ m/s}^2$ and γ is a unitless scaling factor.

¹ \dot{x} denotes derivative w.r.t. time of x .

4) *Measurement model*: A linear measurement model is used, the likelihood is

$$p(\mathbf{z}_k | \mathbf{x}_k^{(i)}) = \mathcal{N}(\mathbf{z}_k; H^{(i)}\mathbf{x}_k^{(i)}, R) \quad (29a)$$

$$H^{(i)} = [\mathbf{I}_d \ \mathbf{0}_{d \times (n_i - d)}] \quad (29b)$$

$$R = \sigma_w^2 \mathbf{I} \quad (29c)$$

for $i = 1, 2$, where $d = 2$ for the first two motion model combinations, and $d = 3$ for the thrusting ballistic projectile.

B. True trajectory and measurement generation

The sampling times were set to $T_s = 1 \text{ s}$ in the two 2D scenarios; in the 3D scenario the sampling time was set to $T_s = 1/30 \text{ s}$, as suggested in [10], [14]. The motion model pairs in Section IV-A were simulated for process noise scaling factors

$$\gamma \in \{10^{-2}, 10^{-1.75}, 10^{-1.5}, 10^{-1.25}, \dots, 10^{-1}, 10^{-0.75}, 10^{-0.5}, 10^{-0.25}, 1\}. \quad (30)$$

For the thrusting ballistic projectile simulation the measurement noise standard deviation was set to $\sigma_w = 2 \text{ m}$, as in [10]. For the two 2D simulations measurements were generated with measurement noise standard deviations

$$\sigma_w \in \{2, 20, 200\} \text{ m} \quad (31)$$

For each γ and each σ_w , 100 target trajectories were generated, i.e. in total $3 \times 3 \times 100 = 900$ trajectories for each 2D model pair, and $3 \times 1 \times 100 = 300$ for the 3D model pair.

1) *CV/CA scenario*: Scenarios with 200 time steps were simulated; the true mode is CV for time steps $k = 1, \dots, 50, 101, \dots, 150$, and CA for time steps $k = 51, \dots, 100, 151, \dots, 200$. Initial position and velocity were set to $\mathbf{p} = [1000 \ 1000]^T \text{ m}$ and $\mathbf{v} = [55 \ 0]^T \text{ m/s}$. Upon mode switch from CV to CA, the true acceleration was initialized as follows

$$\mathbf{a}_k = \begin{cases} a_0 \begin{bmatrix} 1 & 1 \end{bmatrix}^T \text{ m/s}^2, & k = 51 \\ -a_0 \begin{bmatrix} 1 & 1 \end{bmatrix}^T \text{ m/s}^2, & k = 151 \end{cases} \quad (32)$$

where a_0 was randomly sampled at the beginning of the simulation from the distribution $\mathcal{U}(a_0; 5, 10)$. An example true track is given in Figure 1a.

2) *CV/CT scenario*: Scenarios with 200 time steps were simulated; the true mode is CV for time steps $k = 1, \dots, 50, 101, \dots, 150$, and CT for time steps $k = 51, \dots, 100, 151, \dots, 200$. Initial position and velocity were set to $\mathbf{p} = [1000 \ 1000]^T \text{ m}$ and $\mathbf{v} = [55 \ 0]^T \text{ m/s}$. Upon mode switch from CV to CT, the turnrate was initialized as follows

$$\omega_k = \begin{cases} \omega_0 \text{ deg/s}, & k = 51 \\ -\omega_0 \text{ deg/s}, & k = 151 \end{cases} \quad (33)$$

where ω_0 was randomly sampled at the beginning of the simulation from the distribution $\mathcal{U}(\omega_0; 3.6, 7.2)$. An example true track is given in Figure 1b.

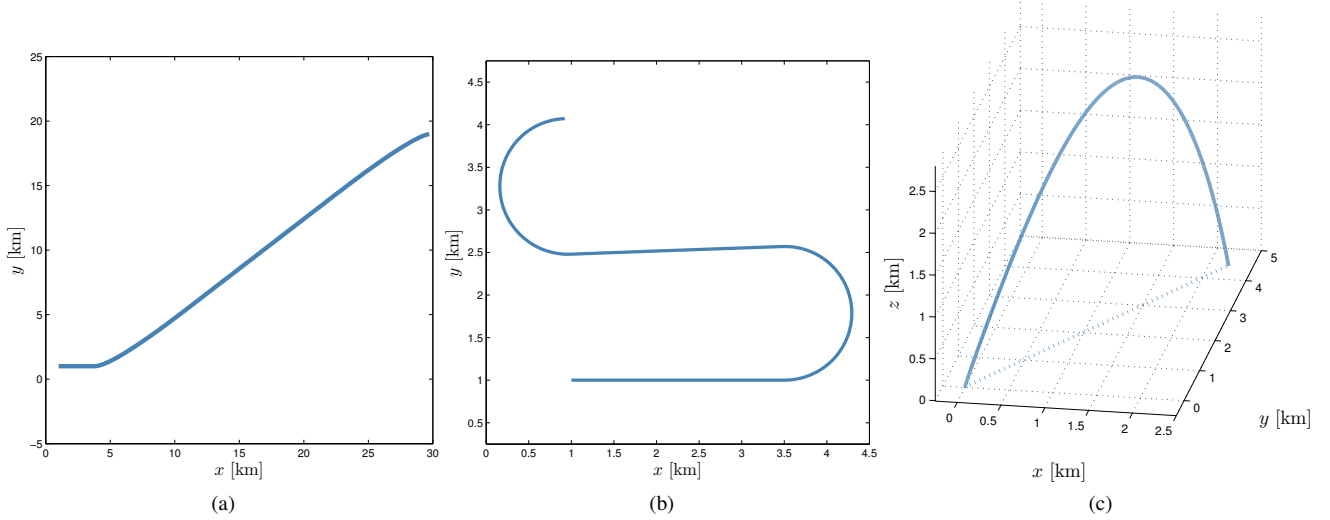


Fig. 1. Trajectory examples. (a) CA/CV, $a_0 = 5 \text{ m/s}^2$, $\gamma = 10^{-2}$. (b) CT/CV, $\omega_0 = 3.6 \text{ degrees/s}$, $\gamma = 10^{-2}$. (c) TM/BM, $\tau_0 = 100 \text{ m}^2/\text{s}$, $\gamma = 10^{-2}$. Trajectory (solid) and ground projection (dashed).

3) *TM/BM scenario*: Projectile trajectories were simulated with $\mathbf{p} = [0 \ 0 \ 0]^T \text{ m}$ as initial position and $\mathbf{v} = [0.03 \ 0.03 \ 0.09]^T \text{ m/s}$ as initial velocity.² The drag coefficient was initialized to $\alpha = 0.1025 \text{ m}^2/\text{kg}$. The initial thrust $\tau_0 \text{ m/s}^2$ was randomly sampled from the distribution $\mathcal{U}(\tau_0; 100, 500)$, and the thrust mode was simulated for four seconds. After that the ballistic mode was simulated until the projectile hit the ground (z -position = 0). The length of the simulation depends on the thrust that is applied: the larger the thrust, the longer the projectile path and the longer the simulation. An example true track is given in Figure 1c.

C. IMM estimator settings

We compare four different IMM estimators denoted I_1 , I_2 , I_3 and I_4 . The IMM estimators are identical except for the mixing step augmentation, where four different approaches are considered:

- 1) I_1 uses a simple approach $q(\theta) = \delta(\theta)$.
- 2) I_2 uses an unbiased approach [14]

$$q(\theta) = \mathcal{N}(\theta; \hat{\theta}^{(1)}, P^{(\theta\theta,1)}). \quad (34)$$

- 3) I_3 uses a uniform distribution $q(\theta) = \mathcal{U}(\theta; a, b)$. If θ is a vector, a and b are vectors of same length, and we have

$$q(\theta) = \prod_{j=1}^{n_\theta} \mathcal{U}(\theta_j; a_j, b_j) \quad (35)$$

Following the Gaussian approximation (8) the expected value and covariance of the distribution $q(\theta)$ are needed. For a uniform distribution, the expected value and covariance are

$$E[\theta] = \left[\frac{a_1+b_1}{2} \quad \dots \quad \frac{a_{n_\theta}+b_{n_\theta}}{2} \right]^T \quad (36a)$$

$$\text{Cov}(\theta) = \text{diag} \left(\left[\frac{(b_1-a_1)^2}{12} \quad \dots \quad \frac{(b_{n_\theta}-a_{n_\theta})^2}{12} \right] \right) \quad (36b)$$

²In reality the initial velocity for a thrusting ballistic projectile is zero. However, to avoid division by zero in the Matlab implementations a small initial velocity must be used.

In practice, the parameters a_j and b_j should be chosen to represent the underlying properties of the scenario. For example, if we know that the target's maximum turnrate is $\xi \text{ deg/s}$, we set $a = -\xi$ and $b = \xi$. For the three scenarios considered here, the uniform distribution parameters are chosen as

$$b = -a = 10 \text{ m/s}^2 \quad \text{for CA/CV} \quad (37a)$$

$$b = -a = 10 \text{ deg/s} \quad \text{for CT/CV} \quad (37b)$$

$$b = 500, a = 100 \text{ m/s}^2 \quad \text{for TM/BM} \quad (37c)$$

- 4) I_4 uses a high uncertainty approach $q(\theta) = \mathcal{N}(\theta; \mathbf{0}, \sigma^2 \mathbf{I})$, where the scalar standard deviation σ is chosen as large as Matlab numerically would allow in our implementations,

$$\sigma = 10^{10} \text{ m/s}^2 \quad \text{for CA/CV} \quad (38a)$$

$$\sigma = \frac{180}{\pi} 10^3 \text{ deg/s} \quad \text{for CT/CV} \quad (38b)$$

$$\sigma = 10^{16} \text{ m/s}^2 \quad \text{for TM/BM} \quad (38c)$$

In a sense this approach is analogous to (15), (16) since zero information corresponds to infinite covariance.

In all IMM estimators the mode transition probability matrix was set to

$$\begin{bmatrix} 0.99 & 0.01 \\ 0.01 & 0.99 \end{bmatrix} \quad (39)$$

In addition to comparing the performance of the four IMM estimators, we also compare to one Kalman filter based on the motion model from mode 1, denoted as K_1 ; and another Kalman filter based on the motion model from mode 2, denoted as K_2 .

D. Performance evaluation

The results for the different approaches are compared in terms of the root mean square errors (RMSE) of the estimate of target position and the mode estimate. Let the posterior

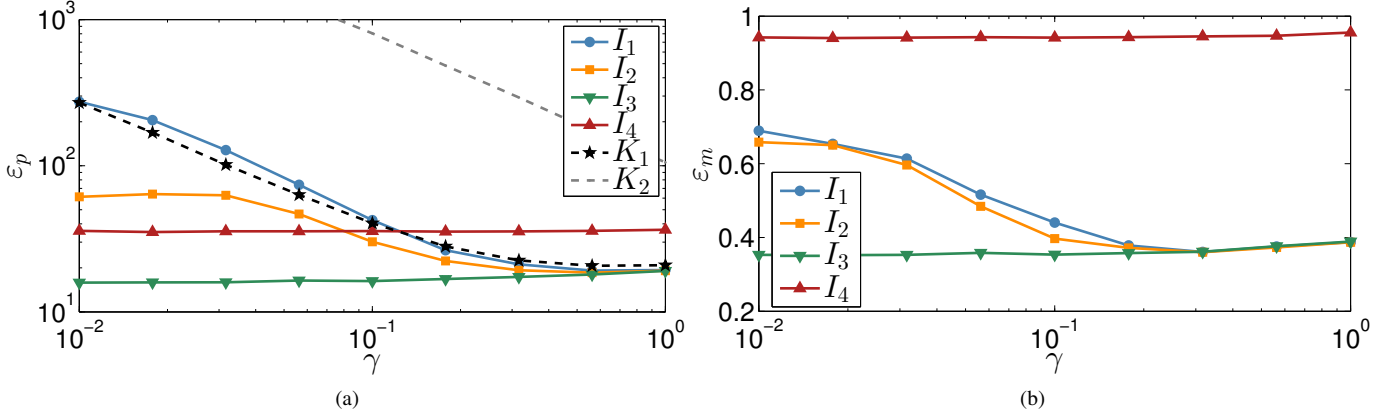


Fig. 2. CV/CA results, equal model and filter tuning parameters: Position RMSE (a) and mode RMSE (b).

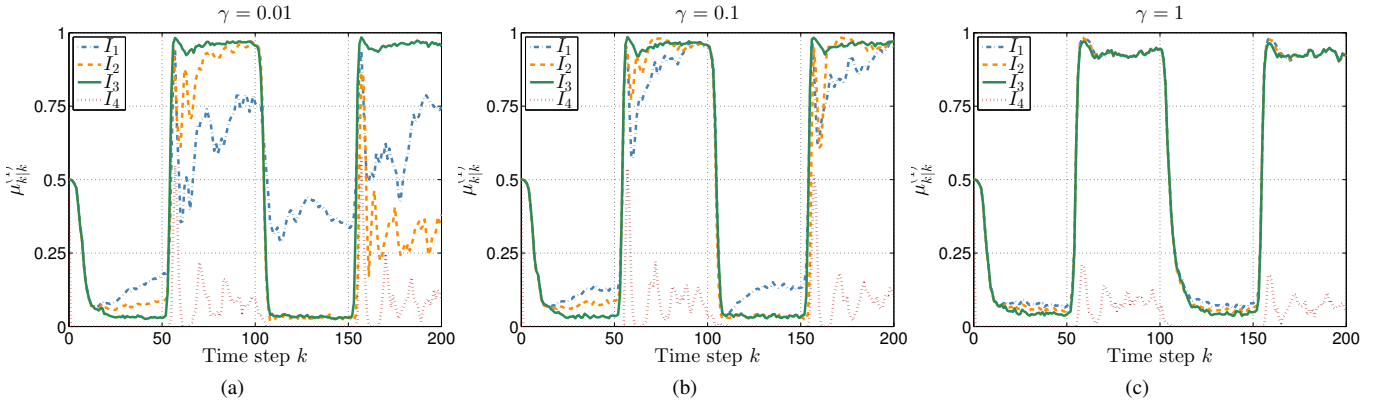


Fig. 3. Selected CV/CA results, equal model and filter tuning parameters: estimated mode probabilities for $\gamma = 0.01$ (a), $\gamma = 0.1$ (b) and $\gamma = 1$ (c). Results are averaged over 100 Monte Carlo runs.

position estimates for the two modes be denoted $\hat{\mathbf{p}}_{k|k}^{(i)}$ for $i = 1, 2$, let the true position be denoted \mathbf{p}_k , let $\mu_{k|k}^{(i)}$ for $i = 1, 2$ denote the posterior mode weights, and let the “true” mode weight vector be defined as

$$\boldsymbol{\mu}_k = \begin{cases} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}^T & \text{if } m_k = 1, \\ \begin{bmatrix} 0 & 1 \end{bmatrix}^T & \text{if } m_k = 2. \end{cases} \quad (40)$$

The position and mode RMSE are

$$\varepsilon_p = \left(\frac{1}{N} \sum_{k=1}^N \left\| \mu_{k|k}^{(1)} \hat{\mathbf{p}}_{k|k}^{(1)} + \mu_{k|k}^{(2)} \hat{\mathbf{p}}_{k|k}^{(2)} - \mathbf{p}_k \right\|^2 \right)^{\frac{1}{2}}, \quad (41a)$$

$$\varepsilon_m = \left(\frac{1}{N} \sum_{k=1}^N \left\| \begin{bmatrix} \mu_{k|k}^{(1)} & \mu_{k|k}^{(2)} \end{bmatrix}^T - \boldsymbol{\mu}_k \right\|^2 \right)^{\frac{1}{2}}. \quad (41b)$$

V. SIMULATION RESULTS

The noise parameters of the motion and measurement models — see (20), (24), (28) and (29c) — determine the properties and characteristics of the modeled target. As mentioned in the previous section, the different IMM estimators and Kalman filters are compared for data generated using different model parameters.

When simulated data are used the model parameters used to generate the data are known, and the filter tuning parameters

of the IMM estimator or Kalman filter can easily be set equal to them. However, when real world experimental data is used there are no true underlying model parameters, and tuning the estimator/filter parameters is therefore harder. It is therefore interesting to compare estimators for different parameter settings, and evaluate if any estimator is more robust to the parameter settings.

We will first present results for the case when the estimator/filter tuning parameters are equal to the simulated model parameters. Then we present results for the same scenarios, but where the filter tuning parameters are allowed to be larger or smaller than the model parameters.

A. Equal tuning and model parameters

The results for the 2D models are similar for the different measurement noise settings (see (31)), however they vary more significantly across the different process noise settings. For brevity we only illustrate the results for $\sigma_w = 20$ meters, and highlight the differences that show when the process noise settings are changed.

Position RMSE and mode RMSE results are given in Fig. 2, 4 and 6 for CA/CV, CT/CV and TM/BM, respectively. Estimated mode probabilities, averaged over 100 Monte Carlo runs, are shown in Fig. 3, 5 and 7.

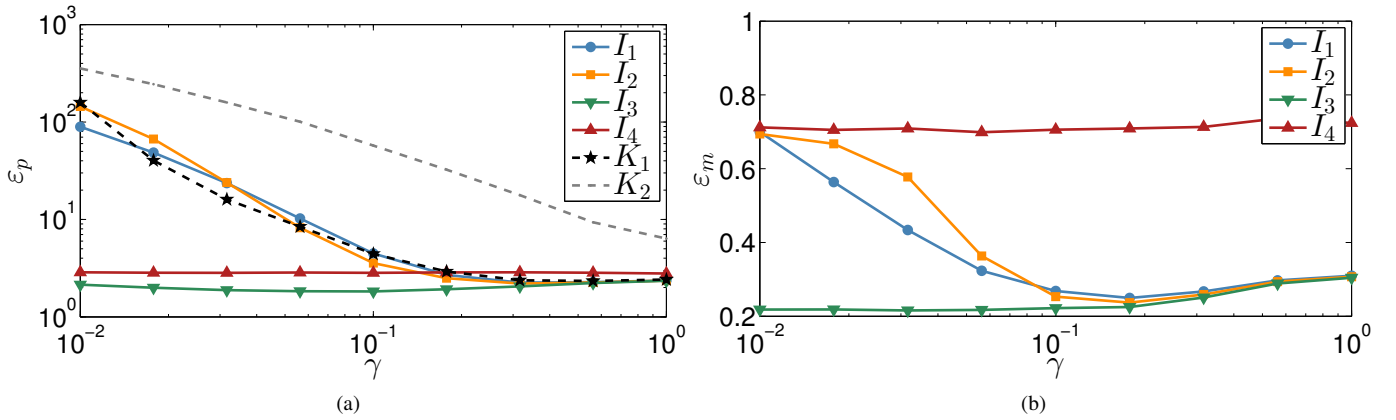


Fig. 4. CV/CT results, equal model and filter tuning parameters: Position RMSE (a) and mode RMSE (b).

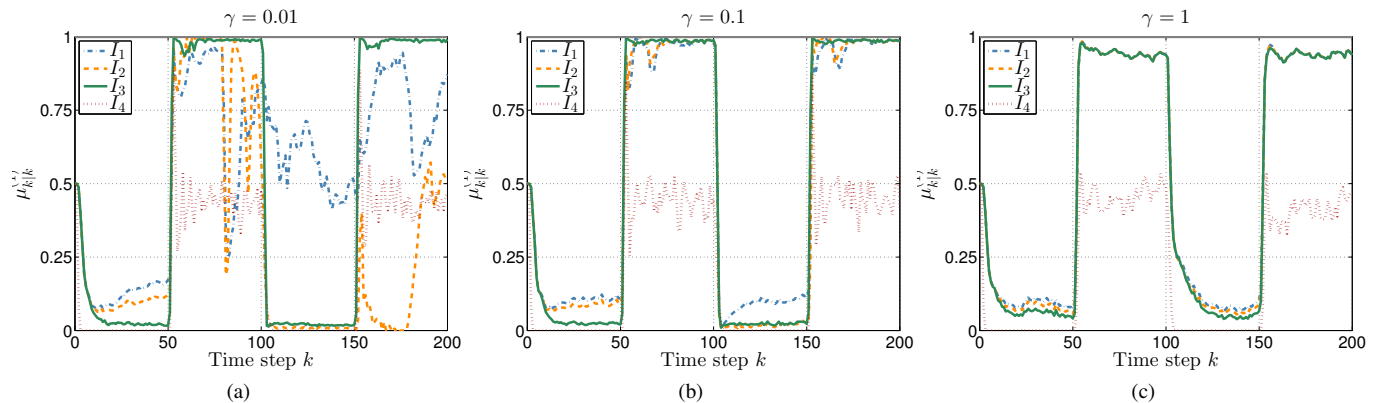


Fig. 5. Selected CV/CT results, equal model and filter tuning parameters: estimated mode probabilities for $\gamma = 0.01$ (a), $\gamma = 0.1$ (b) and $\gamma = 1$ (c). Results are averaged over 100 Monte Carlo runs.

For the 2D scenarios, the difference is largest when the process noise is low ($\gamma \leq 10^{-1}$). The performance of I_3 can only be matched for the larger simulated process noise, where all IMM estimators and the mode 1 Kalman filters (K_1) all perform similarly. In the 3D scenario I_3 performs best for all γ ; the difference in position RMSE is rather small but the mode RMSE is significantly lower.

Overall the results show that I_3 achieves best performance, as it has both lowest position RMSE and mode RMSE for all three motion model pairs. These results are intuitive:

- When low process noise is used to generate the trajectories, the differences between the two motion models stand out and hence the IMM estimators outperform the Kalman filters. When high process noise is used to generate the trajectories, the differences between the two motion models “drown” in the process noise, and modeling exactly zero acceleration/turnrate/thrust is less important. Hence the IMM estimators and mode 1 Kalman filters all perform similarly for high process noise.
- When low process noise is used to generate the trajectories, the largest changes are due to the initializations a_0 , ω_0 and τ_0 . This is accurately captured by the uniform augmentation distribution in I_3 , however the augmentation methods used in I_1 , I_2 and I_4 do not model this

well enough. Hence, I_3 outperform the other three IMM estimators.

For the second of these two points, it is important to note that the proposed augmentation method, i.e. the one used in I_3 , models the initial values for the additional state θ , without negatively affecting the estimation performance for the states that are common to both modes. The standard augmentation methods used in I_1 and I_2 do not capture these initial values, and for lower γ the process noise cannot correctly capture the “jumps” (sudden increases) that the initial values represent — hence the poor performance of I_1 and I_2 compared to I_3 . The suggested augmentation method overcomes the low process noise and introduces a robustness in the IMM estimator.

The difference between the IMM estimators for low process noise can possibly be reduced if the corresponding tuning parameters are changed. This is investigated in the next simulation comparison.

B. Free tuning parameters

For each value of γ , the IMM estimators and Kalman filters were tested with process noise tuning parameters that were larger, equal to, and smaller than the process noise parameters used to generate the trajectories. Results for CA/CV, CT/CV and TM/BM, are shown in Fig. 8, 9 and 10, respectively. For

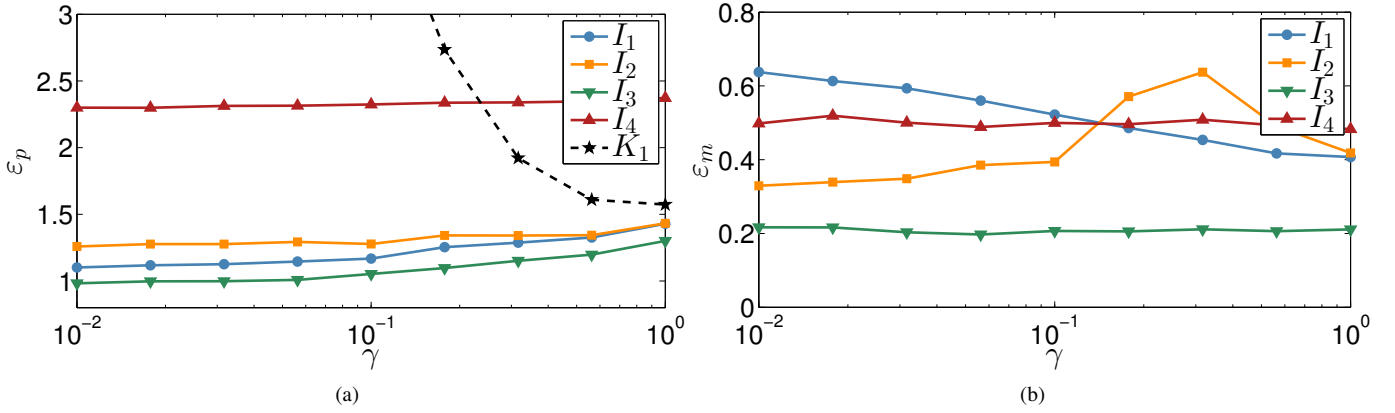


Fig. 6. TM/BM results, equal model and filter tuning parameters: Position RMSE (a) and mode RMSE (b). The position RMSE for K_2 are very large, and are excluded to make the plot easier to read.

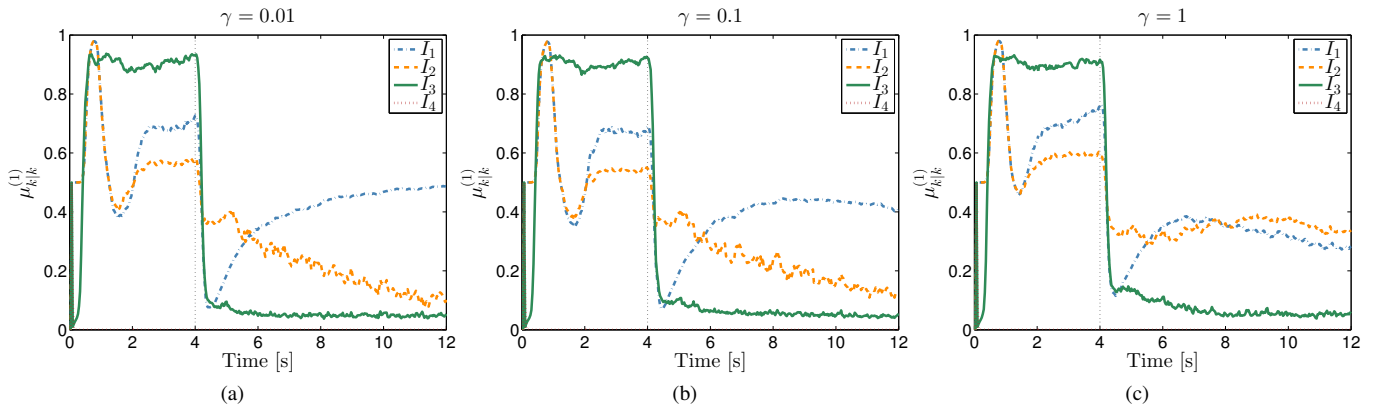


Fig. 7. Selected TM/BM results, equal model and filter tuning parameters: estimated mode probabilities for the first 12 seconds for $\gamma = 0.01$ (a), $\gamma = 0.1$ (b) and $\gamma = 1$ (c). Results are averaged over 100 Monte Carlo runs.

clarity, the filter tuning parameter is denoted here as $\hat{\gamma}$, while the true value in the model is γ .

The results show that IMM estimators I_1 and I_2 achieve their lowest position RMSE when the process noise parameter is tuned to a high value: a factor 10^2 higher than the true value used to generate the trajectory. By increasing the process noise the jumps in the initial values of θ are better estimated by the IMM estimator. The lowest position RMSEs for I_1 and I_2 are almost as low as the lowest position RMSE for I_3 , however the position RMSE for the Kalman filters are equally low, i.e. the conditions are such that there is very little gain in using an IMM estimator instead of a Kalman filter.

It is noteworthy that I_3 achieves low position and mode RMSE even when the process noise is tuned very low, up to a factor 10^{-2} lower than the true value used to generate the trajectory. This shows that using carefully tuned distribution $q(\theta)$ for the mode augmentation gives an IMM estimator that is more robust to parameter tuning, compared to the two standard approaches.

VI. CONCLUSIONS

A systematic approach to mode mixing with unequal state dimensions was taken, and the simulation study shows that the

choice of augmentation distribution has a significant impact on the performance of the IMM estimator. When the augmentation distribution for the mixing is chosen to reflect the magnitude of the “extra” state component, significant performance increases are achieved compared to the standard approaches that can be found in the literature. The IMM estimators based on the standard approaches can only match the performance in scenarios with such high process noise that there is essentially very little benefit to using an IMM estimator instead of a Kalman filter.

The proposed augmentation approach yields an IMM estimator that can correctly estimate “jumps” in the states that occur upon mode switching, is insensitive to the process noise settings, and can achieve this without affecting the estimation of the states that are common to all modes in the IMM estimator. The proposed approach is shown to introduce a significant robustness in the IMM estimator.

REFERENCES

- [1] T. Ardeshtiri, K. Granström, E. Özkan, and U. Orguner, “Greedy reduction algorithms for mixtures of exponential family,” *IEEE Signal Processing Letters*, vol. 22, no. 6, pp. 676–680, Jun. 2015.
- [2] Y. Bar-Shalom, S. Challa, and H. A. P. Blom, “The IMM Estimator as the Natural Approximation of the Optimal Estimator for Hybrid Systems,”

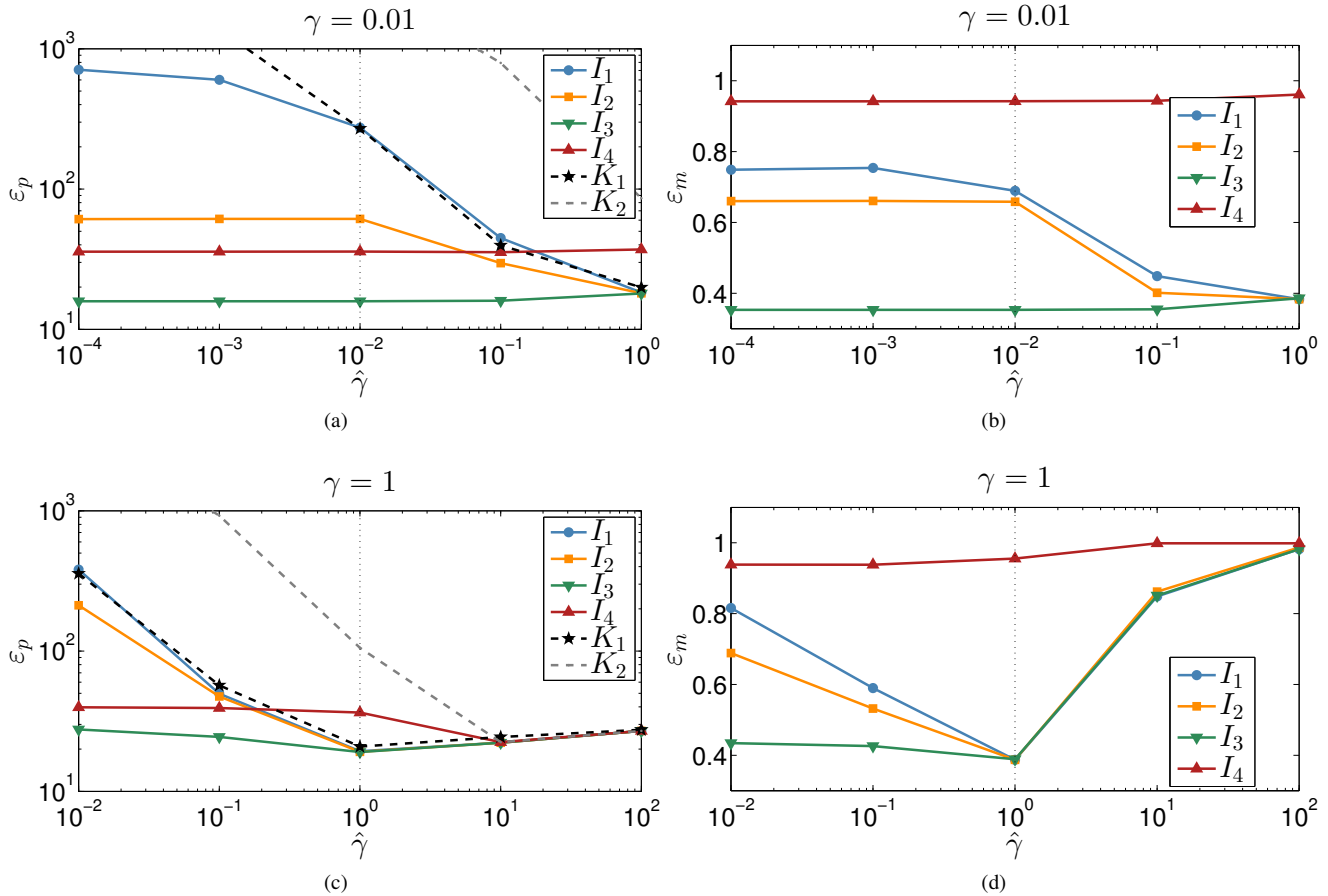


Fig. 8. Selected CA/CV results, free filter tuning parameters: position RMSE ((a) and (c)) and mode RMSE ((b) and (d)) for different simulated model process noise magnitudes γ . The dashed black lines indicate $\hat{\gamma} = \gamma$, i.e. the points that are plotted in Fig. 2. Note γ is true, $\hat{\gamma}$ is parameter to which the filters are tuned.

- IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 3, pp. 986–991, Jul. 2005.
- [3] Y. Bar-Shalom, P. K. Willett, and X. Tian, *Tracking and data fusion, a handbook of algorithms*. YBS Publishing, 2011.
- [4] C. M. Bishop, *Pattern recognition and machine learning*. New York, USA: Springer, 2006.
- [5] H. Blom and Y. Bar-Shalom, “The interacting multiple model algorithm for systems with Markovian switching coefficients,” *IEEE Transactions on Automatic Control*, vol. 33, no. 8, pp. 780–783, Aug. 1988.
- [6] K. Granström and U. Orguner, “On the Reduction of Gaussian inverse Wishart mixtures,” in *Proceedings of the International Conference on Information Fusion*, Singapore, Jul. 2012, pp. 2162–2169.
- [7] S. Kullback and R. A. Leibler, “On information and sufficiency,” *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, Mar. 1951.
- [8] R. Lopez, P. Danes, and F. Royer, “Extending the IMM filter to heterogeneous-order state space models,” in *Proceedings of the IEEE Conference on Decision and Control*, Atlanta, GA, USA, Dec. 2010, pp. 7369–7374.
- [9] T. Minka, “A family of algorithms for approximate Bayesian inference,” Ph.D. dissertation, Massachusetts Institute of Technology, Jan. 2001.
- [10] V. C. Ravindra, Y. Bar-Shalom, and P. Willett, “Projectile identification and impact point prediction,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, no. 4, pp. 2004–2021, Oct. 2010.
- [11] A. R. Runnalls, “Kullback-Leibler approach to Gaussian mixture reduction,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 3, pp. 989–999, Jul. 2007.
- [12] D. Schieferdecker and M. F. Huber, “Gaussian Mixture Reduction via Clustering,” in *Proceedings of the International Conference on Information Fusion*, Seattle, WA, USA, Jul. 2009.
- [13] J. L. Williams and P. S. Maybeck, “Cost-Function-Based Gaussian Mixture Reduction for Target Tracking,” in *Proceedings of the International Conference on Information Fusion*, Cairns, Queensland, Australia, Jul. 2003.
- [14] T. Yuan, Y. Bar-Shalom, P. Willett, E. Mozeson, S. Pollak, and D. Hardiman, “A multiple IMM estimation approach with unbiased mixing for thrusting projectiles,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 3250–3267, Oct. 2012.



Karl Granström (M’08) is a postdoctoral research fellow at the Department of Electrical and Computer Engineering at University of Connecticut, USA. He received the MSc degree in Applied Physics and Electrical Engineering in May 2008, and the PhD degree in Automatic Control in November 2012, both from Linköping University, Sweden. He previously held a postdoctoral position at the Department of Electrical Engineering of Linköping University from December 2012 to August 2014. His research interests include estimation theory, multiple model estimation, sensor fusion and target tracking, especially for extended targets. He has received paper awards at the FUSION 2011 and FUSION 2012 conferences.

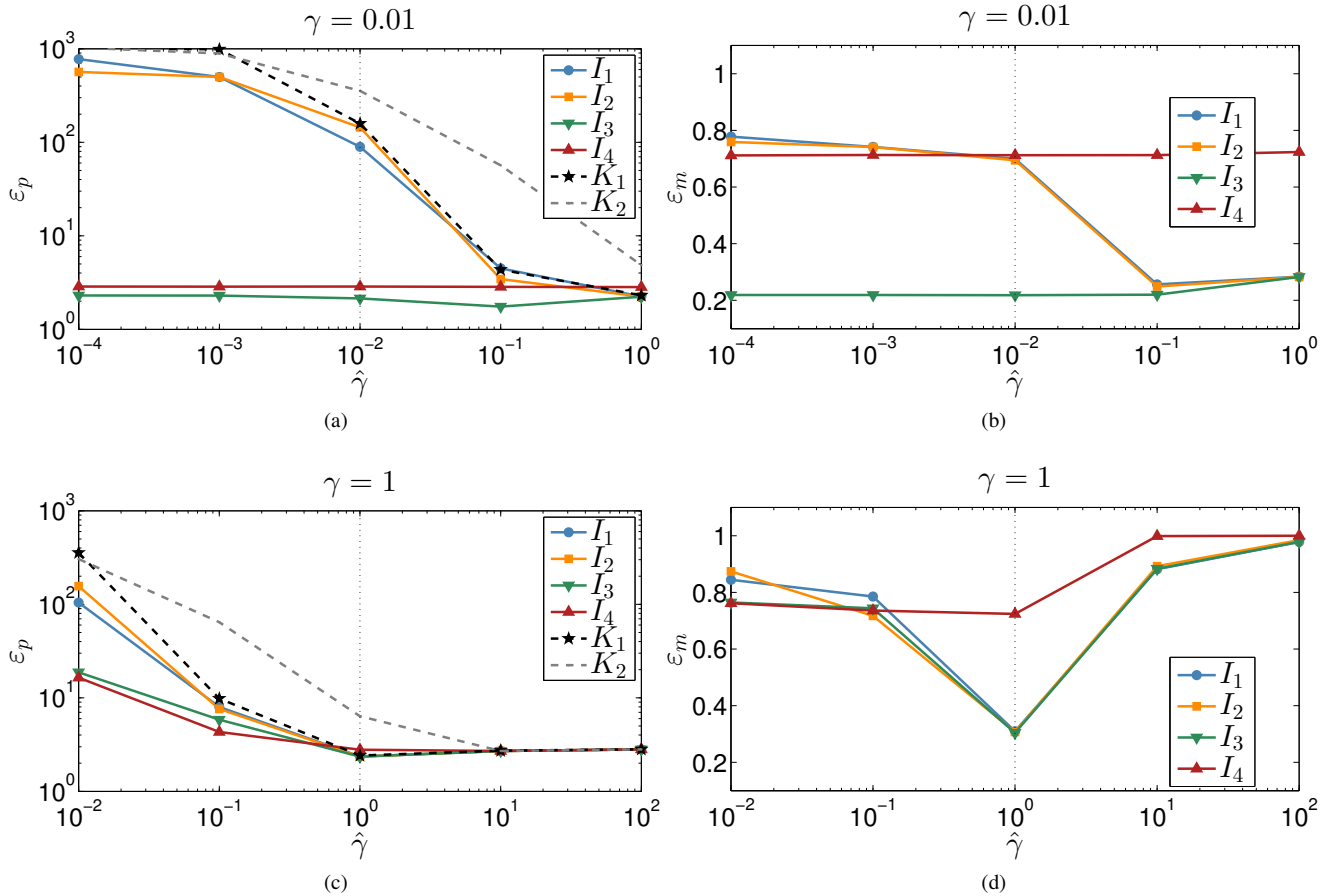


Fig. 9. Selected CT/CV results, free filter tuning parameters: position RMSE ((a) and (c)) and mode RMSE ((b) and (d)) for different simulated model process noise magnitudes γ . The dashed black lines indicate $\hat{\gamma} = \gamma$, i.e. the points that are plotted in Fig. 4. Note γ is true, $\hat{\gamma}$ is parameter to which the filters are tuned.



Peter Willett (F'03) received his B.A.Sc. (engineering science) from the University of Toronto in 1982, and his Ph.D. degree from Princeton University in 1986. He has been a faculty member at the University of Connecticut ever since, and since 1998 has been a Professor. His primary areas of research have been statistical signal processing, detection, machine learning, data fusion and tracking. He also has interests in and has published in the areas of change/abnormality detection, optical pattern recognition, communications and industrial/security condition monitoring. He is editor-in-chief of *IEEE Signal Processing Letters*. He was editor-in-chief for *IEEE TRANSACTIONS ON AEROSPACE AND ELECTRONIC SYSTEMS* (2006-2011), and was Vice President for Publications for AESS (2012-2014). He is a member of the IEEE AESS Board of Governors 2003-2009, 2011 to present. He was General Co-Chair (with Stefano Coraluppi) for the 2006 ISIF/IEEE Fusion Conference in Florence, Italy and for the 2008 ISIF/IEEE Fusion Conference in Cologne, Germany, Program Co-Chair (with Eugene Santos) for the 2003 IEEE Conference on Systems, Man & Cybernetics in Washington DC, and Program Co-Chair (with Pramod Varshney) for the 1999 Fusion Conference in Sunnyvale.



Yaakov Bar-Shalom received the B.S. and M.S. degrees from the Technion, Israel Institute of Technology, in 1963 and 1967 and the Ph.D. degree from Princeton University in 1970, all in electrical engineering. Currently he is Board of Trustees Distinguished Professor in the Dept. of Electrical and Computer Engineering and Marianne E. Klewin Professor in Engineering at the University of Connecticut. He is also Director of the ESP (Estimation and Signal Processing) Lab. His current research interests are in estimation theory, target tracking and data fusion. He has published over 500 papers and book chapters in these areas and in stochastic adaptive control as well as 8 books including *Estimation with Applications to Tracking and Navigation* (Wiley, 2001), *Tracking and Data Fusion* (YBS Publishing, 2011). He has been elected Fellow of IEEE for "contributions to the theory of stochastic systems and of multitarget tracking". He served as President of the International Society of Information Fusion in 2000 and 2002 and Vice President for Publications in 2004-13. He is corecipient of the M. Barry Carlton Award for the best paper in the *IEEE Transactions on Aerospace and Electronic Systems* in 1995 and 2000 and recipient of the J. Mignona Data Fusion Award from the DoD JDL Data Fusion Group. He is a member of the Connecticut Academy of Science and Engineering. In 2008 he was awarded the IEEE Dennis J. Picard Medal for Radar Technologies and Applications, and in 2012 the Connecticut Medal of Technology. He has been listed by *academic.research.microsoft* (top authors in engineering) as #1 among the researchers in Aerospace Engineering based on the citations of his work.

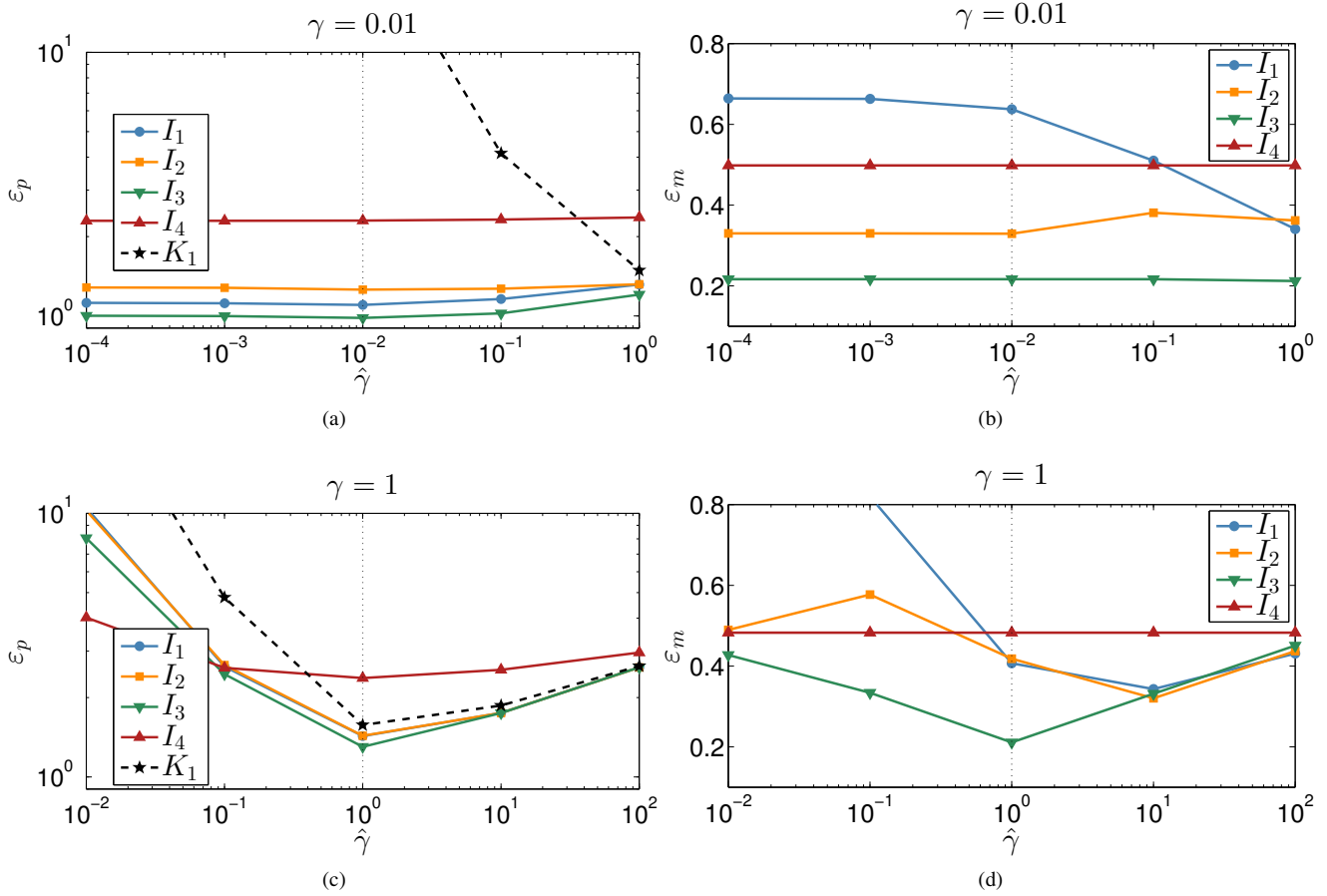


Fig. 10. Selected TM/BM results, free filter tuning parameters: position RMSE ((a) and (c)) and mode RMSE ((b) and (d)) for different simulated model process noise magnitudes γ . The position RMSE for K_2 are very large, and are excluded to make the plot easier to read. The dashed black lines indicate $\hat{\gamma} = \gamma$, i.e. the points that are plotted in Fig. 6. Note γ is true, $\hat{\gamma}$ is parameter to which the filters are tuned.