September 17, 2020

# The Interacting Multiple Model Filter on Boxplus-Manifolds*

Tom L. Koller[1] and Udo Frese[1]

*Abstract*— **The interacting multiple model filter is the standard in state estimation where different dynamic models are required to model the behavior of a system. It performs a probabilistic mixing of estimates. Up to now, it is undefined how to perform this mixing properly on manifold spaces, e.g. quaternions. We present the proper probabilistic mixing on differentiable manifolds based on the boxplus-method. The result is the interacting multiple model filter on boxplus-manifolds. We prove that our approach is a first order correct approximation of the optimum. The approach is evaluated in a simulation and performs as good as the ad-hoc solution for quaternions. A generic implementation of the boxplus interacting multiple model filter for differentiable manifolds is published alongside with this paper.**

## I. Introduction

The interacting multiple model filter (IMM) is widely used in the field of target tracking. After its original invention [1] for radar based aircraft tracking [2], [3], [4], it has been used in various applications as attitude estimation [5], lane change prediction [6] and sensor fault detection [7].

The IMM is applied when a single dynamical model does not predict the behavior of the system accurately [8]. This is the case when the system dynamics depend on modes, i.e. discrete states that change abruptly. The original IMM runs one Kalman Filter (KF) per mode and fuses the estimates of the filters probabilistically based on the likelihood of the models. Up to now, several adaptations of the IMM have been published to use different nonlinear filters as the Extended Kalman Filter (EKF) [8], the Unscented Kalman Filter (UKF) [4] or the Particle Filter (PF) [9].

Typically, these nonlinear filters operate on vector spaces ($\mathbb{R}^N$). Thus, it is difficult to maintain manifold structures as the rotation quaternion in the filter. As it is often required to estimate orientations in target tracking, various extensions [10], [11], [12] have been developed to use rotation quaternions or matrices without destroying their manifold properties, e.g. unit norm or orthogonality. To our knowledge, no such extension exists for the IMM.

For quaternions, the approach to normalize the quaternion exists [5]. However, such an approach usually degrades the performance of the estimator due to erroneous covariance matrices. Besides that approach, publications avoid to use quaternions in the state by using 2D rather than 3D [6] or by using models that work in world coordinates only, as it is common in aircraft tracking [2]. In [13] the so called delta-quaternion is used in the state, while the quaternion orientation is predicted outside of the IMM.

We see a gap in the literature on how to handle manifolds in the IMM. Thus, we are motivated to develop an IMM which can handle manifolds properly.

The core difficulty of an IMM on manifolds is the probabilistic mixing of states. In the IMM the estimates of all filters are mixed in a weighted sum. Unfortunately, the operator + breaks the manifold structure, wherefore a sum cannot be computed [10]. Furthermore, the mixing degrades the covariance which may result in inconsistency.

To overcome this problem for single mode filtering on quaternions, the multiplicative EKF (MEKF)[11] or the error state KF (ESKF)[12] were developed. Both methods update the quaternion estimate by quaternion multiplication only, which sustains the manifold structure in contrast to addition. The boxplus-method ($\boxplus$-method) of Hertzberg et al. [10] generalizes this concept for manifolds. It only allows changes to the manifold which do not break its structure. It gained attention in pose tracking in the last years since it is a general approach to handle manifolds in nonlinear filtering [14], [15] and least squares optimization [16]. The $\boxplus$-method encapsulates manifolds as black boxes, so that algorithms can handle them generically. Furthermore, it provides the necessary definitions to calculate a weighted sum of Gaussians over manifolds as it is required for the IMM on manifolds.

The contribution of this paper is the theoretic derivation of an IMM which uses the $\boxplus$-method to properly handle the mixing of states and covariances in the manifold case. The proposed method is proved to perform a first order correct probabilistic mixing of Gaussians. The solution is a generalization of averaging rotations [17], since it applies to arbitrary $\boxplus$-manifolds.

The focus of this work is the theoretic extension of the $\boxplus$-method. The method is extended to hybrid estimation, which underlines the character of the $\boxplus$-method as a general solution to handle manifolds in state estimation.

The remainder of the paper is structured as follows: The theoretic foundation of probabilistic mixing on $\boxplus$-manifolds is shown in Section II. The IMM on $\boxplus$-manifolds is presented in Section III. In Section IV we give an example for the new algorithm and analyze its performance compared to a naive solution. At the end, we conclude and discuss future work.

## II. Weighted sum of Gaussians on $\boxplus$-manifolds

Usually, the representations of manifolds $\mathcal{S}$ are overparametrizations i.e., they are represented with more parameters than they have degrees of freedom (DOF). The key idea of the $\boxplus$-method is to allow changes of the manifolds only in the direction of the DOFs [10]. This direction is called the tangent space $\mathcal{V} \subset \mathbb{R}^{DOF}$. Small changes to the manifold instance can be expressed in the tangent space and are applied with the operator $\boxplus : \mathcal{S} \times \mathcal{V} \mapsto \mathcal{S}$. The $\boxplus$-operator enforces the manifold structure.

*DFG-Funding: ZaVI FR 2620/3-1

[1]Multi-Sensor Interactive Systems Group, University of Bremen, 28359 Bremen, Germany {tkoller,ufrese}@uni-bremen.de

The difference of two manifold instances is also expressed in the tangent space and can be derived by the complementary boxminus-operator $\boxminus : \mathcal{S} \times \mathcal{S} \mapsto \mathcal{V}$. The $\boxminus$-operator calculates the geodesic between two manifold instances, i.e. the shortest path between them on the manifold. The quadruplet $\{\mathcal{S}, \boxplus, \boxminus, \mathcal{V}\}$ is called a $\boxplus$-manifold. The operators for commonly used manifolds can be found in [10].

The $\boxplus$-method allows to compound a state of multiple manifolds and vectors. The $\boxplus/\boxminus$-operators of a compound manifold $x$ simply apply the operators elementwise:

$$x = \{x_1, \cdots, x_n\} \tag{1}$$

$$x \boxplus \delta = \{x_1 \boxplus \delta_1, \cdots, x_n \boxplus \delta_n\} \tag{2}$$

$$y \boxminus x = \{y_1 \boxminus x_1, \cdots, y_n \boxminus x_n\} \tag{3}$$

For vectors, the operators naturally reduce to $+/-$. Thus, the method allows to compound states of manifolds and vectors seamlessly, which is essential for target tracking applications.

Hertzberg et al. already stated that it is not possible to compute a weighted sum of manifolds with the classic definition [10]. Instead, they derived the implicit definition of the weighted sum using the expected value:

$$\mathrm{E}\left(X \boxminus \overline{X}\right) = 0 \tag{4}$$

where $\mathrm{E}(\cdots)$ computes the expected value, $X \subseteq \mathcal{S}$ is the set of all weighted manifold instances and $\overline{X}$ is the expected value of X, i.e. the weighted sum.

In the IMM, Gaussian distributions are mixed instead of simple instances. Thus, $X$ is a mixture of Gaussians $X_j = \mathcal{N}(\overline{x}_j, P_j)$ with mean $\overline{x}_j$ and covariance $P_j$. For manifolds the Gaussian is defined as:

$$\mathcal{N}(\mu, P) := \mu \boxplus \mathcal{N}(0, P) \tag{5}$$

where $P \in R^{DOFxDOF}$.

In the manifold case, the weighted sum of the mean values of the Gaussians $\overline{x}$ is not guaranteed to be the weighted sum $\overline{X}$ of the complete distribution. However, we prove that it is at least a first order correct approximation of $\overline{X}$.

*Theorem 1:* Let $X$ be a set of $M$ Gaussian Distributions $X_j$ with probabilities $p(X_j)$. Then, the weighted sum $\overline{x}$ of their mean values $\overline{x}_j$ is a first order correct approximation of the weighted sum $\overline{X}$ of all elements in $X$.

*Proof:*

$$\mathrm{E}(X \boxminus \overline{x}) \overset{?}{\approx} \mathrm{E}(X \boxminus \overline{X}) = 0 \tag{6}$$

$$\mathrm{E}(X \boxminus \overline{x}) = \int_X p(x) \cdot (x \boxminus \overline{x})\, dx,\ x \in X \tag{7}$$

$$= \sum_{j=1}^{M} \int_{X_j} p(x_j) \cdot (x_j \boxminus \overline{x})\, dx_j,\ x_j \in X_j \tag{8}$$

$$\int_{X_j} p(x_j) dx_j = p(X_j) \tag{9}$$

Each element $x_j$ can be expressed relative to the mean of the respective Gaussian using the axioms of $\boxplus$-manifolds [10]:

$$x_j = \overline{x}_j \boxplus \delta_j, \quad \delta_j = x_j \boxminus \overline{x}_j \tag{10}$$

$$\mathrm{E}(X \boxminus \overline{x}) = \sum_{j=1}^{M} \int_{X_j} p(x_j) \cdot ((\overline{x}_j \boxplus \delta_j) \boxminus \overline{x})\, dx_j \tag{11}$$

We approximate the $\boxplus$-operator with a first order Taylor series around $\delta_j = \vec{0}$:

$$\begin{aligned}\mathrm{E}(X \boxminus \overline{x}) \approx \sum_{j=1}^{M} \int_{X_j} p(x_j) \cdot \Big( \left(\overline{x}_j \boxplus \vec{0}\right) \boxminus \overline{x} \\ + J_j * (\delta_j - \vec{0}) \Big)\, dx_j\end{aligned} \tag{12}$$

$$J_j = \left. \frac{\partial\left((\overline{x}_j \boxplus \delta_j) \boxminus \overline{x}\right)}{\partial \delta_j} \right|_{\delta_j = \vec{0}} \tag{13}$$

Using (9) we can split into:

$$\begin{aligned}\mathrm{E}(X \boxminus \overline{x}) \approx \sum_{j=1}^{M} \Bigg( p(X_j)\,(\overline{x}_j \boxminus \overline{x}) \\ + J_j * \int_{X_j} p(x_j) \delta_j dx_j \Bigg)\end{aligned} \tag{14}$$

The first summand is the expected value $\mathrm{E}(\overline{x}_j - \overline{x})$ which is 0 by definition of $\overline{x}$.

$$\mathrm{E}(X \boxminus \overline{x}) \approx \sum_{j=1}^{M} J_j * \int_{X_j} p(x_j) \delta_j dx_j \tag{15}$$

Using the definition of $\overline{x}_j$:

$$\mathrm{E}(X_j \boxminus \overline{x}_j) = 0 = \int_{X_j} p(x_j)\,(x_j \boxminus \overline{x}_j)\, dx_j \tag{16}$$

$$= \int_{X_j} p(x_j) \delta_j dx_j \tag{17}$$

We can reduce to:

$$\mathrm{E}(X \boxminus \overline{x}) \approx 0 \tag{18}$$

Thus, the approximation is first order correct. ■

In the IMM, the mixed distribution is approximated with a Gaussian. Since the covariance of manifolds is expressed in the tangent space, the covariances cannot be summed up as in the original IMM. Instead, we use a first order propagation of the covariances.

The standard definition of covariance can be adapted to $\boxplus$-manifolds [10]. We express the covariance $P$ with respect to $\overline{x}$ since it is a first order correct approximation of the real mean of the weighted sum:

$$P = \mathrm{E}\left([x \boxminus \overline{x}]^{\otimes}\right) = \int_X p(x)\, [x \boxminus \overline{x}]^{\otimes}\, dx \tag{19}$$

where $[...]^{\otimes}$ is a short hand notation for the outer product of a vector with itself, i.e.:

$$[v]^{\otimes} = v \otimes v = vv^T \tag{20}$$

Again, each element of $X$ is expressed with respect to the mean of the corresponding Gaussian using (10):

$$P = \sum_{j=1}^{M} \int_{X_j} p(x_j) \left[ (\overline{x}_j \boxplus \delta_j) \boxminus \overline{x} \right]^{\otimes} dx_j \tag{21}$$

And linearise around $\delta_j = \vec{0}$ with $J_j$ as in (13):

$$P = \sum_{j=1}^{M} \int_{X_j} p(x_j) \left[ \left( \overline{x}_j \boxplus \vec{0} \right) \boxminus \overline{x} + J_j \delta_j \right]^{\otimes} dx_j \tag{22}$$

$$= \sum_{j=1}^{M} \int_{X_j} p(x_j) \left[ \overline{x}_j \boxminus \overline{x} + J_j \delta_j \right]^{\otimes} dx_j \tag{23}$$

By expansion we get:

$$\begin{aligned} P = \sum_{j=1}^{M} \int_{X_j} p(x_j) \Big( & \left[ \overline{x}_j \boxminus \overline{x} \right]^{\otimes} + \left[ J_j \delta_j \right]^{\otimes} \\ & + (\overline{x}_j \boxminus \overline{x}) (J_j \delta_j)^T + (J_j \delta_j) (\overline{x}_j \boxminus \overline{x})^T \Big) dx_j \end{aligned} \tag{24}$$

$$\begin{aligned} = \sum_{j=1}^{M} \int_{X_j} p(x_j) \Big( & \left[ \overline{x}_j \boxminus \overline{x} \right]^{\otimes} + \left( J_j \delta_j \delta_j^T J_j^T \right) \\ & + (\overline{x}_j \boxminus \overline{x}) \left( \delta_j^T J_j^T \right) + (J_j \delta_j) (\overline{x}_j \boxminus \overline{x})^T \Big) dx_j \end{aligned} \tag{25}$$

Which we can further reduce by rearranging the sums to:

$$\begin{aligned} P = \sum_{j=1}^{M} & p(X_j) \left[ \overline{x}_j \boxminus \overline{x} \right]^{\otimes} + p(X_j) J_j P_j J_j^T \\ & + (\overline{x}_j \boxminus \overline{x}) \int_{X_j} p(x_j) \delta_j^T dx_j * J_j^T \\ & + J_j \int_{X_j} p(x_j) \delta_j dx_j * (\overline{x}_j \boxminus \overline{x})^T \end{aligned} \tag{26}$$

Using (17) yields:

$$P = \sum_{j=1}^{M} p(X_j) \left( \left[ \overline{x}_j \boxminus \overline{x} \right]^{\otimes} + J_j P_j J_j^T \right) \tag{27}$$

Equation (27) is similar to the original equation of the IMM. The first summand expresses the spread of the mean in $\boxplus$-terms. The second summand propagates the covariances of the Gaussians to the new mean. The main difference to the original equation is the transformation of $P_j$ with the Jacobian $J_j$. In the vector case, the Jacobian equals identity.

## III. THE IMM ON $\boxplus$-MANIFOLDS

The IMM runs a recursive filter, e.g. an EKF or UKF, for each mode of the system. At every time step, it performs the three steps interaction, filtering and combination [8] (see TABLE I). The interaction mixes the estimates of all filters according to their mode and transition probabilities. The filtering performs the prediction and update of each filter. It calculates the new mode probabilities based on the measurements. The details of this step depend on the chosen filter type. The combination step only combines all estimates according to their mode probability to create the output of the IMM which is the most likely state of the system.

TABLE I

ORIGINAL IMM [8] VS $\boxplus$-IMM. BOXED LINES EXCHANGE THE PRIOR LINE TO FORM THE $\boxplus$-IMM. $z(k)$ MAY BE A $\boxplus$-MANIFOLD $\mathcal{D}$.

State, input, process and measurement models:

$$x_j(k) \in \mathcal{S},\ P_j(k) \in \mathbb{R}^{DOF \times DOF},\ u(k) \in \mathbb{R}^{\nu},\ z(k) \in \mathcal{D} \tag{28}$$

$$g_j : \mathcal{S} \times \mathbb{R}^{\nu} \times \mathbb{R}^{n} \mapsto \mathcal{S},\ X_j(k+1) = g_j(X_j(k), u(k), \epsilon_j(k)) \tag{29}$$

$$\epsilon_j(k) = \mathcal{N}_n(0, Q_j(k)),\ Q_j(k) \in \mathbb{R}^{n \times n} \tag{30}$$

$$h : \mathcal{S} \mapsto \mathcal{D},\ z(k) = h(X(k)) \boxplus \mathcal{N}_d(0, R_j(k)),\ R_j(k) \in \mathbb{R}^{d \times d} \tag{31}$$

Initialization: $\forall j \in [1, M]$

$$x_j(0) = x_0,\ P_j(0) = P_0,\ \mu_j(0) = \mu_{j0} \tag{32}$$

Interaction: $\forall i, j \in [1, M]$

$$c_j = \sum_{i=1}^{M} p_{ij} \mu_i(k-1) \tag{33}$$

$$\mu_{i|j}(k-1) = \frac{p_{ij} \mu_i(k-1)}{c_j},\ p_{ij} \text{ are transition probabilities.} \tag{34}$$

$$x_{0j}(k-1) = \sum_{i=1}^{M} x_i(k-1) \mu_{i|j}(k-1) \tag{35}$$

$$\boxed{x_{0j}(k-1) = \boxplus\text{-}\ \textbf{WeightedSum}(x_j(k-1), x_i(k-1), \mu_{i|j}(k-1))} \tag{36}$$

$$\begin{aligned} P_{0j}(k-1) = \sum_{i=1}^{M} \mu_{i|j}(k-1) \Big( & P_i(k-1) \\ & + \left[ x_i(k-1) - x_{0j}(k-1) \right]^{\otimes} \Big) \end{aligned} \tag{37}$$

$$\boxed{\begin{aligned} P_{0j}(k-1) = \boxplus\text{-}\ & \textbf{WeightedCovarianceSum}(x_{0j}(k-1), \\ & P_i(k-1), x_i(k-1), \mu_{i|j}(k-1)) \end{aligned}} \tag{38}$$

Filtering ($\boxplus$-EKF): $\forall j \in [1, M]$

Prediction:

$$\hat{x}_j(k) = g(x_{0j}(k-1), u(k-1), \vec{0}) \tag{39}$$

$$F_j(k-1) = \left. \frac{\partial g(x, u(k-1), \vec{0})}{\partial x} \right|_{x = x_{0j}(k-1)} \tag{40}$$

$$U_j(k-1) = \left. \frac{\partial g(x_{0j}(k-1), u(k-1), \epsilon)}{\partial \epsilon} \right|_{\epsilon = \vec{0}} \tag{41}$$

$$\begin{aligned} \hat{P}_j(k) = & F_j(k-1) P_{0j}(k-1) F_j(k-1)^T \\ & + U_j(k-1) Q_j(k-1) U_j(k-1)^T \end{aligned} \tag{42}$$

Update:

$$H_j(k) = \left. \frac{\partial h(x)}{\partial x} \right|_{x = \hat{x}_j(k)} \tag{43}$$

$$S_j(k) = H_j(k) \hat{P}_j(k) H_j(k)^T + R_j(k) \tag{44}$$

$$W_j(k) = \hat{P}_j(k) H_j(k)^T S_j(k)^{-1} \tag{45}$$

$$r_j(k) = z(k) \boxminus h(\hat{x}_j(k)) \tag{46}$$

$$x_j(k) = \hat{x}_j(k) \boxplus W_j(k) r_j(k) \tag{47}$$

$$P_j(k) = \hat{P}_j(k) - W_j(k) S_j(k) W_j(k)^T \tag{48}$$

$$\Lambda_j(k) = N(r_j(k); 0, S_j(k)) \tag{49}$$

$$\mu_j(k) = \frac{1}{c} \Lambda_j(k) c_j,\ c \text{ is a normalization factor} \tag{50}$$

Combination:

$$x(k) = \sum_{j=1}^{M} x_j(k) \mu_j(k) \tag{51}$$

$$\boxed{x(k) = \boxplus\text{-}\ \textbf{WeightedSum}(x(k-1), x_j(k), \mu_j(k))} \tag{52}$$

$$P(k) = \sum_{j=1}^{M} \mu_j(k) \left( P_j(k) + \left[ x_j(k) - x(k) \right]^{\otimes} \right) \tag{53}$$

$$\boxed{P(k) = \boxplus\text{-}\ \textbf{WeightedCovarianceSum}(x(k), P_j(k), x_j(k), \mu_j(k))} \tag{54}$$

To use the IMM on $\boxplus$-manifolds two changes are applied:

1) The inner filter must be able to handle $\boxplus$-manifolds.
2) The weighted sum of Gaussians in the interaction and combination step needs to be calculated with the $\boxplus$-equivalents.

The first change can be introduced by using the $\boxplus$-EKF [14], [15] or $\boxplus$-UKF [10]. The measurement likelihood can be calculated using Gaussians on $\boxplus$-manifolds.

The second change has to be introduced with the results from Section II. Using Theorem 1, we approximate the mean of the weighted sum of Gaussians with the weighted sum of the means of the Gaussians. The weighted sum can be calculated using the iterative algorithm **$\boxplus$-WeightedSum** adapted from [10]:

$$\text{Input: } \overline{X}_0, \overline{x}_j, p(X_j) \quad \forall j \in [1, M] \tag{55}$$

$$\overline{X}_{k+1} = \overline{X}_k \boxplus \sum_{j=1}^{M} p(X_j)(\overline{x}_j \boxminus \overline{X}_k) \tag{56}$$

$$\overline{x} = \lim_{k \mapsto \infty} \overline{X}_k \tag{57}$$

In practice, the iteration can be stopped when the change of the calculated mean is small. The convergence speed depends greatly on the choice of the initial guess $\overline{X}_0$. The algorithm is identical to the computation of the mean on compact lie groups [18].

With the **$\boxplus$-WeightedSum** algorithm the mode estimates can be mixed without destroying the manifold structure. In the IMM (35) and (51) have to be exchanged with (36) and (52) respectively.

Following (27), the mixed covariance can be calculated using the function **$\boxplus$-WeightedCovarianceSum**:

$$\text{Input: } \overline{x}, P_j, \overline{x}_j, p(X_j) \quad \forall j \in [1, M] \tag{58}$$

$$J_j = \left. \frac{\partial \left(\overline{x}_j \boxplus \delta \boxminus \overline{x}\right)}{\partial \delta} \right|_{\delta = \vec{0}} \tag{59}$$

$$P = \sum_{j=1}^{M} p(X_j) \left( [\overline{x}_j \boxminus \overline{x}]^{\otimes} + J_j P_j J_j^T \right) \tag{60}$$

With the **$\boxplus$-WeightedCovarianceSum** function the covariances can be mixed properly. To apply it to the IMM (37) and (53) need to be exchanged with (38) and (54) respectively.

This results in a generic IMM that properly mixes the state estimates based on the $\boxplus$-method: The $\boxplus$-IMM. It does not require any ad-hoc implementation to mix the states as it only uses the $\boxplus$/$\boxminus$-interface of the manifold.

## IV. Example Application and Performance Discussion

We test the $\boxplus$-IMM in a simulated environment, to provide first insights into the performance of the new algorithm. We choose the following setup inspired by classic radar tracking: A drone flies across known terrain. It has an stereo-camera facing downwards. With the camera it detects known landmarks in the terrain. The task is to track the position of the drone. Since the camera is mounted on the drone, its measurements are in body coordinates. Hence, it is required to estimate the orientation of the drone to make use of the measurements.

The drone has two different flight modes. In the first mode it flies straight with a constant velocity. In the second mode it flies a curve with a constant angular rate.

We model the dynamics of the drone with the state $x$:

$$x = (q_b^w \quad \vec{p}_w \quad \vec{v}_w \quad \vec{\omega}_w)^T \tag{61}$$

where $q_b^w$ is the rotation quaternion that rotates a world frame vector to body frame, $\vec{p}_w$ is the position in world frame, $\vec{v}_w$ is the velocity in world frame and $\vec{\omega}_w$ is the angular rate in world frame. The straight dynamic is modeled as:

$$g_s(x(k), \epsilon_s(k)) = \begin{pmatrix} q_b^w \\ \vec{p}_w + (\vec{v}_w + \epsilon_s)\Delta t \\ \vec{v}_w + \epsilon_s \\ \vec{\omega}_w \end{pmatrix} \tag{62}$$

where $\Delta t = 0.05\,s$ is the time difference between time $k$ and $k+1$. The constant turn dynamic is modeled as in [2] with a change for the orientation and angular rate:

$$g_c(x(k), \epsilon_c(k)) = \begin{pmatrix} q_b^w * \exp(\frac{\Delta t}{2}\vec{\omega}_w)^{-1} \\ \vec{p}_w + (\Delta t I_{3\times 3} + B)\vec{v}_w \\ (I_{3\times 3} + A)\vec{v}_w \\ \vec{\omega}_w + \epsilon_c \end{pmatrix} \tag{63}$$

where $\exp(\cdots)$ forms a quaternion from the given Euler-angle-axis [10] and $A, B \in \mathbb{R}^{3\times 3}$ as given in [2].

For simplification, we assume that the camera measures the position of the landmark in body coordinates. The measurement model is:

$$h(x) = q_b^w * \vec{p}_w * (q_b^w)^{-1} \tag{64}$$

The covariance matrices can be found in the Appendix.

The simulated drone flies a trajectory of two straights and two $180°$ curves over four visible landmarks. It is evaluated against the Naive-IMM with naive mixing, i.e. the quaternions are averaged in parameter space and normalized after. The covariances are summed up as in the original IMM. However, the inner filter is a $\boxplus$-EKF as well since we only want to evaluate the effect of the $\boxplus$-mixing of states on the estimation performance. Furthermore, the algorithm is compared against a $\boxplus$-EKF on the constant turn model, to show the overall benefit of the IMM. The root mean squared error (RMSE) to the ground truth is used for comparison.

The RMSE of the $\boxplus$-IMM and Naive-IMM are shown in Table II. The developed $\boxplus$-IMM has the same RMSE as the Naive-IMM. Thus, the $\boxplus$-mixing of state estimates does not improve the estimation accuracy.

TABLE II

RMSE comparison for aircraft tracking

| EKF | $\boxplus$-IMM | Naive-IMM | $\boxplus$/Naive-IMM diff. |
|---|---|---|---|
| 0.502367 | 0.488076 | 0.488084 | -7.79736e-06 |

This result is unsatisfying since the $\boxplus$-mixing should yield better accuracy and consistency. However, the error of the naive mixing is negligible in the presented example.

We try to quantify the error induced by naive mixing. We calculate the weighted mean of two quaternion Gaussian

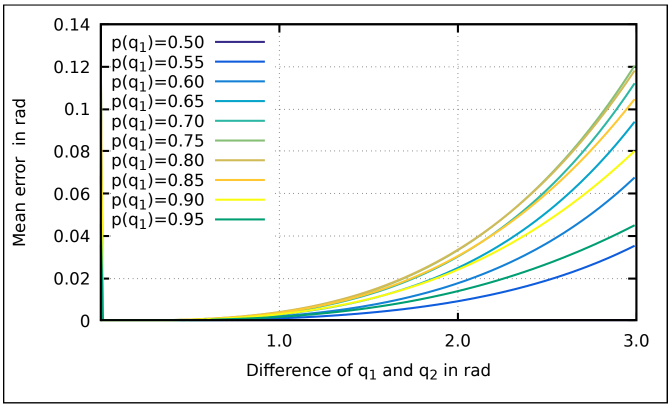


Fig. 1. The difference of means ($\boxminus$-norm) between $\boxplus$- and naive mixing over the angular differences of $q_1$ and $q_2$.

distributions $q_1, q_2$ for different angular differences and different probabilities (see Fig. 1).

The difference is in the range of $10^{-4}$ rad for angular differences below $0.35$ rad (ca. $20°$), for all probabilities. Since the IMM usually operates at small differences between the models, the error of the naive mixing is negligible for the mean.

Similarly, the effect of the $\boxplus$-mixing on the covariance is small (see Fig. 2). Hence, the two mixing methods differ only for high differences of the mixed quaternions. In the presented simulation example, the angular differences are small, wherefore the mixing methods have equal results.

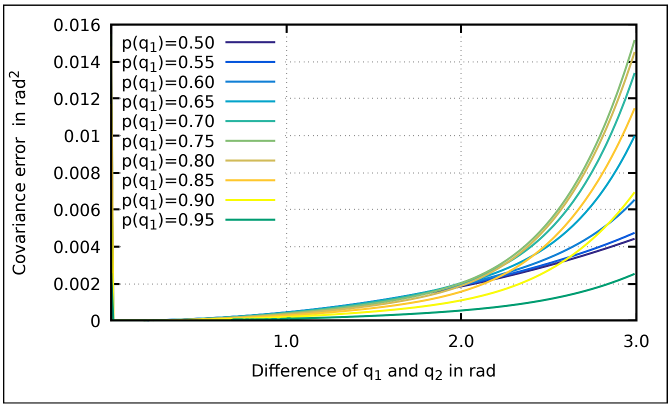


Fig. 2. The covariance difference (Schur norm) between $\boxplus$- and naive mixing over the angular differences of $q_1$ and $q_2$.

In general, it is unlikely that the quaternion estimates of the IMM differ greatly. The mixing is always performed after the update step. Hence, even big differences of the dynamic models are compensated by the update.

The $\boxplus$-mixing may perform better for higher differences. However, its original purpose is subverted. With an increasing difference between the quaternions the linearization errors also increase since it is only first order correct.

To show the increasing error, $\boxplus$- and naive mixing are compared to an optimal solution in Fig. 3 and Fig. 4. Since their is no closed form solution to mix the Gaussians, the optimal solution is obtained numerically. Quaternions are sampled uniformly from the Gaussians to approximate the complete distribution.

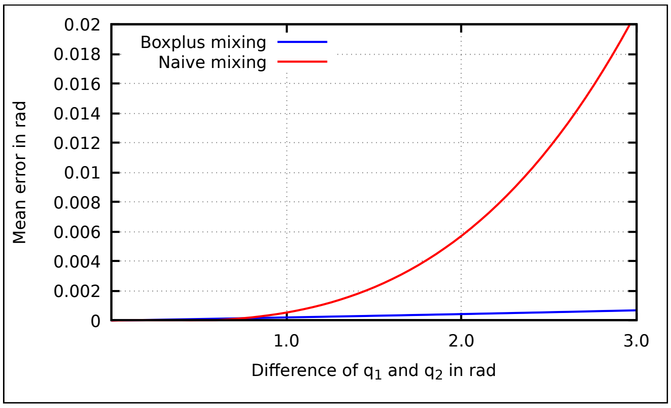


Fig. 3. The error of means ($\boxminus$-norm) of $\boxplus$- and naive mixing compared to optimal mixing over the angular differences of $q_1$ and $q_2$.

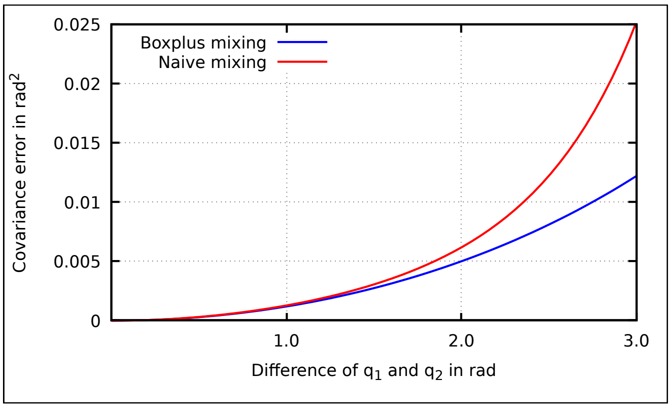


Fig. 4. The covariance error (Schur norm) of $\boxplus$- and naive mixing compared to optimal mixing over the angular differences of $q_1$ and $q_2$.

The mean and covariance error of the $\boxplus$-mixing increase with the distance between the Gaussian means. The $\boxplus$-mixing outperforms the naive mixing at higher differences between the two quaternions. Still, the approaches are almost equal for small differences.

Overall, using the $\boxplus$-mixing with a first order approximation does not give a performance boost for the IMM on quaternions. Instead, it consumes more computational power since it requires an iterative calculation of the mean and the calculation of additional Jacobians.

The $\boxplus$-mixing does not improve the performance of the IMM, but it enables a generic IMM on differentiable $\boxplus$-manifolds. The method encapsulates the manifold properties of the state so that it can be treated as a black box. Therefore, the IMM can be implemented independently of the used state representation. It does not require any ad-hoc solutions to mix the states. An open source C++ implementation of the generic $\boxplus$-IMM is provided at: `https://github.com/TomLKoller/Boxplus-IMM`. It uses automatic differentiation [19] to calculate all required Jacobians for state mixing and for the internal $\boxplus$-EKF [20]. It simplifies the use of the IMM immensely, as the developer can focus on tuning the dynamic models without taking care of Jacobians or the mixing step. The repository also contains the presented simulation example.

## V. Conclusion

The first order correct IMM on $\boxplus$-manifolds has been derived. Proofs are provided, how the weighted mean and covariance of mixtures of Gaussians on $\boxplus$-manifolds are calculated. With these, the $\boxplus$-method is applied to the IMM.

The $\boxplus$-IMM has been evaluated on a simulated aircraft tracking example. The evaluation of the algorithm shows, that the accuracy of the IMM is not improved compared to a naive approach of mixing quaternions. Thus, it is shown that mixing in the parameter space of quaternions followed by a normalization, is a simple, but suitable way to handle the quaternion manifold structure in the IMM.

The $\boxplus$-IMM still has high theoretical value as it can be directly derived from the basic definitions of the expected value and the covariance on $\boxplus$-manifolds. Thus, it is a justified algorithm instead of an ad-hoc solution.

This paper extended the family of $\boxplus$-algorithms to the the IMM. The presented IMM is fully generic, since the $\boxplus$-method encapsulates the manifold properties and separates them from the algorithm. No further ad-hoc implementations are required to perform the mixing, regardless of the state. Therefore, the $\boxplus$-IMM enables the implementation of a generic IMM library that can handle $\boxplus$-manifold states. A first prototype is published alongside this paper.

The presented method is only first order correct. Thus, one may develop higher order methods or use an UKF style method to mix the Gaussians. Presumably, this will not reduce the error, as the error compared to the numerical solution was quite low for small distances anyway. Instead, it should be investigated whether the proposed method has a visible advantage on other $\boxplus$-manifolds as the rotation matrix. This may be the case, since the normalization of quaternions is quite simple in comparison.

## Appendix

Covariances of dynamic models:

$$Q_s(k) = \begin{pmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{pmatrix},\ Q_c(k) = \begin{pmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.1 \end{pmatrix}$$

Covariance of Measurement and transition probabilities:

$$R(k) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},\ p_{trans} = \begin{pmatrix} 0.95 & 0.05 \\ 0.05 & 0.95 \end{pmatrix}$$

## References


[1] H. A. P. Blom, “An efficient filter for abruptly changing systems”, in The 23rd IEEE Conference on Decision and Control, Las Vegas, Nevada, USA, USA, Dec. 1984, pp. 656–658, doi: 10.1109/CDC.1984.272089.

[2] Ronghui Zhan and Jianwei Wan, “Passive maneuvering target tracking using 3D constant-turn model”, in 2006 IEEE Conference on Radar, Apr. 2006, p. 8, doi: 10.1109/RADAR.2006.1631832.

[3] J. D. Glass, W. D. Blair, and Y. Bar-Shalom, “IMM estimators with unbiased mixing for tracking targets performing coordinated turns”, in 2013 IEEE Aerospace Conference, Mar. 2013, pp. 1–10, doi: 10.1109/AERO.2013.6496912.

[4] W. Zhu, W. Wang, and G. Yuan, “An Improved Interacting Multiple Model Filtering Algorithm Based on the Cubature Kalman Filter for Maneuvering Target Tracking”, Sensors, vol. 16, no. 6, Art. no. 6, Jun. 2016, doi: 10.3390/s16060805.

[5] C.-D. Wann and J.-H. Gao, “Orientation Estimation for Sensor Motion Tracking Using Interacting Multiple Model Filter”, IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences, vol. E93-A, no. 8, pp. 1565–1568, Aug. 2010.

[6] Jiang Liu, Baigen Cai, J. Wang, and Wei Shangguan, “GNSS/INS-based vehicle lane-change estimation using IMM and lane-level road map”, in 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013), Oct. 2013, pp. 148–153, doi: 10.1109/ITSC.2013.6728225.

[7] N. Sadeghzadeh-Nokhodberiz and J. Poshtan, “Distributed Interacting Multiple Filters for Fault Diagnosis of Navigation Sensors in a Robotic System”, IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 47, no. 7, pp. 1383–1393, Jul. 2017, doi: 10.1109/TSMC.2016.2598782.

[8] 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, Jan. 1998, doi: 10.1109/7.640267.

[9] Y. Boers and J. N. Driessen, “Interacting multiple model particle filter”, Sonar and Navigation IEEE Proceedings - Radar, vol. 150, no. 5, pp. 344–349, Oct. 2003, doi: 10.1049/ip-rsn:20030741.

[10] C. Hertzberg, R. Wagner, U. Frese, and L. Schröder, “Integrating Generic Sensor Fusion Algorithms with Sound State Representations through Encapsulation of Manifolds”, Information Fusion, vol. 14, no. 1, pp. 57–77, 2013.

[11] F. L. Markley, “Attitude Error Representations for Kalman Filtering”, Journal of Guidance, Control, and Dynamics, vol. 26, no. 2, pp. 311–317, 2003, doi: 10.2514/2.5048.

[12] V. Madyastha, V. Ravindra, S. Mallikarjunan, and A. Goyal, “Extended Kalman Filter vs. Error State Kalman Filter for Aircraft Attitude Estimation”, in AIAA Guidance, Navigation, and Control Conference, American Institute of Aeronautics and Astronautics, Jun. 2012.

[13] H. Himberg, Y. Motai, and A. Bradley, “A Multiple Model Approach to Track Head Orientation With Delta Quaternions”, IEEE Transactions on Cybernetics, vol. 43, no. 1, pp. 90–101, Feb. 2013, doi: 10.1109/TSMCB.2012.2199311.

[14] J. Clemens and K. Schill, “Extended Kalman filter with manifold state representation for navigating a maneuverable melting probe”, in 2016 19th International Conference on Information Fusion (FUSION), Jul. 2016, pp. 1789–1796.

[15] D. Nakath, J. Clemens, and K. Schill, “Multi-Sensor Fusion and Active Perception for Autonomous Deep Space Navigation”, in 2018 21st International Conference on Information Fusion (FUSION), Jul. 2018, pp. 2596–2605, doi: 10.23919/ICIF.2018.8455788.

[16] T. L. Koller and U. Frese, “State Observability through Prior Knowledge: Analysis of the Height Map Prior for Track Cycling”, Sensors, vol. 20, no. 9, Art. no. 9, Jan. 2020, doi: 10.3390/s20092438.

[17] R. Hartley, J. Trumpf, Y. Dai, and H. Li, “Rotation Averaging”, International Journal of Computer Vision, vol. 103, no. 3, pp. 267–305, Jul. 2013, doi: 10.1007/s11263-012-0601-0.

[18] J. H. Manton, “A globally convergent numerical algorithm for computing the centre of mass on compact Lie groups”, in ICARCV 2004 8th Control, Automation, Robotics and Vision Conference, 2004., Dec. 2004, vol. 3, pp. 2211-2216 Vol. 3, doi: 10.1109/ICARCV.2004.1469774.

[19] P. H. W. Hoffmann, “A Hitchhiker’s Guide to Automatic Differentiation”, Numerical Algorithms, vol. 72, no. 3, pp. 775–811, Jul. 2016, doi: 10.1007/s11075-015-0067-6.

[20] L. Post and T.L. Koller, ”An Automatically Differentiating Extended Kalman Filter on Boxplus Manifolds”, 2020 `https://github.com/TomLKoller/ADEKF`