Runtime Optimization in Interacting Multiple Model Filtering with Down-Sampling and Out-of-Sequence Measurements

Pascal Ketterer, Patrick Hoher, Johannes Reuter
Institute of System Dynamics (ISD)
University of Applied Sciences Konstanz (HTWG)
Konstanz, Germany
pascal_ketterer@web.de, {phoher, jreuter}@htwg-konstanz.de

***Abstract*—Interacting multiple model filters are most commonly used in the context of maneuvering targets, as they can represent the different dynamics of a real system by combining the estimates of multiple models. However, the interacting multiple model approach generally requires more computational effort than a single Kalman filter. In this work, down-sampling is used to reduce the computational effort. We propose an adaptive scheme to maintain the accuracy of the estimator to a defined level. To this end, the trace of the innovation covariance matrix is evaluated, and if it lies above a certain threshold, out-of-sequence measurements are iteratively used to improve the estimate until the uncertainty threshold is met. The approach is evaluated by Monte Carlo analysis. The results show that with this approach, the number of measurements to be processed, and thus the computational effort can be dynamically reduced, while the accuracy remains at a desired level.**

***Index Terms*—IMM, OOSM, runtime optimization**

I. Introduction

The estimation of maneuvering targets usually reflects the motion dynamics, which can be described by different underlying mathematical motion models such as the constant velocity (CV), constant acceleration (CA) and constant turn (CT) model. Multiple model (MM) approaches can combine the information of independent filters to achieve an improved estimation of the system state. They are therefore particularly used in such applications. One of the most commonly used MM methods is the interacting multiple model (IMM) approach, which was introduced in 1988 [1] and is considered the best compromise between computational cost and accuracy.

The continued scientific relevance of the IMM algorithm is reflected in its use in various modern applications. In [2] from 2020, the IMM algorithm is used for visual-inertial navigation (VIN) of microaerial vehicles (MAV)s. VINs are modules that combine information from a camera and an inertial measurement unit. In this paper, a conventional VIN and a drag force VIN are used in an IMM framework. Especially in aggressive flight maneuvers, drag force VINs offer improved performance over the conventional variant. Conventional VINs, on the other hand, prove themselves in scenarios in which the MAVs are disturbed by external influences. This paper demonstrates the increased performance by combining both approaches using the IMM algorithm. The 2017 publication [3] applies the IMM algorithm in the context of autonomous driving. Autonomous driving is highly complex due to the interaction of the vehicle with the surrounding traffic. To increase the reliability of the environment estimations, the information from on board sensors and the communication between the vehicles is combined. The dedicated short-range communication (DSRC) system is used for wireless communication between the vehicles. In this paper, the use of a track-based IMM algorithm with sequential hypothesis testing (IMM-SMHT) is presented to solve the problem of inherent uncertainty of the observation data and the delays and information loss. The work in [4] from 2018 shows an interesting application of the IMM algorithm in a medical domain. This work uses the IMM framework for two particle filters and a probabilistic data association for a real-time motion model analysis of human gait. With this approach, information about a patient's kinematics is analyzed in order to appropriately adjust a patient's prosthesis by a walking robot. Accuracy and robustness have been demonstrated while maintaining high efficiency by using a small number of particles for the particle filters. Further recent research and applications of the IMM algorithm can be found, for example, in [5]–[7].

In this paper, we propose a novel usage of the IMM algorithm in combination with down-sampling and methods of out-of-sequence measurement (OOSM)s to maintain a defined uncertainty level and decrease the computational effort. Especially, by using different motion models, the uncertainty in the estimates varies over time. By down-sampling the obtained information is reduced yielding a higher uncertainty compared to an in-sequence measurement (ISM) processing of all measurements. Our approach applies down-sampling on the received measurements to decrease the computational effort. The concept of OOSMs, which allows to update the measurement at the current time index by earlier measurements, is then used to adaptive apply skipped measurements to meet a pre-defined uncertainty level.

This paper is organized as follows. Section II contains the outline of the IMM algorithm, followed by the extension of the IMM algorithm by the OOSM approach. Next, the basic concept of unbiased mixing is briefly explained using the example from [8]. This section provides a more general overview

of solving the unbiased mixing problem for different system setups. The approach proposed in this paper to reduce the computational effort while maintaining an uncertainty threshold is then described in the following section. Section V defines the setup, such as the motion models and the reference trajectory, as well as the parameters used in the simulation. Visual comparisons and numerical evaluations are used to demonstrate the differences in the estimates and computational effort of the different setups investigated in this work in Section VI.

II. Fundamentals

A. Interacting Multiple Model

The IMM algorithm is briefly presented below. A more detailed introduction with graphical representation can be found in [9] and a complete pseudocode using the linear Kalman filter implementation is given in [10, p. 1276, Table II]. The basic idea of the IMM approach is to combine the information provided by r independent filters running in parallel. The IMM algorithm allows a seamless switch between the different mathematical models M_k at time index k from a finite set of models defined as [11, p. 444]

$$M_k \in \left\{ M^{(j)} \right\}_{j=1}^{r}. \tag{1}$$

A property common to all Markov processes is that the time between switching from one state to another is described by an exponentially distributed random variable [12, p. 775]. For two dimensions, this is given by

$$\Pi(T) = \frac{1}{k_s} \begin{pmatrix} k_2 + k_1 e^{-(k_s)T} & k_1 - k_1 e^{-(k_s)T} \\ k_2 - k_2 e^{-(k_s)T} & k_1 + k_2 e^{-(k_s)T} \end{pmatrix}, \tag{2}$$

where k_1 and k_2 are tuning parameters, $k_s = k_1 + k_2$ and $T = t_k - t_{k-1}$ [12, p. 782]. The matrix elements $\pi^{(ij)} \in \Pi(T)$ define the probability of a transition from model $M^{(i)}$ to model $M^{(j)}$. The sum over the row elements is thereby one. In general, the sampling time T is static, which in turn simplifies the transition probability matrix so that it is time-invariant with static probability elements. With the transition between the individual models defined, the remaining IMM algorithm can be summarized in four steps: interaction, model-dependent filtering, model probability update and state combination, which are discussed for example in [13] and [11, p. 453 f.]. In this work, the first step is of particular importance, which is the key part of the IMM algorithm. In this step, the reduction of the computational effort is achieved by the approximation of the model-dependent posterior probability density function (PDF)

$$p\left(x_{k-1|k-1} \middle| M_{k-1}^{(j)}, Z^{k-1} \right) \tag{3}$$

at time index $k-1$. The exact solution for the model-dependent posterior PDF is given by the weighted summation of the posterior PDF of each model used, represented by Gaussian components [11, p. 454]

$$\sum_{i=1}^{r} \mathcal{N}\left(x_k; E\left\{ x_k \middle| M_k^{(j)}, \hat{x}_{k-1|k-1}^{(i)} \right\}, cov\left\{\cdot\right\} \right) \mu_{k-1|k-1}^{(ij)}, \tag{4}$$

where $\mu_{k-1|k-1}^{(ij)}$ is the mixing probability used to weight the components. To reduce the computational effort, the joint Gaussian PDF is approximated by a single Gaussian component defined as [11, p. 454]

$$\mathcal{N}\left(x_k; E\left\{ x_k \middle| M_k^{(j)}, \sum_{i=1}^{r} \hat{x}_{k-1|k-1}^{(i)} \mu_{k-1|k-1}^{(ij)} \right\}, cov\left\{\cdot\right\} \right), \tag{5}$$

where $cov\{\cdot\}$ is the covariance using the same argument as the prior expectation value $E\{\cdot\}$. Equation (5) shows that the approximation requires a weighted summation of the individual models. Therefore, the state vectors must have the same dimensions, which is not always the case. Section III shows how to deal with state vectors of different dimensions. The focus here is on the unbiased mixing approach.

B. IMM with Out-of-Sequence Measurements

In tracking, it is usually assumed that a state evolves from timestamp t_{k-1} to timestamp t_k. This assumption is not always guaranteed, especially with multi sensor systems. Data from the same sensor may arrive at the processing unit with different time delays. It is therefore possible that a measurement with an earlier timestamp t_τ, so that

$$t_{k-1} < t_\tau < t_k \tag{6}$$

has been received. This type of problem involves so called OOSMs and is discussed in detail in [14].

The concept consists of three steps. First, the state retrodiction, in which a backward prediction of the state at time index τ based on the estimated state $\hat{x}_{k|k}$ at time index k, resulting in $\hat{x}_{\tau|k}$. In the next step, the update components, namely the residual and the Kalman gain, are calculated similarly to the standard Kalman filter algorithm. Finally, the state $\hat{x}_{k|k}$ is updated at time index k using the previously determined update components, resulting in the state estimates $\hat{x}_{k|\tau}$. If a measurement with a time index greater than the current time index k is received, the standard Kalman filter algorithms are applied.

The extension of this approach for application to the IMM algorithm has already been presented in [15]. This approach uses the concept of *equivalent measurement* [16], which allows to solve the OOSM problem defined as

$$t_{k-l-1} < t_\tau < t_{k-l} \tag{7}$$

with multiple delays, in one step. The concept of the IMM with OOSMs is to use the algorithms of the standard IMM method when a measurement with timestamp $t > t_k$ is received. If the measurement has an earlier timestamp than the most recent state, the methods of OOSM are applied. In the IMM approach with OOSM, the state retrodiction is calculated independently for each model. In contrast to the standard IMM forward filtering no interaction step is performed. Further differences are that the mixing probability $\mu_{k-1|k-1}^{(ij)}$ has to be retrodicted and is later updated based on the likelihood of how well the specific models match the true target motion.

III. Generalized Unbiased Mixing

The interaction step of the IMM algorithm requires additional effort if the state vectors of the different motion models used have different dimensions. The example in [8, p. 3251 f.] provides a fairly simple example for dealing with different dimensions in the state vectors using the example of the two models

$$x^{(1)} = \begin{pmatrix} x_c \\ x_e \end{pmatrix} \quad \text{and} \quad x^{(2)} = \begin{pmatrix} x_c \\ \underline{0} \end{pmatrix}, \tag{8}$$

where the first model contains common states x_c and extra states x_e. The second model only contains the common states x_c, therefore zeros need to be added to the second model to be able to mix both states. This results in a bias for the extra states due to the additional zeros of the second model. An approach that removes the bias towards zero is called unbiased mixing. The concept of unbiased mixing is to augment the lower dimension state vector with the additional estimate of the first state vector. In the remainder of this paper, however, the concept of runtime optimized out-of-sequence measurement (RO-OOSM) is introduced and simulation results are shown using the three different motion models CV, CA and CT. In this case, the CA and CT models contain different additional states, which is not discussed in [8]. Therefore, we want to introduce unbiased mixing more generally and therefore demonstrate unbiased mixing using the following four state vectors and covariances:

$$x^{(1)} = x_c \qquad P^{(1)} = P_c \tag{9}$$

$$x^{(2)} = \begin{pmatrix} x_c \\ x_{e_1} \end{pmatrix} \qquad P^{(2)} = \begin{pmatrix} P_c & P_{ce_1} \\ P_{e_1c} & P_{e_1} \end{pmatrix} \tag{10}$$

$$x^{(3)} = \begin{pmatrix} x_c \\ x_{e_1} \end{pmatrix} \qquad P^{(3)} = \begin{pmatrix} P_c & P_{ce_1} \\ P_{e_1c} & P_{e_1} \end{pmatrix} \tag{11}$$

$$x^{(4)} = \begin{pmatrix} x_c \\ x_{e_2} \end{pmatrix} \qquad P^{(4)} = \begin{pmatrix} P_c & P_{ce_2} \\ P_{e_2c} & P_{e_2} \end{pmatrix}. \tag{12}$$

The first model only contains the states x_c that are common to all other models. Models two and three also contain the same additional states x_{e_1}, which would be possible, for example, by using the same mathematical models with different process noise settings. The last model contains the additional states $x_{e_2} \neq x_{e_1}$. In this scenario, the preparation before unbiased mixing depends on the state vector to be mixed. The mixing for the first model can be done quite simply by ignoring the additional states and covariance elements of the other models and using the standard IMM mixing. The second and third models use the same states. To apply unbiased mixing, the states x_{e_1} and the corresponding covariances of models two and three must be premixed. The standard IMM mixture is again used for premixing, whereby the model probabilities of these models must be normalized so that they add up to $100\,\%$. The resulting premixed states and covariances, defined as $\hat{x}_p$ and P_p, are then used to extend models one and four. Since the fourth model also contains additional states x_{e_2} that are not used by models two and three, these additional states must be ignored before the premixed state and covariance are added. The premixing for the fourth state vector is again quite simple as it is the only model that contains the states x_{e_2}. Therefore, the premixed state and the covariance are equal to x_{e_2} and P_{e_2} respectively. After premixing, unbiased mixing can be applied, where the general calculation for the model $M^{(j)}$ is defined by

$$\hat{x}_M^{(j)} = \sum_{i=0}^{r} \begin{pmatrix} \hat{x}_c^{(i)} \\ \hat{x}_e^{(i)} \end{pmatrix} \mu^{(ij)} \quad \text{and} \tag{13}$$

$$P_M^{(j)} = \sum_{i=0}^{r} = \left[\begin{pmatrix} P_c^{(i)} & P_{ce}^{(i)} \\ P_{ec}^{(i)} & P_e^{(i)} \end{pmatrix} + \begin{pmatrix} \tilde{P}_c^{(i)} & \tilde{P}_{ce}^{(i)} \\ \tilde{P}_{ec}^{(i)} & \tilde{P}_e^{(i)} \end{pmatrix} \right] \mu^{(ij)}, \tag{14}$$

where the elements of the covariance are

$$\tilde{P}_c^{(i)} = (\hat{x}_c^{(i)} - \hat{x}_c^{(j)})(\hat{x}_c^{(i)} - \hat{x}_c^{(j)})^\intercal \tag{15}$$

$$\tilde{P}_{ce}^{(i)} = (\hat{x}_c^{(i)} - \hat{x}_c^{(j)})(\hat{x}_e^{(i)} - \hat{x}_e^{(j)})^\intercal \tag{16}$$

$$\tilde{P}_{ec}^{(i)} = (\hat{x}_e^{(i)} - \hat{x}_e^{(j)})(\hat{x}_c^{(i)} - \hat{x}_c^{(j)})^\intercal \tag{17}$$

$$\tilde{P}_e^{(i)} = (\hat{x}_e^{(i)} - \hat{x}_e^{(j)})(\hat{x}_e^{(i)} - \hat{x}_e^{(j)})^\intercal. \tag{18}$$

The terms $\hat{x}_e^{(i)}$ and $P_e^{(i)}$ equal the premixed estimates $\hat{x}_p^{(j)}$ and $P_p^{(j)}$ if the model itself does not contain these extra states. From the assumption that the states of different models are independent, it follows that

$$P_{ce}^{(i)} = P_{ec}^{(i)} = \underline{0} \tag{19}$$

for all models extended by the premixed state estimates [8, p. 3261 f.]. Since

$$\hat{x}_e^{(i)} = \hat{x}_e^{(j)} = \hat{x}_p^{(j)}, \tag{20}$$

it holds

$$\tilde{P}_{ce}^{(i)} = \tilde{P}_{ec}^{(i)} = \tilde{P}_e^{(i)} = \underline{0} \tag{21}$$

for all extended models.

IV. Runtime Optimized IMM with OOSM

In this paper, we propose a variant of the IMM OOSM algorithm, the RO-OOSM approach, which allows a lower computational load while maintaining a certain uncertainty threshold U. We want to introduce the idea of the proposed algorithm through the progression of the trace of the innovation covariance matrix S for the ISM and down-sampled processing over time, which is shown in Fig. 1. Analyzing these traces, we find that the traces fluctuate over time and obviously there is a higher uncertainty due to the information reduction in the down-sampling approach. For the proposed algorithm, we define an uncertainty threshold U that lies somewhere between these two traces, at least most of the time. The first step is to down-sample the measurements and process only every nth measurement using the standard IMM algorithm. In addition to the standard IMM algorithm, the uncertainty is calculated by the trace of the innovation covariance matrix S. The second step consists of iteratively evaluating and processing the measurements skipped due to down-sampling until the uncertainty

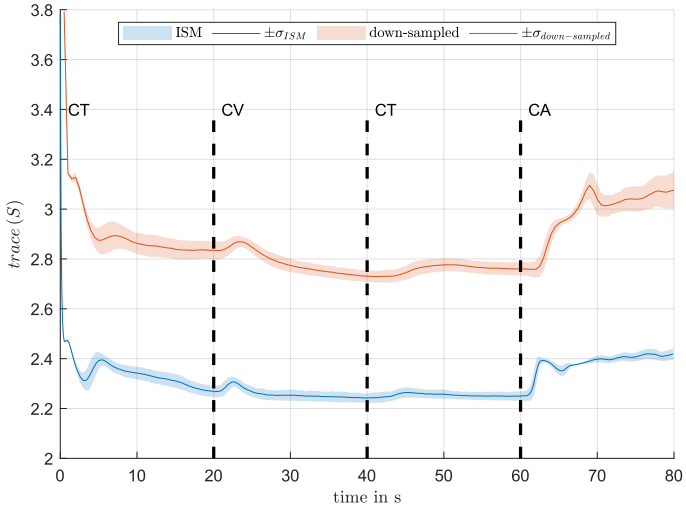


Figure 1: Comparison of the traces for ISM processing and the down-sampled processing of the measurements

threshold is met. The effect of this approach is that not all measurements are required to maintain a predefined uncertainty, thus reducing the overall computational effort. Moreover, a more stable uncertainty is obtained over the whole scenario, where the uncertainty criterion can be chosen by the user. In this work, for example, we used the trace of the innovation covariance matrix as uncertainty criterion.

One cycle of the RO-OOSM process is represented by the pseudocode in Algorithm 1. Even though we use the concept of OOSM and it is not strictly necessary, we still assume that the measurements arrive at the processing unit in the correct order. The intermediate results and the required quantities are specified for each sub-step. The processing of the measurements begins with the most recently skipped measurements, which, as already mentioned (see section II-A), have the greatest influence on the currently estimated system state. Note that the state can only be improved by $n-1$ measurements. If the uncertainty is still violated afterwards, no further improvement can be achieved. Explicit formulas for standard IMM filtering and IMM OOSM with the OOSM B-algorithm with multiple lags, using the linear Kalman filter algorithms can be found, for example, in [15]. However, this algorithm can easily be transferred to other OOSM algorithms or variants of the Kalman filter algorithm such as the extended Kalman filter algorithm. In this paper, we restrict the OOSM problem to a OOSM problem with a single lag, which is defined in (6). For this reason, the IMM OOSM approach implemented in this work is based on the suboptimal OOSM B-algorithm presented in [14, p. 772].

An essential part of this approach is to find the range of practically usable thresholds. We focus on a static threshold for the whole scenario. To estimate a suitable threshold, the discrete algebraic Riccati equation [11, p. 211], [17, p. 39]

$$P = F\left[P - PH^{\intercal}\left(HPH^{\intercal}+R\right)^{-1}HP\right]F^{\intercal}+Q \qquad (22)$$

Algorithm 1: Pseudocode RO-OOSM approach

Data: $\left\{\hat{x}^{(j)}_{k|k}, P^{(j)}_{k|k}\right\}_{j=1}^{r}$, $\{z_{k+i}\}_{i=1}^{n}$, $\{t_{k+i}\}_{i=0}^{n}$, $\mu_{k|k}$, U
Result: $\hat{x}_{k+n|Z_k}$, $P_{k+n|Z_k}$, $\mu_{k+n|Z_k}$

step 1 Standard IMM filtering (down-sampled)
$dt = t_k - t_{k+n}$
$Z_k = z_{k+n}$
for *j=1, …, r* **do**
  step 1.1 Interaction/Mixing
  $\hat{x}^{(0j)}_{k|k}, P^{(0j)}_{k|k} \leftarrow \left\{\hat{x}^{(i)}_{k|k}, P^{(i)}_{k|k}\right\}_{i=1}^{r}, \mu_{k|k}, \Pi\,(dt)$
  step 1.2 Model-matched filtering
  $\hat{x}^{(j)}_{k+n|k}, P^{(j)}_{k+n|k} \leftarrow \hat{x}^{(0j)}_{k|k}, P^{(0j)}_{k|k}, F^{(j)}_{k}, Q^{(j)}_{k}$
  $\hat{z}^{(j)}_{k+n}, S^{(j)}_{k+n} \leftarrow \hat{x}^{(j)}_{k+n|k}, P^{(j)}_{k+n|k}, H^{(j)}_{k}, R^{(j)}_{k}$
  $\hat{x}^{(j)}_{k+n|Z_k}, P^{(j)}_{k+n|Z_k} \leftarrow \hat{x}^{(j)}_{k+n|k}, P^{(j)}_{k+n|k}, \hat{z}^{(j)}_{k+n}, S^{(j)}_{k+n}$
step 1.3 Model probability update
$\mu_{k+n|Z_k} \leftarrow \left\{\hat{z}^{(i)}_{k+n}, S^{(i)}_{k+n}\right\}_{i=1}^{r}, \mu_{k|k}$
step 1.4 State combination
$\hat{x}_{k+n|Z_k}, P_{k+n|Z_k} \leftarrow \left\{\hat{x}^{(i)}_{k+n|Z_k}, P^{(i)}_{k+n|Z_k}\right\}_{i=1}^{r}, \mu_{k+n|Z_k}$
step 1.5 Uncertainty determination
$trace\,(S_{k+n}) \leftarrow P_{k+n|Z_k}, H_k, R_k$

step 2 RO-OOSM
$n_z = 1$
step 2.1 Uncertainty validation
while $trace\,(S_{k+n}) > U$ *and* $n_z < n$ **do**
  $\tau = t_{k+n-n_z}$
  $dt = \tau - t_{k+n}$
  $Z_\tau = \{z_{k+i}\}_{i=n-n_z}^{n}$
  $n_z = n_z + 1$
  for *j=1, …, r* **do**
    step 2.2 Model-matched state retrodiction
    $\hat{x}^{(j)}_{\tau|k+n}, P^{(j)}_{\tau|k+n}, P^{(j)}_{xv} \leftarrow \hat{x}^{(j)}_{k+n|Z_k}, P^{(j)}_{k+n|Z_k}, P^{(j)}_{k+n|k}, Q^{(j)}_{k+n,\tau}, H^{(j)}_{k+n}$
    $\hat{z}^{(j)}_{\tau}, S^{(j)}_{\tau} \leftarrow \hat{x}^{(j)}_{\tau|k+n}, P^{(j)}_{\tau|k+n}, P^{(j)}_{xv}, H^{(j)}_{\tau}, R^{(j)}_{\tau}$
    $\hat{x}^{(j)}_{k+n|Z_\tau}, P^{(j)}_{k+n|Z_\tau} \leftarrow \hat{x}^{(j)}_{\tau|k+n}, P^{(j)}_{\tau|k+n}, P^{(j)}_{xz}, z_\tau, R^{(j)}_{\tau}$
  step 2.3 Model probability update
  $\mu_{k+n|Z_\tau} \leftarrow \left\{\hat{z}^{(i)}_{\tau}, S^{(i)}_{\tau}\right\}_{i=1}^{r}, \mu_{k+n|Z_k}, \Pi\,(dt)^{\intercal}$
  step 2.4 State combination
  $\hat{x}_{k+n|Z_\tau}, P_{k+n|Z_\tau} \leftarrow \left\{\hat{x}^{(i)}_{k+n|Z_\tau}, P^{(i)}_{k+n|Z_\tau}\right\}_{i=1}^{r}, \mu_{k+n|Z_\tau}$
  step 2.5 Uncertainty update
  $trace\,(S_{k+n}) \leftarrow P_{k+n|Z_\tau}, H_\tau, R_\tau$
  $Z_k = Z_\tau$
return $\left\{\hat{x}^{(j)}_{k+n|Z_k}, P^{(j)}_{k+n|Z_k}\right\}_{j=1}^{r}$, $\mu_{k+n|Z_k}$, t_{k+n}

can be applied. The discrete algebraic Riccati equation enables the determination of the covariance $P = \lim_{k\to\infty} P_k$ for linear mathematical models, from which the trace of the innovation covariance $\lim_{k\to\infty} S_k$ can be calculated. To find the corresponding range, we evaluate the discrete algebraic Riccati equation twice. The first time with the discrete time interval for the ISM of all measurements and the second time with the time interval resulting from down-sampling. The uncertainty threshold can then be selected somewhere between these two values.

V. Simulation

In the following section, the simulation setup used for the investigation of the RO-OOSM is presented. The aim of the simulations is to investigate whether the proposed RO-OOSM approach leads to the desired reduction in computational effort and runtime and how well the specified uncertainty thresholds are met.

A. Setup

For the simulation, we will use the three motion models CV, CA and CT in a two-dimensional scenario, where the state vectors are

$$x_k^{(CV)} = \begin{pmatrix} x & y & v_x & v_y \end{pmatrix}^\intercal \tag{23}$$

$$x_k^{(CA)} = \begin{pmatrix} x & y & v_x & v_y & a_x & a_y \end{pmatrix}^\intercal \tag{24}$$

$$x_k^{(CT)} = \begin{pmatrix} x & y & v_x & v_y & \omega \end{pmatrix}^\intercal . \tag{25}$$

The model dependent transition density is defined as

$$p\left(x_{k+1}^{(j)} | x_k^{(j)}\right) = \mathcal{N}\left(x_{k+1}^{(j)}; F^{(j)} x_k^{(j)}, G^{(j)} Q^{(j)} G^{(j)\intercal}\right), \tag{26}$$

where the transition matrices $F^{(j)}$, input matrices $G^{(j)}$ and process noise covariances $Q^{(j)}$ are [18, p. 1498]

$$F^{(CV)} = \begin{pmatrix} I_2 & TI_2 \\ 0_{2\times2} & I_2 \end{pmatrix}, \; F^{(CA)} = \begin{pmatrix} I_2 & TI_2 & \frac{T^2}{2} I_2 \\ 0_{2\times2} & I_2 & TI_2 \\ 0_{2\times2} & 0_{2\times2} & I_2 \end{pmatrix},$$

$$F^{(CT)} = \begin{pmatrix} 1 & 0 & \frac{sin(\omega T)}{\omega} & -\frac{1-cos(\omega T)}{\omega} & 0 \\ 0 & 1 & \frac{1-cos(\omega T)}{\omega} & \frac{sin(\omega T)}{\omega} & 0 \\ 0 & 0 & cos(\omega T) & -sin(\omega T) & 0 \\ 0 & 0 & sin(\omega T) & cos(\omega T) & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$G^{(CV)} = \begin{pmatrix} \frac{T^2}{2} I_2 \\ TI_2 \end{pmatrix}, \qquad Q^{(CV)} = \sigma_a^2 I_2$$

$$G^{(CA)} = \begin{pmatrix} \frac{T^2}{2} I_2 \\ TI_2 \\ I_2 \end{pmatrix}, \qquad Q^{(CA)} = \sigma_a^2 I_2$$

$$G^{(CT)} = \begin{pmatrix} \frac{T^2}{2} I_2 & 0_{2\times1} \\ TI_2 & 0_{2\times1} \\ 0_{1\times2} & T \end{pmatrix}, \quad Q^{(CT)} = \begin{pmatrix} \sigma_a^2 I_2 & 0_{2\times1} \\ 0_{1\times2} & \sigma_\alpha^2 \end{pmatrix}.$$

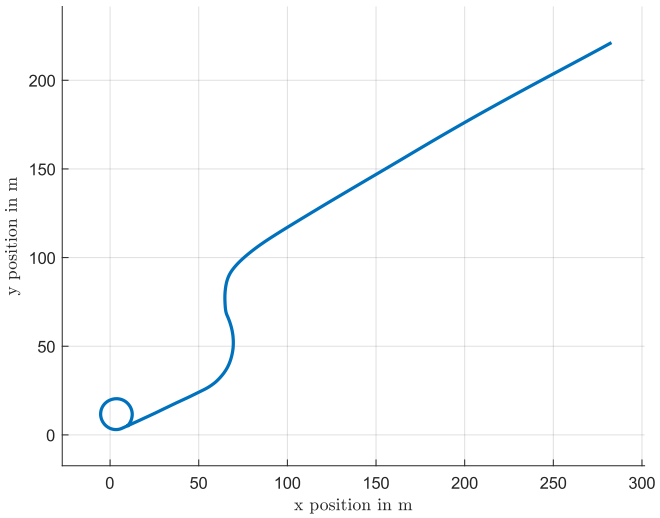


Figure 2: Reference trajectory

The linear state to measurement density is

$$p\left(z_k | x_k^{(j)}\right) = \mathcal{N}\left(z_k; H^{(j)} x_k^{(j)}, R^{(j)}\right), \tag{27}$$

where the output matrices $H^{(j)}$ are defined as

$$H^{(CV)} = \begin{pmatrix} I_2 \\ 0_{2\times2} \end{pmatrix}^\intercal, \; H^{(CA)} = \begin{pmatrix} I_2 \\ 0_{4\times2} \end{pmatrix}^\intercal, \; H^{(CT)} = \begin{pmatrix} I_2 \\ 0_{2\times2} \end{pmatrix}^\intercal$$

and the measurement noise covariance is equally defined for all models as

$$R^{(CV)} = R^{(CA)} = R^{(CT)} = \begin{pmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{pmatrix}.$$

Next, we define the time-dependent model transition probability matrix. An example of this is shown in section II-A. However, deriving an explicit calculation rule turns out to be quite difficult unless all eigenvalues are different [12, p. 778]. We therefore assume that the change from model $M^{(j)}$ to any model $M^{(i)}$, with $j \neq i$, has the same probability. The matrix element for remaining in the same model (the diagonal elements of $\Pi(\Delta t)$) can be calculated by

$$\pi^{(ii)}(\Delta t) = e^{-k_i \Delta t}. \tag{28}$$

Under the above assumption that the transition to any other model has the same probabilities, the remaining matrix elements can be easily calculated using the fact that the individual rows of the transition probability matrix sum to one.

B. Reference Trajectory

The simulations in this work are carried out with the reference trajectory shown in Fig. 2. The trajectory is divided into four sections, each with 200 data points, which correspond to different motion models. The first part corresponds to a movement with CT. The second part consists of a movement with CV that transitions into a movement with CT. The turn rate is significantly less dynamic than in the first part of the

Table I: Trace calculation based on the discrete algebraic Riccati equation

motion model	**sampling time** T	$\lim_{k\to\infty} trace\,(S_k)$
CV	0.10	2.09
	0.50	2.50
CA	0.10	2.44
	0.50	3.59

trajectory. In the last part, a movement with CA is performed. The resulting measured values are sampled at a frequency of 10 Hz. The process noise for the random acceleration in x- and y-direction has the standard deviation $\sigma_a = 1\,\mathrm{m\,s^{-2}}$ and the process noise for the random angular acceleration is defined as $\sigma_\alpha = 0.005\,\mathrm{s^{-2}}$ for the CT motion model.

The standard deviations for the measurement noise in the x- and y-directions are chosen to be the same as $\sigma_x = \sigma_y = 1\,\mathrm{m}$. Based on these standard deviations, the model-dependent process covariance matrix $Q^{(j)}$ and the measurement covariance matrix R are determined to approximate the expected covariance matrix for $k \to \infty$. The discrete algebraic Riccati equation is used to calculate $\lim_{k\to\infty} P_k$ and thus the uncertainty threshold $U = \lim_{k\to\infty} trace\,(S_k)$, in the regions where the model probabilities for the linear CV and CA model are dominant. Tab. I shows the results for the two linear models for the ISM with $T = 0.1\,\mathrm{s}$ and the down-sampling approach, where only every fifth measurement is processed, resulting in $T = 0.5\,\mathrm{s}$.

Based on this evaluation, the two uncertainty thresholds 2.40 and 2.80 are defined for the investigations. In the following section, we use RO-OOSM$_{2.40}$, to refer to the results of the RO-OOSM approach with uncertainty threshold 2.40 and RO-OOSM$_{2.80}$, to refer to the results of the RO-OOSM approach with uncertainty threshold 2.80.

In addition, we define the probability that a motion model remains in the same model as 97 %, which gives the tuning parameter k_i in (28) as 0.3046 for $T = 0.1\,\mathrm{s}$.

VI. Results

The following results are based on a Monte Carlo simulation of 1000 runs in MATLAB on an AMD Ryzen 7 5825U processor. In down-sampling, only every fifth measurement is processed, which in turn means that a minimum of one and a maximum of five measurements per iteration are processed by the RO-OOSM algorithm. Fig. 4b shows the total number of measurements used in the iterations of RO-OOSM. On average, 3.35 measurements are used in the RO-OOSM$_{2.40}$ approach and 1.36 measurements in RO-OOSM$_{2.80}$ instead of the five measurements, used in the ISM approach in the same time interval. Fig. 3 shows the state estimates with the corresponding uncertainty ellipses for the experimental setups with unbiased mixing in the CV motion of the reference. For better illustration, only the results of RO-OOSM$_{2.40}$ are shown. For better comparability, only every fifth measurement is shown for the ISM approach and only the updated states are shown for the RO-OOSM$_{2.40}$ method. These results are consistent with the traces of the innovation covariances shown in Fig. 4a and

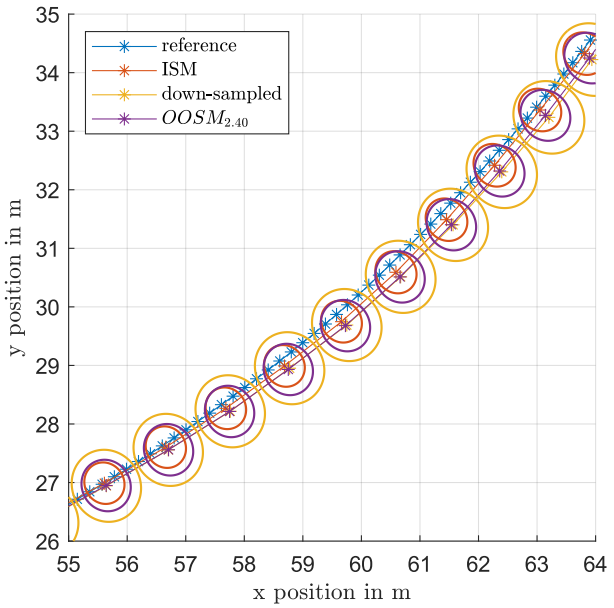
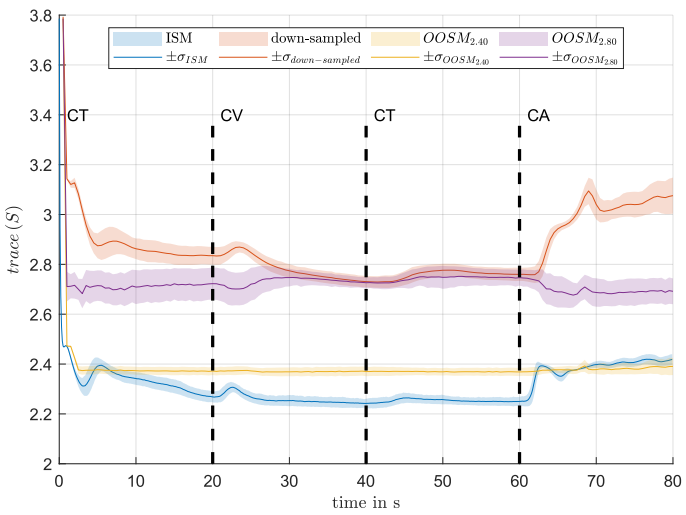


Figure 3: State comparison applying unbiased mixing for the uncertainty threshold of 2.40

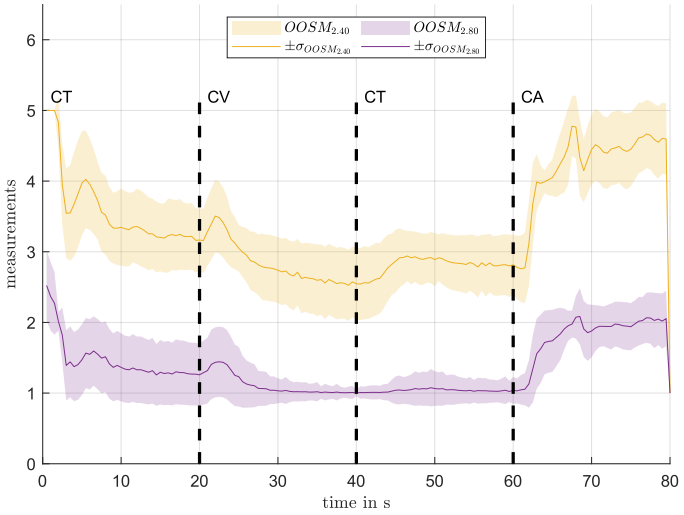
confirm that the uncertainties or uncertainty ellipses for the RO-OOSM$_{2.40}$ method are slightly larger than for the ISM method and even larger for the down-sampling method.

The comparison of the results of the RO-OOSM$_{2.80}$ approach in Fig. 5 shows an only slightly better uncertainty ellipse than the down-sampling approach also determined in the trace comparison (see Fig. 4a). In the range of the CV motion of the reference, the trace comparison shows that the traces of the RO-OOSM$_{2.80}$ approach and the down-sampling approach converge to each other, which is also shown by the similar estimates and uncertainty ellipses in Fig. 6. This agreement can be explained by the number of measurements used (see Fig. 4b). In this range, approximately one measurement is used per iteration, which corresponds to the number used in the down-sampling approach in the same time interval.

Tab. II shows the runtimes of the setups investigated in this work. The runtimes represent the mean value over the Monte Carlo simulations that the filters require for a complete estimation of the reference trajectory. The evaluation shows that the IMM with unbiased mixing generally increases the performance with respect to the root means square (RMS) position and RMS model probability error. However, this additional effort for the interaction step IMM is clearly reflected in the computational efficiency. In contrast, down-sampling the measurements leads to a reduction in the required runtime by the value of the skipped values, which is quite intuitive. Here, down-sampling shows the largest negative impact on the position estimation, while the estimation of the model likelihood is only slightly affected. The evaluation of the standard approach without unbiased mixing, which is shown in the first line of Tab. II, compared to RO-OOSM$_{2.40}$ with unbiased



(a) Trace comparison



(b) Number of used measurements

Figure 4: Runtime reduction performance comparison and number of used measurements for the OOSM approach with unbiased mixing

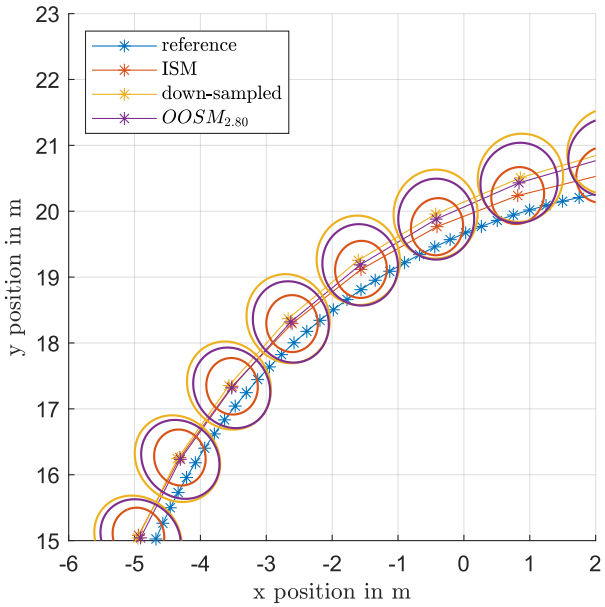


Figure 5: State comparison for the uncertainty threshold of 2.80 in the first CT motion, applying unbiased mixing

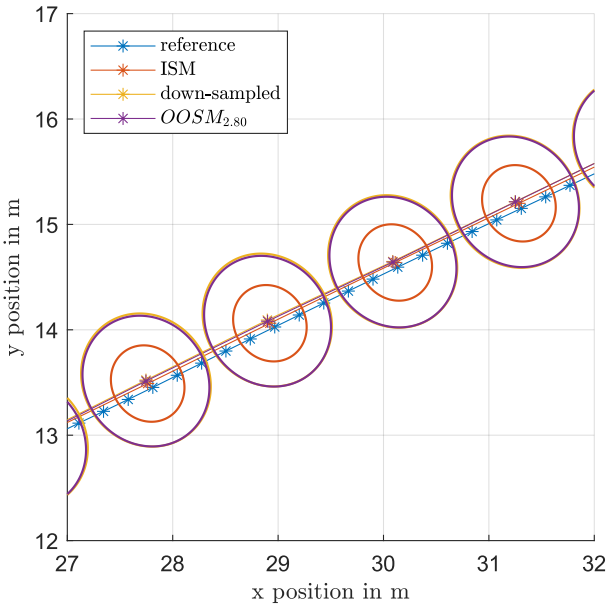


Figure 6: State comparison for the uncertainty threshold of 2.80 in the CV motion, applying unbiased mixing

mixing, gives similar results for the RMS position error and the required runtime. Nevertheless, the RO-OOSM$_{2.40}$ approach with unbiased mixing maintains a better estimate for the model probabilities. Another effect can be seen when comparing the runtimes of IMM with unbiased mixing and down-sampling with the OOSM approach when processing all measurements. This method has a lower computational cost than the ISM approach, which can be explained by the independence of the models for the IMM with OOSM processing. In the IMM approach with OOSMs, the interaction step for processing the OOSMs is omitted. This reduces the computational costs, as the unbiased mixture is only performed for every fifth measurement. However, it should be noted that the estimated state is not perfectly reconstructed due to the non-optimal OOSM B-algorithm and the non-linear CT motion model, the estimated state is not perfectly reconstructed, resulting in a less accurate position estimate. The computational effort for the down-sampling approach and the OOSM processing of

Table II: Performance comparison

unbiased mixing	down-sampling	OOSM	runtime optimized	RMS position error in m	RMS model probability error in %	runtime in ms
				0.20	72.68	79.92
	✗			0.49	74.53	16.01
	✗	✗		0.28	72.79	100.49
	✗	✗	✗[1]	0.33	73.63	64.25
	✗	✗	✗[2]	0.44	74.56	26.07
✗				0.17	60.43	137.75
✗	✗			0.32	62.87	27.57
✗	✗	✗		0.20	58.98	116.27
✗	✗	✗	✗[1]	0.22	60.08	81.87
✗	✗	✗	✗[2]	0.29	62.41	38.52

[1] Uncertainty threshold 2.40 [2] Uncertainty threshold 2.80

all skipped measurements shows that the computational effort without unbiased mixing is larger than with the ISM processing. The higher computational effort is due to the additional matrix inversions performed in the OOSM algorithm. The results for down-sampling with the RO-OOSM method in this paper show a significant reduction in computational effort. The comparison between the uncertainty thresholds of 2.40 and 2.80 shows that the lower the requirements for the uncertainty limits, the more effective the reduction in the computing power of this algorithm.

The great advantage of this approach is that the maximum computational effort can be estimated taking into account all OOSMs. The algorithm therefore tends to be more efficient, as not all measurements always have to be processed in order to comply with the uncertainty limit. Another effect is that by choosing a suitable uncertainty boundary, a more constant uncertainty level can be maintained over the entire scenario if the selected uncertainty boundary U fulfills the condition

$$trace\left(S_{k,ISM}\right) \leq U \leq trace\left(S_{k,down}\right) \quad \forall k \in \mathbb{N}, \qquad (29)$$

where $S_{k,ISM}$ is the innovation covariance matrix at time index k for ISM processing of all data and $S_{k,down}$ is the innovation covariance matrix at time index k for down-sampled processing of the measurements.

VII. Conclusion

In this paper, we have shown the advantage of using unbiased mixing instead of augmentation with zeros. Then we have presented a generic instruction for using unbiased mixing for a variety of different scenarios using an example with four different state vectors. After that, we have presented the main contribution of this paper, namely the RO-OOSM approach, which allows, under certain conditions, to meet a predefined uncertainty threshold and achieve a reduction in computational effort. In the last sections, the simulation environment, which demonstrates the functionality of the approach, and the results are presented. It has been shown that the selection of an appropriate uncertainty threshold can be a difficult task, especially when nonlinear motion models are used where the covariance cannot be estimated using the discrete algebraic Riccati equation. Different configurations were compared for the results, which show that unbiased mixing generally leads to a better estimate at the expense of computational efficiency. In contrast, the results for the RO-OOSM approach show that the lower the uncertainty requirements are chosen, the less computational effort is required.

References

[1] 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, 1988.

[2] M. A. K. Gomaa, O. De Silva, G. K. I. Mann, and R. G. Gosine, "Observability-constrained vins for mavs using interacting multiple model algorithm," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 3, pp. 1423–1442, 2021.

[3] T. Yuan, K. Krishnan, Q. Chen, J. Breu, T. B. Roth, B. Duraisamy, C. Weiss, M. Maile, and A. Gern, "Object matching for inter-vehicle communication systems—an imm-based track association approach with sequential multiple hypothesis test," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 12, pp. 3501–3512, 2017.

[4] G. Chalvatzaki, X. S. Papageorgiou, C. S. Tzafestas, and P. Maragos, "Augmented human state estimation using interacting multiple model particle filters with probabilistic data association," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1872–1879, 2018.

[5] E. Hill, S. A. Gadsden, and M. Biglarbegian, "Explicit nonlinear mpc for fault tolerance using interacting multiple models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 5, pp. 2784–2794, 2021.

[6] X. Fan, G. Wang, J. Han, and Y. Wang, "Interacting multiple model based on maximum correntropy kalman filter," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 8, pp. 3017–3021, 2021.

[7] S. Rahimifard, R. Ahmed, and S. Habibi, "Interacting multiple model strategy for electric vehicle batteries state of charge/health/ power estimation," *IEEE Access*, vol. 9, pp. 109 875–109 888, 2021.

[8] 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, 2012.

[9] A. Genovese, "The interacting multiple model algorithm for accurate state estimation of maneuvering targets," *Johns Hopkins APL Technical Digest (Applied Physics Laboratory)*, vol. 22, pp. 614–623, 10 2001.

[10] X. Rong Li and V. Jilkov, "Survey of maneuvering target tracking. part v. multiple-model methods," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1255–1321, 2005.

[11] Y. Bar-Shalom, T. Kirubarajan, and X.-R. Li, *Estimation with Applications to Tracking and Navigation*. USA: John Wiley & Sons, Inc., 2002.

[12] A. Papoulis, *Probability, Random Variables and Stochastic Processes*, 4th ed. New York: McGraw-Hill, 2002.

[13] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan, "Interacting multiple model methods in target tracking: a survey," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 1, pp. 103–123, 1998.

[14] Y. Bar-Shalom, "Update with out-of-sequence measurements in tracking: exact solution," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 3, pp. 769–777, 2002.

[15] Y. Bar-Shalom and H. Chen, "IMM estimator with out-of-sequence measurements," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 1, pp. 90–98, 2005.

[16] Y. Bar-Shalom, M. Mallick, H. Chen, and R. Washburn, "One-step solution for the general out-of-sequence-measurement problem in tracking," in *Proceedings, IEEE Aerospace Conference*, vol. 4, 2002, pp. 4–4.

[17] B. Anderson and J. Moore, *Optimal Filtering*. Prentice-Hall, 1979.

[18] X. Yuan, C. Han, Z. Duan, and M. Lei, "Comparison and choice of models in tracking target with coordinated turn motion," in *2005 7th International Conference on Information Fusion*, vol. 2, 2005, pp. 6 pp.–.