

Article

The Interacting Multiple Model Filter and Smoother on Boxplus-Manifolds [†]

Tom L. Koller *  and Udo Frese 

Multi-Sensor Interactive Systems Group, University of Bremen, 28359 Bremen, Germany; ufrese@informatik.uni-bremen.de

* Correspondence: tkoller@uni-bremen.de; Tel.: +49-421-218-64202

† This paper is an extended version of our published in Koller, T.L.; Frese, U. The Interacting Multiple Model Filter on Boxplus-Manifolds. In Proceedings of the 2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Karlsruhe, Germany, 14–16 September 2020.

Abstract: Hybrid systems are subject to multiple dynamic models, or so-called modes. To estimate the state, the sequence of modes has to be estimated, which results in an exponential growth of possible sequences. The most prominent solution to handle this is the interacting multiple model filter, which can be extended to smoothing. In this paper, we derive a novel generalization of the interacting multiple filter and smoother to manifold state spaces, e.g., quaternions, based on the boxplus-method. As part thereof, we propose a linear approximation to the mixing of Gaussians and a Rauch–Tung–Striebel smoother for single models on boxplus-manifolds. The derivation of the smoother equations is based on a generalized definition of Gaussians on boxplus-manifolds. The three, novel algorithms are evaluated in a simulation and perform comparable to specialized solutions for quaternions. So far, the benefit of the more principled approach is the generality towards manifold state spaces. The evaluation and generic implementations are published open source.



Citation: Koller, T.L.; Frese, U. The Interacting Multiple Model Filter and Smoother on Boxplus-Manifolds. *Sensors* **2021**, *21*, 4164. <https://doi.org/10.3390/s21124164>

Academic Editors: Sukhan Lee, Uwe D. Hanebeck and Florian Pfaff

Received: 13 April 2021

Accepted: 12 June 2021

Published: 17 June 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: IMM; RTS; smoothing; manifolds; hybrid estimation; orientation estimation; quaternion smoothing

1. 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 [2–4], it has been used in various applications such 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 such as the Extended Kalman Filter (EKF) [8], the Unscented Kalman Filter (UKF) [4], or the Particle Filter (PF) [9].

The accuracy of IMM estimates can be improved by smoothing [10]. Smoothing improves the past estimate by propagating the information of measurements backwards [11]. Retrospectively, it uses measurements from the past and future. Similarly to the IMM filter, the IMM smoother uses a smoother for each mode and fuses their estimates probabilistically. Several mixing schemes already exist and further schemes are being developed [12,13].

Typically, nonlinear filters and smoothers are designed to operate on vector spaces (\mathbb{R}^n). Thus, it is difficult to maintain manifold structures like the rotation quaternion in the filter. As it is often required to estimate orientations in target tracking, various extensions [14–17] have been developed to handle rotation quaternions or matrices correctly without destroying

their manifold properties, e.g., unit norm or orthogonality. To our knowledge, no such extension exists for the IMM nor the IMM smoother.

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 using quaternions in the state by using 2D rather than 3D [6] or by using models that work in world coordinates only, as is common in aircraft tracking [2]. In [18], 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 smoother in a principled way. This paper closes the gap by proposing an IMM smoother that can handle manifolds properly. While we use quaternions as an accessible application example, we designed our novel algorithms to handle more general manifolds—e.g., a sphere.

The core difficulty of an IMM smoother 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 [14]. Similar problems appear for the covariance, which conceptually assumes additive noise on the state.

To overcome this problem for single-mode filtering on quaternions, the multiplicative EKF (MEKF) [15] or the error state KF (ESKF) [16] 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. [14] 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 [19,20] and least squares optimization [21]. 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.

This paper is an extended version of [22], where we derived an IMM filter that uses the \boxplus -method to properly handle the mixing of states and covariances in the manifold case. The method has been proven to perform a first-order correct probabilistic mixing of Gaussians. The additional contribution of this paper is the extension of the IMM filter on \boxplus -manifolds to an IMM smoother. To do so, we extend the \boxplus -method to Rauch–Tung–Striebel (RTS) smoothing [23].

In this work, we focus on the tangent spaces of manifolds to derive the RTS smoother equations and reinterpret the mixing of covariances with tangent space transforms. To achieve this, we propose a generalized definition of Gaussians on \boxplus -manifolds, which clarifies how to transform Gaussians between different tangent spaces. The rigorous consideration of tangent space transformations results in a generic Extended Kalman smoother (EKS) on \boxplus -manifolds. The combination of the novel EKS and the probabilistic mixing of Gaussians of [22] directly yields the IMM smoother on \boxplus -manifolds based on the scheme of [11].

The remainder of the paper is structured as follows: The theoretic foundation of probabilistic mixing on \boxplus -manifolds with a new derivation of the covariance mixing based on our generalized definition of Gaussians is shown in Section 2. The resulting IMM Filter on \boxplus -manifolds is presented in Section 3. As a basis for the IMM smoother, an RTS smoother on \boxplus -manifolds is derived in Section 4. The novel IMM smoother is shown in Section 5. In Section 6, we give a simulation example for the new algorithms and analyze their performance compared to state of the art approaches. Finally, we conclude and discuss future work in Section 7.

2. Weighted Sum of Gaussians on \boxplus -Manifolds

Usually, the representation of a manifold \mathcal{S} is overparametrized, i.e., represented with more parameters than it has 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 [14]. This direction is called the tangent space $\mathcal{V} \subset \mathbb{R}^{DOF}$. The direction of the DOFs depends on the specific

instance of the manifold. Thus, the tangent space is always defined with respect to a reference instance. 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.

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 ($x \boxplus (y \boxminus x) = y$). The quadruplet $\{\mathcal{S}, \boxplus, \boxminus, \mathcal{V}\}$ is called a \boxplus -manifold. The operators for commonly used manifolds can be found in [14].

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

$$x = \{x_1, \dots, x_n\}, \quad (1)$$

$$x \boxplus \delta = \{x_1 \boxplus \delta_1, \dots, x_n \boxplus \delta_n\}, \quad (2)$$

$$y \boxminus x = \{y_1 \boxminus x_1, \dots, y_n \boxminus x_n\}. \quad (3)$$

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.

One axiomatic property of \boxplus -manifolds is that every instance y of the manifold can be represented in the tangent space of any other instance x by $y = x \boxplus (y \boxminus x)$ [14]. Thus, all operations can be defined with respect to the tangent space of x although singularities may arise if x and y are distant. Since the tangent space is a vector space, standard definitions and operations can be used. By choosing the tangent space appropriately, troublesome singularities are avoided. Essentially, this is the engineering "trick" of the \boxplus -method.

Hertzberg et al. already stated that it is not possible to compute an expected value or a weighted sum of manifolds with the classic definition since the $+$ -operator is undefined [14]. Instead, they derived an implicit definition of the expected value:

$$E(X \boxminus \bar{X}) = 0, \quad (4)$$

where $E(\cdot)$ computes the expected value, X is a random variable on \mathcal{S} , and \bar{X} is the expected value of X . Essentially, $X \boxminus \bar{X}$ expresses all manifolds elements in the tangent space of \bar{X} , which allows us to apply the standard computation of the expected value.

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

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

where $P \in \mathbb{R}^{DOF \times DOF}$. The key of this definition is that the covariance of the Gaussian is defined with respect to the tangent space of the mean. Hence, if two Gaussians have different means, their covariances refer to different tangent spaces.

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

Theorem 1. *Let X be the mixture of M normally distributed random variables $X_j \sim \mathcal{N}(\bar{x}_j, P_j)$ with probabilities p_j , where $X \sim \sum_{i=1}^M p_j \mathcal{N}(\bar{x}_j, P_j)$. Then, the weighted sum \bar{x} of the expected values \bar{x}_j (defined implicitly by $\sum_{i=1}^M p_j (\bar{x}_j \boxminus \bar{x}) = 0$) is a first-order correct approximation of the expected value \bar{X} of all elements in X .*

Proof.

$$E(X \boxminus \bar{x}) \stackrel{?}{\approx} E(X \boxminus \bar{X}) = 0, \quad (6)$$

$$E(X \boxminus \bar{x}) = \int p(x) \cdot (x \boxminus \bar{x}) dx = \sum_{j=1}^M \int p(x_j) \cdot (x_j \boxminus \bar{x}) dx_j, \quad (7)$$

$$\int p(x_j) dx_j = p_j, \quad (8)$$

where we use the conventions $p(x) = p(X = x)$ and $p(x_j) = p(X_j = x_j)$. Each element x_j can be expressed in the tangent space of the respective mean \bar{x}_j using the axioms of \boxplus -manifolds [14]:

$$x_j = \bar{x}_j \boxplus \delta_j, \quad \delta_j = x_j \boxminus \bar{x}_j, \quad (9)$$

$$E(X \boxminus \bar{x}) = \sum_{j=1}^M \int p(x_j) \cdot ((\bar{x}_j \boxplus \delta_j) \boxminus \bar{x}) dx_j. \quad (10)$$

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

$$E(X \boxminus \bar{x}) \approx \sum_{j=1}^M \int p(x_j) \cdot \left((\bar{x}_j \boxplus \vec{0}) \boxminus \bar{x} + J_j \cdot (\delta_j - \vec{0}) \right) dx_j, \quad (11)$$

$$J_j = \left. \frac{\partial((\bar{x}_j \boxplus \delta_j) \boxminus \bar{x})}{\partial \delta_j} \right|_{\delta_j = \vec{0}}. \quad (12)$$

Using Equation (8) we can split into

$$E(X \boxminus \bar{x}) \approx \sum_{j=1}^M \left(p_j (\bar{x}_j \boxminus \bar{x}) + J_j \cdot \int p(x_j) \delta_j dx_j \right). \quad (13)$$

The first summand is 0 by the implicit definition of \bar{x} .

$$E(X \boxminus \bar{x}) \approx \sum_{j=1}^M J_j \cdot \int p(x_j) \delta_j dx_j. \quad (14)$$

Using the definition of \bar{x}_j :

$$E(X_j \boxminus \bar{x}_j) = 0 = \int p(x_j) (x_j \boxminus \bar{x}_j) dx_j = \int p(x_j) \delta_j dx_j. \quad (15)$$

We can reduce to

$$E(X \boxminus \bar{x}) \approx 0. \quad (16)$$

Thus, the approximation is first-order correct. \square

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 mixed as in the original IMM. First, the Gaussians have to be expressed with respect to the same tangent space.

Naturally, Gaussians on \boxplus -manifolds are defined in the tangent space of their expected value, as in (5). However, we can retrieve a more general definition of the Gaussian by defining it in the tangent space of an arbitrary reference r :

$$\mathcal{N}_r(E_r(X), \text{Cov}_r(X)) = r \boxplus \mathcal{N}(E_r(X), \text{Cov}_r(X)), \quad (17)$$

$$E_r(X) = E(X \boxminus r) = \int (x \boxminus r) \cdot p(x) dx, \quad (18)$$

$$\text{Cov}_r(X) = \text{Cov}(X \boxminus r) = \int ((x \boxminus r) - E(X \boxminus r))((x \boxminus r) - E(X \boxminus r))^T \cdot p(x) dx. \quad (19)$$

Choosing $r = E(X)$ directly yields the definitions of [14]. The standard computation rules for the expectation and covariance do not necessarily hold for this generalization, e.g., $E_r(X) + E_r(Y) \neq E_r(X + Y)$, since the $+$ -operator is undefined on the manifolds.

With the new definition, the reference can be arbitrarily chosen as in Figure 1. To change the reference of the Gaussian from r_1 to r_2 , the Gaussian has to be transformed to the manifold space and afterwards, to the tangent space of r_2 :

$$E_{r_2}(X) = E((r_1 \boxplus \mathcal{N}(E_{r_1}(X), \text{Cov}_{r_1}(X))) \boxminus r_2), \quad (20)$$

$$\text{Cov}_{r_2}(X) = \text{Cov}((r_1 \boxplus \mathcal{N}(E_{r_1}(X), \text{Cov}_{r_1}(X))) \boxminus r_2). \quad (21)$$

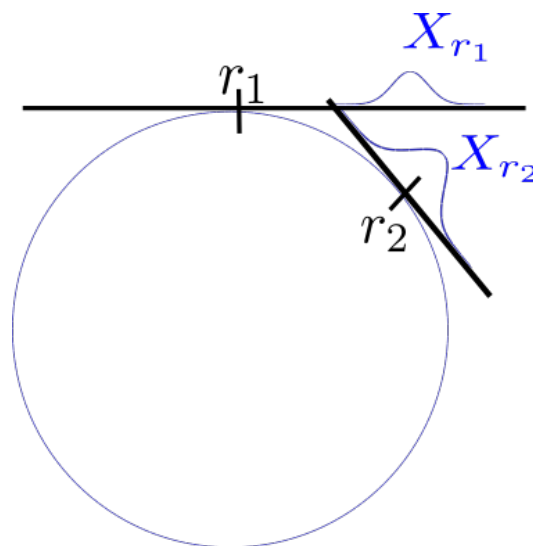


Figure 1. The Gaussian distribution X approximated in two tangent spaces on a unit circle manifold.

As usual, no general closed form solution exists, since the operators are nonlinear. A first-order Taylor approximation at $E_{r_1}(X)$ yields

$$E_{r_2}(X) \approx (r_1 \boxplus E_{r_1}(X)) \boxminus r_2, \quad (22)$$

$$\text{Cov}_{r_2}(X) \approx J \text{Cov}_{r_1}(X) J^T, \quad (23)$$

$$J = \frac{\partial((r_1 \boxplus (E_{r_1}(X) + \delta)) \boxminus r_2)}{\partial \delta}. \quad (24)$$

To retrieve the representation in Equation (5), the expected value is computed in the manifold space:

$$E(X) \approx r_2 \boxplus ((r_1 \boxplus E_{r_1}(X)) \boxminus r_2) = r_1 \boxplus E_{r_1}(X). \quad (25)$$

Two major types of reference transformations occur in state estimation. The first type is the reference transform from a Gaussian expressed in the tangent space of r_1 to the tangent space of its expected value r_2 . This reference transform is required to reestablish the standard form of Gaussians on \boxplus -manifolds, as defined in [14]. We call this type

“centered transform”, since it transforms the Gaussian to the tangent space of its center, i.e., $r_2 = r_1 \boxplus E_{r_1}(X)$.

The second type transforms a Gaussian that is expressed in the tangent space of its expected value, i.e., $E_{r_1}(X) = \bar{0}$, to another reference r_2 . This transform is required to express two Gaussians in the same tangent space to allow further computations. We name it “displaced transform”.

Using displaced transforms, all Gaussians of a mixture can be expressed in the tangent space of \bar{x} . Then, the standard covariance mixing can be applied, which results in

$$P = \sum_{j=1}^M p(X_j) \left([\bar{x}_j \boxminus \bar{x}]^{\otimes} + J_j P_j J_j^T \right), \quad (26)$$

where $[\cdot]^{\otimes}$ denotes the outer product and J_j as in (12). The same equation can be derived by linearization of the covariance equation like in our conference paper [22].

The first summand of (26) expresses the spread of the means in \boxplus -terms. The second summand propagates the covariances of the Gaussians to the new mean. The structural difference to the original equation is the transformation of P_j with the Jacobian J_j . In the vector case, the Jacobian equals identity. Thus, this formula is a generalization of the Gaussian mixing of vectors to \boxplus -manifolds.

3. An IMM Filter on \boxplus -Manifolds

The IMM runs a filter, e.g., an EKF or UKF, for each mode of the system. At every time step, it performs three steps: interaction, filtering, and combination [8]. 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 combines all estimates according to their mode probability to create the output of the IMM, which is the expected state of the system.

We derive the IMM on \boxplus -manifolds (\boxplus -IMM) by two changes:

1. We use the \boxplus -EKF [19,20] as the single mode filter. This adapts the filtering step to \boxplus -manifolds. The measurement likelihood can be calculated using Gaussians on \boxplus -manifolds.
2. We use the probabilistic mixing of Gaussians of Section 2 in the interaction and combination step.

The implicit definition of the expected values on \boxplus -manifolds does not give a direct rule to compute the value. It can be computed using the iterative algorithm \boxplus -WeightedSum adapted from [14]:

$$\text{Input: } \bar{X}_0, \bar{x}_j, p(X_j) \quad \forall j \in [1, M], \quad (27)$$

$$\bar{X}_{k+1} = \bar{X}_k \boxplus \sum_{j=1}^M p(X_j) (\bar{x}_j \boxminus \bar{X}_k), \quad (28)$$

$$\bar{x} = \lim_{k \rightarrow \infty} \bar{X}_k. \quad (29)$$

In practice, the iteration is stopped when the change of the mean is small. The convergence behavior depends primarily on the choice of the initial guess \bar{X}_0 . The algorithm is a generalization of [24] and identical to the computation of the mean on compact Lie groups [25].

To shorten the notation of covariance mixing, we define the function \boxplus -WeightedCov:

Input: $\bar{x}, P_j, \bar{x}_j, p(X_j) \forall j \in [1, M]$,

$$J_j = \left. \frac{\partial(\bar{x}_j \boxplus \delta \boxminus \bar{x})}{\partial \delta} \right|_{\delta=\vec{0}}, \quad (30)$$

$$P = \sum_{j=1}^M p(X_j) \left([\bar{x}_j \boxminus \bar{x}]^{\otimes} + J_j P_j J_j^T \right). \quad (31)$$

The \boxplus -IMM (See Table 1) can properly handle generic \boxplus -manifolds and their covariances. It does not require any ad hoc implementation to mix the states as it only uses the \boxplus/\boxminus -interface of the manifold.

Table 1. Original IMM [8] vs. \boxplus -IMM. Dashed boxes belong to the standard IMM only and full boxes to the \boxplus -IMM. z_k may be a \boxplus -manifold \mathcal{D} . Modified from [22].

State, input, process and measurement models	$x_k^j \in \mathcal{S}, P_k^j \in \mathbb{R}^{DOF \times DOF}, u_k \in \mathbb{R}^v, z_k \in \mathcal{D}$ $g_j: \mathcal{S} \times \mathbb{R}^v \times \mathbb{R}^n \mapsto \mathcal{S}, x_{k+1}^j = g_j(x_k^j, u_k, \epsilon_k^j)$ $\epsilon_k^j = \mathcal{N}_n(0, Q_k^j), Q_k^j \in \mathbb{R}^{n \times n}$ $h: \mathcal{S} \mapsto \mathcal{D}, z_k = h(x_k) \boxplus_Z \mathcal{N}_d(0, R_k), R_k \in \mathbb{R}^{d \times d}$
Initialization $\forall j \in [1, M]$	$\hat{x}_{0 0}^j = x_0, P_{0 0}^j = P_0, \mu_{0 0}^j = \mu_0^j$
Interaction $\forall i, j \in [1, M]$	$c_k^j = \sum_{i=1}^M p_{ij} \mu_{k k}^i$ $\mu_{k k}^{ij} = \frac{p_{ij} \mu_{k k}^i}{c_k^j}, p_{ij} \text{ are transition probabilities.}$ <div style="border: 1px dashed black; padding: 5px; margin: 5px 0;"> $\hat{x}_{k k}^{0j} = \sum_{i=1}^M \hat{x}_{k k}^i \mu_{k k}^{ij}$ </div> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> $\hat{x}_{k k}^{0j} = \boxplus\text{-WeightedSum}_{i \in [1, M]}(\hat{x}_{k k}^i, \mu_{k k}^{ij})$ </div> <div style="border: 1px dashed black; padding: 5px; margin: 5px 0;"> $P_{k k}^{0j} = \sum_{i=1}^M \mu_{k k}^{ij} \left(P_{k k}^i + [\hat{x}_{k k}^i - \hat{x}_{k k}^{0j}]^{\otimes} \right)$ </div> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> $P_{k k}^{0j} = \boxplus\text{-WeightedCov}_{i \in [1, M]}(\hat{x}_{k k}^{0j}, P_{k k}^i, \hat{x}_{k k}^i, \mu_{k k}^{ij})$ </div>
Filtering: Prediction (\boxplus-EKF) $\forall j \in [1, M]$	$\hat{x}_{k+1 k}^j = g_j(\hat{x}_{k k}^{0j}, u_k, \vec{0})$ $F_k^j = \left. \frac{\partial(g_j(\hat{x}_{k k}^{0j} \boxplus \delta, u_k, \vec{0}) \boxminus \hat{x}_{k+1 k}^j)}{\partial \delta} \right _{\delta=\vec{0}}$ $U_k^j = \left. \frac{\partial(g_j(\hat{x}_{k k}^{0j}, u_k, \epsilon) \boxminus \hat{x}_{k+1 k}^j)}{\partial \epsilon} \right _{\epsilon=\vec{0}}$ $P_{k+1 k}^j = F_k^j P_{k k}^{0j} (F_k^j)^T + U_k^j Q_k^j (U_k^j)^T$

Table 1. Cont.

Filtering: Update (\boxplus-EKF) $\forall j \in [1, M]$	$H_k^j = \left. \frac{\partial \left(h(\hat{x}_{k+1 k}^j \boxplus \delta) \boxminus h(\hat{x}_{k+1 k}^j) \right)}{\partial \delta} \right _{\delta=\vec{0}}$ $S_k^j = H_k^j P_{k+1 k}^j \left(H_k^j \right)^T + R_k$ $W_k^j = P_{k+1 k}^j \left(H_k^j \right)^T \left(S_k^j \right)^{-1}$ $r_k^j = z_k \boxminus h \left(\hat{x}_{k+1 k}^j \right)$ $\hat{x}_{k+1 k+1}^j = \hat{x}_{k+1 k}^j \boxplus W_k^j r_k^j$ $J_k^j = \left. \frac{\partial \left(\hat{x}_{k+1 k}^j \boxplus (r_k^j + \delta) \boxminus \hat{x}_{k+1 k+1}^j \right)}{\partial \delta} \right _{\delta=\vec{0}}$ $P_{k+1 k+1}^j = J_k^j \left(P_{k+1 k}^j - W_k^j S_k^j \left(W_k^j \right)^T \right) \left(J_k^j \right)^T$ $\Lambda_k^j = N \left(r_k^j; 0, S_k^j \right)$ $\mu_{k+1 k+1}^j = \frac{1}{c} \Lambda_k^j c_k^j, \text{ } c \text{ is a normalization factor}$
Combination	$\hat{x}_{k+1 k+1} = \sum_{j=1}^M \hat{x}_{k+1 k+1}^j \mu_{k+1 k+1}^j$ $\hat{x}_{k+1 k+1} = \boxplus\text{-WeightedSum} \left(\hat{x}_{k k}, \hat{x}_{k+1 k+1}^j, \mu_{k+1 k+1}^j \right)_{j \in [1, M]}$ $P_{k+1 k+1} = \sum_{j=1}^M \mu_k^j \left(P_{k+1 k+1}^j + [\hat{x}_{k+1 k+1}^j - \hat{x}_{k+1 k+1}] \otimes \right)$ $P_{k+1 k+1} = \boxplus\text{-WeightedCov} \left(\hat{x}_{k+1 k+1}, P_{k+1 k+1}^j, \hat{x}_{k+1 k+1}^j, \mu_{k+1 k+1}^j \right)_{j \in [1, M]}$

4. An RTS Smoother on \boxplus -Manifolds

Since there is no single model smoother for general \boxplus -manifolds available, we derive an RTS Extended Kalman Smoother on \boxplus -manifolds (\boxplus -EKS) in this section.

For filtering algorithms, it is suitable to express the covariance with respect to the expected value, but care must be taken at smoothing. The reason is that different estimates (the filtered, smoothed, and predicted estimate) participate in the smoothing equations. Thus, every covariance in the equations is defined on a different tangent space. They are incompatible unless they are transformed to the same tangent space. Therefore, we derive the smoother equations for \boxplus -manifolds with special care of the used tangent spaces.

Our derivation of the \boxplus -EKS is based on the discussion of Singer [26]. We start with the theorem of normal correlation for \boxplus -manifolds. The theorem allows us to calculate the conditional probability $p(X|Z)$ of two jointly Gaussian distributed random variables X and Z .

We express the distributions in the two different tangent spaces with references r for X and ζ for Z . This enables us to apply the standard theorem of normal correlation on the tangent spaces around r and ζ :

$$E_r(X|Z) = E_r(X) + \text{Cov}_{r\zeta}(X, Z) \text{Cov}_{\zeta}(Z)^{-1} [(Z \boxminus \zeta) - E_{\zeta}(Z)], \quad (32)$$

$$\text{Cov}_r(X|Z) = \text{Cov}_r(X) - \text{Cov}_{r\zeta}(X, Z) \text{Cov}_{\zeta}(Z)^{-1} \text{Cov}_{r\zeta}(X, Z)^T. \quad (33)$$

A formal requirement for this is that $X \boxplus r$ and $Z \boxplus \zeta$ are jointly Gaussian-distributed. For \boxplus -manifolds, this almost only holds for linear approximation. We still use the notation $=$ instead of \approx to clarify where further approximations are required. All expected values and covariances are expressed in the tangent spaces of r and ζ , and $\text{Cov}_{r\zeta}(X, Z)$ implicitly converts the reference tangent space. As a side remark, choosing $\zeta = E(Z)$ would result in the theorem of normal correlations that is used in current \boxplus -estimators [14,20].

The theorem can be applied to any probability space, conditioned under some other variables Y :

$$E_r(X|Z, Y) = E_r(X|Y) + \text{Cov}_{r\zeta}(X, Z|Y) \text{Cov}_{\zeta}(Z|Y)^{-1} \left[\underline{(Z|Y \boxplus \zeta)} - E_{\zeta}(Z|Y) \right], \quad (34)$$

$$\text{Cov}_r(X|Z, Y) = \text{Cov}_r(X|Y) - \text{Cov}_{r\zeta}(X, Z|Y) \text{Cov}_{\zeta}(Z|Y)^{-1} \text{Cov}_{r\zeta}(X, Z|Y)^T. \quad (35)$$

For readability, we underline conditional distributions if neither their expected value nor their covariance is meant. Following [26], the variables X , Z , and Y are interpreted in an unusual fashion: X is the state x_k at time k , Z is the state x_{k+1} at time $k+1$, and Y is the vector of all measurements $z_{1:k}$ up to time k .

$$E_r(x_k|x_{k+1}, z_{1:k}) = E_r(x_k|z_{1:k}) + C_k \left[\underline{(x_{k+1}|z_{1:k} \boxplus \zeta)} - E_{\zeta}(x_{k+1}|z_{1:k}) \right], \quad (36)$$

$$\text{Cov}_r(x_k|x_{k+1}, z_{1:k}) = \text{Cov}_r(x_k|z_{1:k}) - C_k \text{Cov}_{\zeta}(x_{k+1}|z_{1:k}) C_k^T, \quad (37)$$

$$C_k := \text{Cov}_{r\zeta}(x_k, x_{k+1}|z_{1:k}) \text{Cov}_{\zeta}(x_{k+1}|z_{1:k})^{-1}, \quad (38)$$

where C_k is known as the smoother gain. These formulas essentially describe the fusion of the past $z_{1:k}$ with a hypothetically fixed next state x_{k+1} . The next state is actually not fixed but distributed, given all measurements $z_{1:N}$. Thus, we have to use the formulas in [26] to calculate the expectation and variance:

$$E_r(x_k|z_{1:N}) = E(E_r(x_k|x_{k+1}, z_{1:k})|z_{1:N}), \quad (39)$$

$$\text{Cov}_r(x_k|z_{1:N}) = E(\text{Cov}_r(x_k|x_{k+1}, z_{1:k})|z_{1:N}) + \text{Cov}(E_r(x_k|x_{k+1}, z_{1:k})|z_{1:N}). \quad (40)$$

Note that only the distribution $x_{k+1}|z_{1:k}$ has to be conditioned on $z_{1:N}$ since all other values are known constants. The expected value and covariance of (39) can be calculated by standard rules:

$$\begin{aligned} E_r(x_k|z_{1:N}) &= E \left(E_r(x_k|z_{1:k}) + C_k \left[\underline{(x_{k+1}|z_{1:N} \boxplus \zeta)} - E_{\zeta}(x_{k+1}|z_{1:k}) \right] \right) \\ &= E_r(x_k|z_{1:k}) + C_k \left[E_{\zeta}(x_{k+1}|z_{1:N}) - E_{\zeta}(x_{k+1}|z_{1:k}) \right], \end{aligned} \quad (41)$$

$$\begin{aligned} \text{Cov}(E_r(x_k|x_{k+1}, z_{1:k})|z_{1:N}) &= \text{Cov} \left(E_r(x_k|z_{1:k}) + C_k \left[\underline{(x_{k+1}|z_{1:N} \boxplus \zeta)} - E_{\zeta}(x_{k+1}|z_{1:k}) \right] \right) \\ &= \text{Cov} \left(C_k \left(\underline{(x_{k+1}|z_{1:N} \boxplus \zeta)} \right) \right) \\ &= C_k \text{Cov} \left(\underline{(x_{k+1}|z_{1:N} \boxplus \zeta)} \right) C_k^T \\ &= C_k \text{Cov}_{\zeta}(x_{k+1}|z_{1:N}) C_k^T. \end{aligned} \quad (42)$$

Substituting all into Equation (40) yields

$$\text{Cov}_r(x_k|z_{1:N}) = \text{Cov}_r(x_k|z_{1:k}) + C_k \left[\text{Cov}_{\zeta}(x_{k+1}|z_{1:N}) - \text{Cov}_{\zeta}(x_{k+1}|z_{1:k}) \right] C_k^T. \quad (43)$$

From here, it is required to choose suitable references r and ζ to be able to compute the smoothing step. Choosing the references is crucial for the mathematical complexity of the computation. Covariances in the \boxplus -IMM all live in the tangent space of the expected value, i.e., the estimate. Thus, they are all in different tangent spaces and have to be transferred to the tangent spaces required by the smoothing step. The choice of the references decides how many transforms have to be calculated. We choose $r = E(x_k|z_{1:k})$ and $\zeta = E(x_{k+1}|z_{1:k})$.

This avoids transforming the covariances $\text{Cov}_r(x_k|z_{1:k})$ and $\text{Cov}_\zeta(x_{k+1}|z_{1:k})$ as they are already available in these tangent spaces. It also reduces the expected value to

$$\begin{aligned} E_r(x_k|z_{1:N}) &= E(x_k|z_{1:k}) \boxplus E(x_k|z_{1:k}) + C_k \left[E(x_{k+1}|z_{1:N}) \boxminus E(x_{k+1}|z_{1:k}) \right] \\ &= C_k E(x_{k+1}|z_{1:N}) \boxplus E(x_{k+1}|z_{1:k}). \end{aligned} \quad (44)$$

By linear approximation, we retrieve

$$E_r(x_k|z_{1:N}) \approx C_k [E(x_{k+1}|z_{1:N}) \boxminus E(x_{k+1}|z_{1:k})]. \quad (45)$$

The value of $\text{Cov}_\zeta(x_{k+1}|z_{1:N})$ can be retrieved by a displaced transform. Finally, we give a formula for $\text{Cov}_{r\zeta}(x_k, x_{k+1}|z_{1:k})$ by approximation of

$$\text{Cov}_{r\zeta}(x_k, x_{k+1}|z_{1:k}) = \text{Cov}(x_k|z_{1:k} \boxminus E(x_k|z_{1:k}), g(x_k|z_{1:k}) \boxminus E(x_{k+1}|z_{1:k})), \quad (46)$$

where g is the dynamic model function. Approximating the second argument yields

$$\begin{aligned} \text{Cov}_{r\zeta}(x_k, x_{k+1}|z_{1:k}) &\approx \text{Cov}(x_k|z_{1:k} \boxminus E(x_k|z_{1:k}), g(E(x_k|z_{1:k})) \boxminus E(x_{k+1}|z_{1:k}) + F_k [x_k|z_{1:k} \boxminus E(x_k|z_{1:k})]) \\ &= \text{Cov}_r(x_k|z_{1:k}) F_k^T \end{aligned} \quad (47)$$

$$F_k = \left. \frac{\partial (g(E(x_k|z_{1:k})) \boxplus \delta) \boxminus E(x_{k+1}|z_{1:k}))}{\partial \delta} \right|_{\delta=\vec{0}}, \quad (48)$$

where F_k is commonly known as the state transition matrix.

With the chosen references, the complete smoothing step is (using more common symbols)

$$\hat{x}_{k|N} = \hat{x}_{k|k} \boxplus C_k [\hat{x}_{k+1|N} \boxminus \hat{x}_{k+1|k}], \quad (49)$$

$$P_{k|N} = J_k \left(P_{k|k} + C_k [B_k P_{k+1|N} B_k^T - P_{k+1|k}] C_k^T \right) J_k^T, \quad (50)$$

$$C_k = P_{k|k} F_k^T P_{k+1|k}^{-1}, \quad (51)$$

$$F_k = \left. \frac{\partial (g(\hat{x}_{k|k}) \boxplus \delta) \boxminus \hat{x}_{k+1|k}}{\partial \delta} \right|_{\delta=\vec{0}}, \quad (52)$$

$$B_k = \left. \frac{\partial ((\hat{x}_{k+1|N} \boxplus \delta) \boxminus \hat{x}_{k+1|k})}{\partial \delta} \right|_{\delta=\vec{0}}, \quad (53)$$

$$J_k = \left. \frac{\partial ((\hat{x}_{k|k} \boxplus (C_k [\hat{x}_{k+1|N} \boxminus \hat{x}_{k+1|k}] + \delta)) \boxminus \hat{x}_{k|N})}{\partial \delta} \right|_{\delta=\vec{0}}. \quad (54)$$

These formulas are structurally similar to the standard RTS formulas [23]:

$$\hat{x}_{k|N} = \hat{x}_{k|k} + C_k [\hat{x}_{k+1|N} - \hat{x}_{k+1|k}], \quad (55)$$

$$P_{k|N} = P_{k|k} + C_k [P_{k+1|N} - P_{k+1|k}] C_k^T, \quad (56)$$

$$C_k = P_{k|k} F_k^T P_{k+1|k}^{-1}, \quad (57)$$

$$F_k = \left. \frac{\partial g(x)}{\partial x} \right|_{x=\hat{x}_{k|k}}. \quad (58)$$

In the vector case, the \boxplus/\boxminus operators reduce to $+/-$ and the computation of the expected value agrees completely. In contrast, the covariance computation shows important differences — the transforms with the matrices J_k and B_k . B_k transforms the covariance

of the smoothed future state to the tangent space of the filtered predicted state. Thus, the matrices $B_k P_{k+1|N} B_k^T$ and $P_{k+1|k}$ are on the same tangent space, which allows them to be subtracted. J_k conducts a centered transform to reestablish the standard Gaussian form. Nevertheless, the covariance reduces to the standard RTS formulas in the vector case as well, since B_k and J_k evaluate to identity. Therefore, the derived smoothing formulas can be seen as a generalization of the standard RTS formulas.

Note that the choice of the references r and ζ is crucial for the structure of the final formulas. Other references may alter the performance since other linearization points are used. Regardless of the choice of the references, the formulas reduce to the standard RTS formulas in the vector case, because all transformation matrices reduce to identity and the reference ζ cancels out at the computation of the expected value. Thus, countless possibilities exist to generalize the RTS formulas on \boxplus -manifolds.

The presented \boxplus -EKS differs from existing RTS smoothers on quaternions [17]. Since smoothing on a quaternion state is a special case of our general \boxplus -EKS, it should coincide. The smoothed covariance is calculated differently from the existing smoothers as they use the vector Equation (56). The existing RTS smoothers ignore that the covariances are defined on different tangent spaces. In practice, the formulas are almost identical for quaternions if the estimates are close to each other. In that case, the \boxplus/\boxminus -operators are almost linear; wherefore, B_k and J_k evaluate close to identity.

5. IMM RTS Smoother

Since the IMM performs an arbitrary (but well-chosen) approximation of the hybrid estimation problem, there is no single answer on how to perform the smoothing. Several smoothing schemes exist in the literature [11–13]. We choose the scheme of [11] as it requires only one smoothing step per mode and is straightforward to adapt to \boxplus -manifolds with our results from Sections 2 and 4. [12] pointed out that the chosen scheme uses strong approximations. However, the approach of [12] requires operations that are currently not possible on general \boxplus -manifolds.

The IMM RTS smoother in [11] can be adapted to \boxplus -manifolds in a similar manner as the IMM Filter. The mixture of Gaussian approximations are exchanged with the \boxplus -WeightedSum and \boxplus -WeightedCov algorithms. The mode-matched smoothing is exchanged with the \boxplus -EKS from Section 4. This results in the \boxplus -RTSIMM smoother (\boxplus -RTSIMMS), as shown in Table 2.

The \boxplus -RTSIMMS generically operates on all differentiable \boxplus -manifolds without destroying the manifold properties of the state. To our knowledge, this is the first IMM smoother that can operate properly on quaternions and other manifolds.

Table 2. The \boxplus -RTSIMMS smoothing scheme based on [11].

Backward transition probability	$b_{ij} = \frac{1}{e_i} p_{ji} \mu_{k k}^j$ $e_i = \sum_{l=1}^M p_{li} \mu_{k k}^l$
Backward mixing probability	$\mu_{k+1 N}^{ij} = \frac{1}{d_j} b_{ij} \mu_{k+1 N}^i$ $d_j = \sum_{l=1}^M b_{li} \mu_{k+1 N}^l$
Backward mixing	$\hat{x}_{k+1 N}^{0j} = \boxplus\text{-WeightedSum}_{i \in [1, M]}(\hat{x}_{k+1 N}^j, \hat{x}_{k+1 N}^i, \mu_{k+1 N}^{ij})$ $P_{k+1 N}^{0j} = \boxplus\text{-WeightedCov}_{i \in [1, M]}(\hat{x}_{k+1 N}^{0j}, P_{k+1 N}^i, \hat{x}_{k+1 N}^i, \mu_{k+1 N}^{ij})$

Table 2. Cont.

Mode-matched smoothing	$\hat{x}_{k N}^j = \hat{x}_{k k}^j \boxplus C_k^j [\hat{x}_{k+1 N}^{0j} \boxminus \hat{x}_{k+1 k}^j]$ $P_{k N}^j = J_k^j (P_{k k}^j + C_k^j [B_k^j P_{k+1 N}^{0j} B_k^{jT} - P_{k+1 k}^j] C_k^{jT}) J_k^{jT}$ $C_k^j = P_{k k}^j F_k^{jT} (P_{k+1 k}^j)^{-1}$ $F_k^j = \left. \frac{\partial (g_j(\hat{x}_{k k}^j \boxplus \delta) \boxminus \hat{x}_{k+1 k}^j)}{\partial \delta} \right _{\delta=\vec{0}}$ $B_k^j = \left. \frac{\partial ((\hat{x}_{k+1 N}^j \boxplus \delta) \boxminus \hat{x}_{k+1 k}^j)}{\partial \delta} \right _{\delta=\vec{0}}$ $J_k^j = \left. \frac{\partial ((\hat{x}_{k k}^j \boxplus (C_k^j [\hat{x}_{k+1 N}^{0j} \boxminus \hat{x}_{k+1 k}^j] + \delta)) \boxminus \hat{x}_{k N}^j)}{\partial \delta} \right _{\delta=\vec{0}}$
Smoothed mode probability	$\Lambda_{k N}^j = \sum_{i=1}^M p_{ji} \mathcal{N}(\hat{x}_{k+1 N}^i; \hat{x}_{k+1 k}^j, P_{k+1 k}^j)$ $\mu_{k N}^j = \frac{1}{f} \Lambda_{k N}^j \mu_{k k}^j, \quad f \text{ is a normalizing factor}$
Combine smoothed estimate	$\hat{x}_{k N} = \boxplus\text{-WeightedSum}_{j \in [1, M]}(\hat{x}_{k k}, \hat{x}_{k N}^j, \mu_{k N}^j)$ $P_{k N} = \boxplus\text{-WeightedCov}_{j \in [1, M]}(\hat{x}_{k N}, P_{k N}^j, \hat{x}_{k N}^j, \mu_{k N}^j)$

6. Example Application and Performance Discussion

We test the new algorithms in a simulated environment to provide first insights into their performance. We choose the following setup inspired by classic radar tracking: A drone flies across known terrain. It has a 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 simulated drone flies a trajectory of two straight lines and two 180° curves over four visible landmarks.

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 :

$$x = (q_b^w \quad \vec{p}_w \quad \vec{v}_w \quad \vec{\omega}_w)^T, \quad (59)$$

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 follows:

$$g_s(x, \epsilon_s) = \begin{pmatrix} q_b^w \\ \vec{p}_w + (\vec{v}_w + \epsilon_s) \Delta t \\ \vec{v}_w + \epsilon_s \\ \vec{\omega}_w \end{pmatrix}, \quad (60)$$

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 :

$$g_c(x, \epsilon_c) = \begin{pmatrix} q_b^w \exp(\frac{\Delta t}{2} \vec{\omega}_w)^{-1} \\ \vec{p}_w + (\Delta t I_{3 \times 3} + B) \vec{v}_w \\ (I_{3 \times 3} + A) \vec{v}_w \\ \vec{\omega}_w + \epsilon_c \end{pmatrix}, \quad (61)$$

where $\exp(\cdot)$ forms a quaternion from the given Euler-angle-axis [14] 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 instead of pixel coordinates. The measurement model is

$$h(x) = q_b^w \cdot \vec{p}_w \cdot (q_b^w)^{-1} \quad (62)$$

The covariance matrices can be found in Appendix A.

6.1. Results

The performance of seven algorithms is evaluated in the simulation:

1. \boxplus -EKF: The EKF on \boxplus -manifolds, as presented in [20], on the constant turn model (61).
2. \boxplus -EKS: The RTS EKS on \boxplus -manifolds, as derived in Section 4, on the constant turn model (61).
3. (M)-EKS: The same as the \boxplus -EKS, but it uses the simplified covariance smoothing (56). This algorithm uses the \boxplus -EKF for filtering but its smoothing step is identical to the MEKS or the left invariant EKS on quaternions [17].
4. \boxplus -IMM: The IMM on \boxplus -manifolds, as derived in Section 3.
5. N-IMM: An IMM that handles the manifold state such as a vector state for mixing purposes. We call this approach naive-mixing: The quaternions are summed up in parameter space and normalized afterwards (as in [5]). It uses the standard IMM equation to sum up the covariances. The mode filters of the N-IMM are \boxplus -EKFs, since we want to specifically compare the \boxplus -mixing from Section 2 against naive-mixing.
6. \boxplus -RTSIMMS: The IMM RTS smoother on \boxplus -manifolds, as derived in Section 5.
7. NM-RTSIMMS: An IMM smoother that handles mixing such as the N-IMM and uses the (M)-EKS for mode-matched smoothing. This IMM basically combines the approaches in [5] and [17].

The root mean squared error (RMSE) of the position and orientation are used as performance metrics. The \boxminus -operator is used to calculate the quaternion estimate error (difference as Euler-Angle-Axis). In addition, the consistency measure of [27] is computed; it states that the estimate is consistent if it resembles the true probability distribution. Hence, the expectation bias $E(\hat{x} \boxminus x)$ has to be 0 and the expectation of the squared Mahalanobis distances $E(\|\hat{x} \boxminus x\|_P^2)$ has to equal the *DOF* of the state. Both values are computed on the position and orientation, wherefore *DOF* = 6. The averaged metrics of 100 Monte Carlo runs are shown in Figure 2.

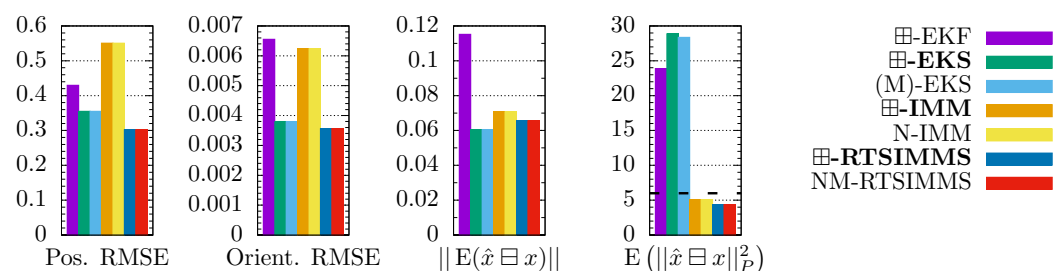


Figure 2. RMSE and consistency comparison for aircraft tracking in 100 Monte Carlo runs. The optimal value is 0 for $\|E(\hat{x} \boxminus x)\|$ and 6 for $E(\|\hat{x} \boxminus x\|_P^2)$ (dashed line).

Both IMM smoother variants caused numerical issues at the computation of the mode-matched smoothed covariances. Apparently, the subtraction of covariances results in indefinite matrices. Since the issue arises in both IMM smoothers, it does not appear to be specific for the changed covariance formula (50). To ensure positive definite covariance matrices, the eigenvalues of the smoothed matrices are set to a minimal threshold $\epsilon > 0$. All metrics of the \boxplus -RTSIMMS and NM-RTSIMMS are affected by this regularization.

As expected, the RMSEs and biases of the smoother variants are lower than their filter counterparts. Remarkably, the single model algorithms perform better than the multiple model algorithms in some of the metrics. However, the IMM variants deliver overall more consistent estimates. In the remainder of our discussion, we focus on the comparison between our three, novel algorithms and their respective state-of-the-art solutions. The \boxplus -EKF metrics are only used to explain different behavior between the single- and multimodel smoothers.

The \boxplus -EKS and (M)-EKS perform equally on the three metrics that concern only the state, since they only differ in the covariance computation. The (M)-EKS is closer to the optimal value $E(\|\hat{x} \boxminus x\|_p^2) = 6$ than the \boxplus -EKS, which implies a slightly better consistency of the covariances. In contrast, the different covariance computation has an effect on the state for the IMM smoothers, as it influences the model probabilities.

The \boxplus -IMM and N-IMM perform identically at the leading digits. Hence, the \boxplus -mixing does not change the computation in the presented example. The IMM smoothers perform almost identically as well.

This results are unsatisfying since the theoretically justified \boxplus -mixing and smoothing should yield better accuracy and consistency. Therefore, we analyze the effects of the \boxplus -mixing and the changed covariance smoothing equation.

6.2. Discussion: \boxplus -Mixing of Gaussians

The error of the naive-mixing is negligible in the presented example. To show this, we try to quantify the error induced by naive-mixing. We calculate the weighted mean of two quaternion Gaussian distributions q_1, q_2 for different angular differences and different probabilities (see Figure 3a).

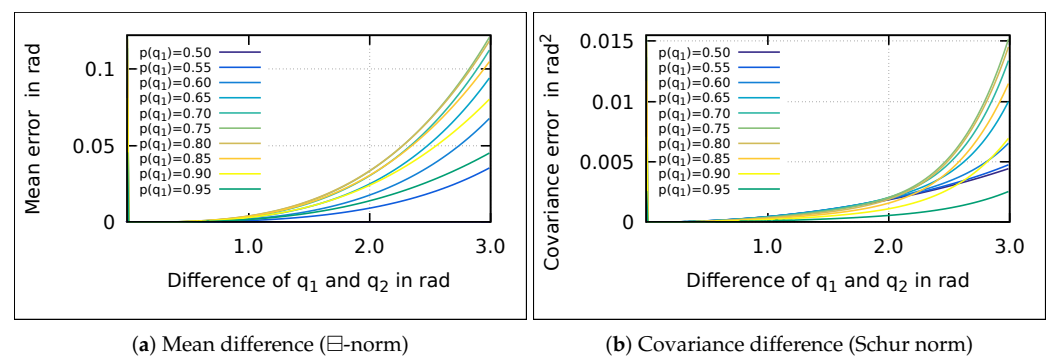


Figure 3. The mean and covariance difference between \boxplus - and naive-mixing over the angular differences of q_1 and q_2 . Reprinted with permission from ref. [22]. Copyright 2020 IEEE.

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 Figure 3b). 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.

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

the dynamic models are compensated by the update. Likewise, in the IMM smoother, the mixing is performed on the updated or smoothed estimates.

The \boxplus -mixing may perform better for higher differences. To analyze this, \boxplus - and naive-mixing are compared to an optimal solution (see Figure 4). Since no closed form solution to mix the Gaussians exists, the optimal solution is obtained numerically. Quaternions are sampled uniformly from the Gaussians to approximate the complete distribution.

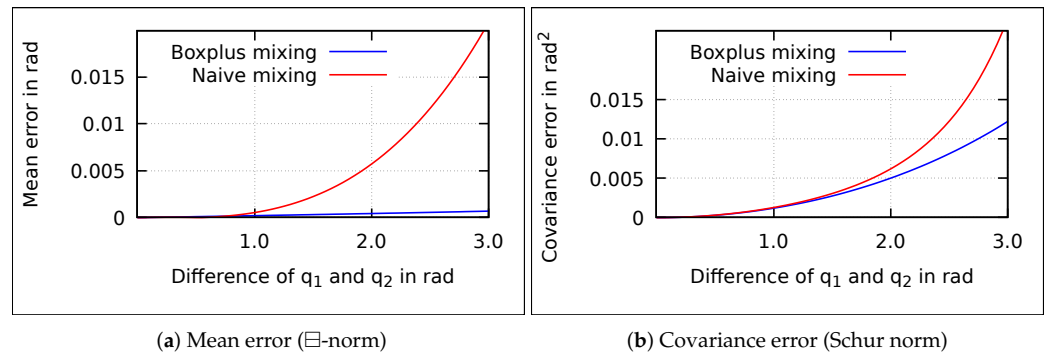


Figure 4. Mean and covariance error of \boxplus - and naive-mixing compared to optimal mixing over the angular differences of q_1 and q_2 . Reprinted with permission from ref. [22]. Copyright 2020 IEEE.

The mean and covariance error of the \boxplus -mixing increase with the distance between the Gaussian means. The approaches are almost equal for small differences between the quaternions. At higher differences, \boxplus - outperforms naive-mixing. Thus, \boxplus -mixing is technically an improvement but its advantage is negligible due to the small differences in IMM applications.

6.3. Discussion: \boxplus -RTS Smoothing

The \boxplus -EKS is slightly less consistent than the (M)-EKS. The difference between the \boxplus -EKS and (M)-EKS is the reference transformation of the covariances. Therefore, we analyze the performance of the linear approximation of the reference transform.

We compare the conducted linear approximation of the reference transform with a numerically computed optimum, where we distinguish between displaced and centered transforms (see Figure 5). For further comparison, we add the error of no transformation, which shows the behavior of the (M)-EKS. The covariance of the Gaussians has been set to $\Sigma = 0.1I_{3 \times 3}$.

The error graphs of the centered and displaced transform clearly differ, which justifies the distinction of the two transform types. The error of the displaced transform (see Figure 5b) is, for high differences of the references, several orders of magnitude higher than the base covariance. The reason is that wrap-around issues arise, since we approach the classic singularity of quaternions close to angles of π . Note that closeness depends on the chosen covariance. With a higher covariance, the wrap-around issues occur earlier. Since the linear transformation evaluates a single point, the wrap-around is not incorporated. However, such extreme differences of the references do not usually occur during estimation.

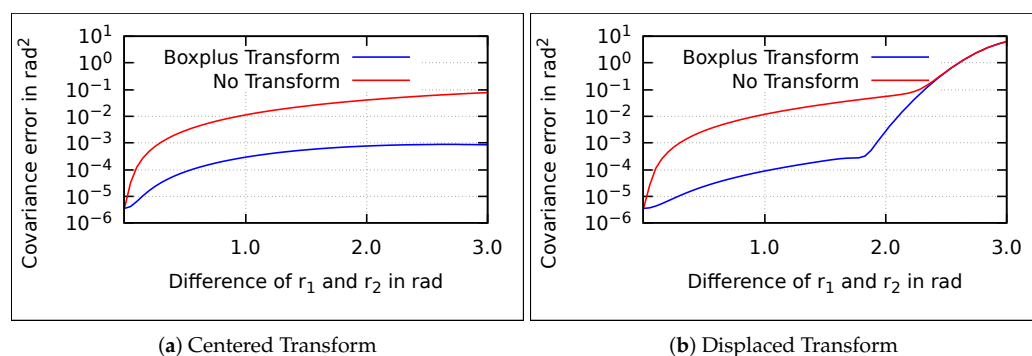


Figure 5. The covariance error (Schur norm) in logarithmic scale of the \boxplus -transform and no transform compared to numerical optimal transformation over the angular differences of r_1 and r_2 .

The comparison shows that the linear approximation has a smaller error than no transformation. Thus, regarding a single smoothing step, the \boxplus -EKS is more precise than the (M)-EKS. Apparently, the improvement of a single step does not improve the whole interval smoothing. To investigate the accumulated effect of the transforms, we plot the ratio of the covariance determinants before and after the reference transform (see Figure 6). In the quaternion case, the displaced transform monotonically increases the covariance and the centered transform reduces it.

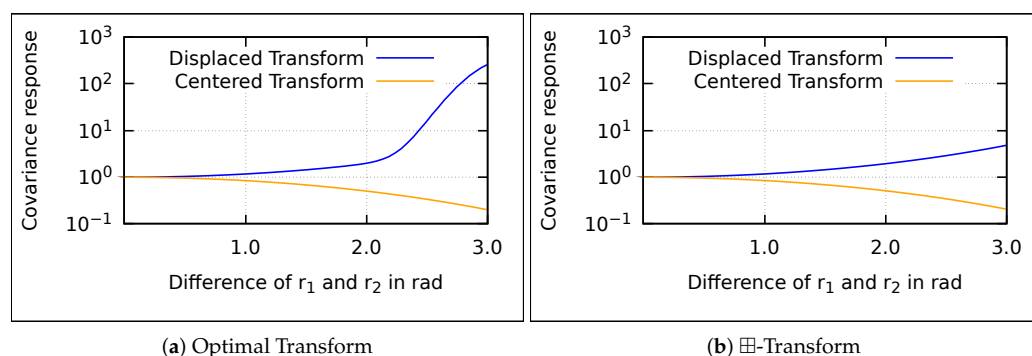


Figure 6. The covariance response in logarithmic scale over the angular differences of r_1 and r_2 . The covariance response is the ratio of the transformed and base covariance determinants.

In the \boxplus -EKS formula, the centered transform is dominant as it encloses the whole Equation (50). Thus, the \boxplus -EKS reduces the covariance marginally in comparison with the (M)-EKS. Since the covariance is already too small in the simulation, the \boxplus -EKS is slightly less consistent. Otherwise, if the covariances are too high, this effect would yield more consistent results. Thus, this is neither a clear performance degradation nor an improvement.

Noticeably, the IMM smoother variants perform almost identically. Since the IMM has an overall smaller bias than the EKF, the differences between the filtered and smoothed estimates are smaller. Thus, the reduction of the covariance matrices becomes negligible. This result coincides with the analysis of Section 6.2, since the \boxplus -mixing of covariances is founded on displaced transforms.

The hypothesis that the bias has a crucial influence can be further substantiated by adapting the noise of the constant turn model to reduce the bias. This is possible since the low inconsistent noise biases the estimators towards the dynamic model instead of the measurements. Thus, the bias can be drastically reduced and the difference in consistency vanishes using an additive noise on the position of $\Sigma = 10\Delta_t I_{3 \times 3}$ (see Figure 7).

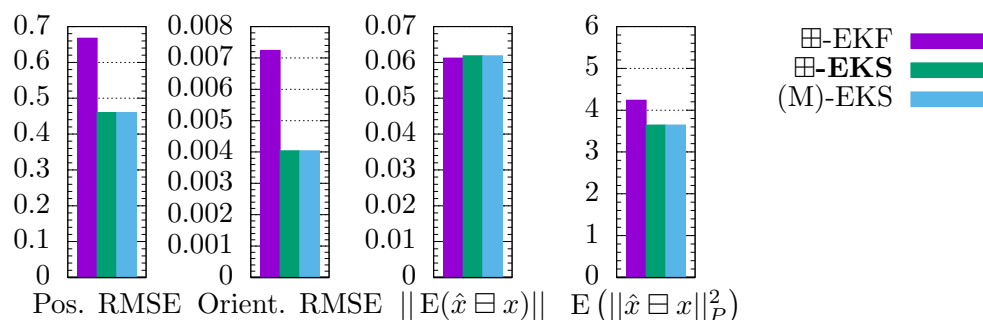


Figure 7. RMSE and consistency comparison of single-model estimators with additional noise on the position.

6.4. Discussion: Wrap-Up

Overall, using \boxplus -mixing with a first-order approximation does not give a performance boost for the IMM on quaternions. Instead, its strength is to enable a generic IMM on differentiable \boxplus -manifolds. The method encapsulates the manifold properties of the state so 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.

It has been shown that, in the quaternion case, the reference transforms are negligible if the references are close to each other. The transforms can be dropped to reduce computational and algorithmic complexity. Thus, we recommend to use the approach of [17] for quaternion smoothing. For other manifolds, the reference transform may be relevant to achieve consistent results.

Open source C++ implementations of the generic \boxplus -EKS (<https://github.com/TomLKoller/Manifold-RTS-Smoother>, accessed on 16 June 2021) and the \boxplus -IMM and \boxplus -RTSIMMS (<https://github.com/TomLKoller/Boxplus-IMM>, accessed on 16 June 2021) are provided. The implementations use automatic differentiation [28] to calculate all required Jacobians for state mixing and for the internal \boxplus -EKF [29]. They use a generic state representation to adapt to other system models. Therefore, they can be used without taking care of the heavy math in this paper. The \boxplus -IMM repository also contains the presented simulation example.

7. Conclusions

Three, novel estimation algorithms based on the \boxplus -method have been presented in this work. All algorithms properly handle the manifold structure of \boxplus -manifolds such as quaternions or rotation matrices. Hence, they do not need ad hoc normalization procedures to preserve manifold structures.

The \boxplus -IMM, a first-order correct IMM on \boxplus -manifolds, has been derived. We described methods to calculate the weighted mean and covariance of Gaussian mixtures on \boxplus -manifolds and provided necessary proofs. With these, the \boxplus -method is applied to the IMM.

By following the RTS derivation of [26], the new \boxplus -EKS has been derived. The \boxplus -EKS uses a different smoothing update of the covariance than published smoothers for quaternions, but the expected value is calculated identically.

By combining the \boxplus -EKS and the \boxplus -IMM, a novel IMM smoother on \boxplus -manifolds, \boxplus -RTSIMMS, has been derived. To our knowledge, the \boxplus -RTSIMMS is the first IMM that enables smoothing on rotation quaternions and other \boxplus -manifolds in hybrid estimation.

All algorithms are evaluated on a simulated aircraft tracking example. The evaluation shows that the estimation accuracies are not improved compared to state-of-the-art smoothing methods on quaternions and ad hoc normalization procedures. For other manifolds, a practical advantage may be achieved. The presented algorithms still have high theoretical value as they can be derived from the basic definitions of the expected value and the covariance on \boxplus -manifolds. Thus, they are justified, generic algorithms.

This paper extended the family of \boxplus -algorithms to hybrid estimation and smoothing. The practical strength of the presented algorithms is their generality. The \boxplus -method encapsulates the manifold properties. No further ad hoc implementations are required to

perform mixing and smoothing, regardless of the state. Therefore, the algorithms enable the implementation of a generic library that can handle \boxplus -manifold states in hybrid estimation and smoothing. A first prototype is published alongside this paper.

The presented methods are only first-order correct, which causes linearization errors. Thus, one may develop higher-order methods or use the unscented transform as approximation, which may improve the consistency of the \boxplus -EKS. Presumably, the unscented transform would not significantly reduce the error, as the error compared to the numerical solutions was quite low for small distances anyway. Instead, it should be investigated whether the indefiniteness of the covariance arises from numerical issues or systematic problems in the formulas, e.g., the approximation pointed out by [12].

Author Contributions: Conceptualization, T.L.K.; data curation, T.L.K.; formal analysis, T.L.K.; funding acquisition, U.F.; investigation, T.L.K. and U.F.; methodology, T.L.K. and U.F.; project administration, U.F.; resources, U.F.; software, T.L.K.; supervision, U.F.; validation, T.L.K. and U.F.; visualization, T.L.K.; writing—original draft, T.L.K.; writing—review & editing, T.L.K. and U.F. Both authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the German Research Foundation (DFG) grant number ZaVI FR 2620/3-1.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The Implementations and simulations are available at: <https://github.com/TomLKoller/Manifold-RTS-Smoother> (accessed on 16 June 2021), and <https://github.com/TomLKoller/Boxplus-IMM> (accessed on 16 June 2021).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

IMM	Interacting Multiple Model (Filter)
EKF	Extended Kalman Filter
EKS	Extended Kalman Smoother
RTS	Rauch–Tung–Striebel
RTSIMMS	Rauch–Tung–Striebel Interacting Multiple Model Smoother
MEKF	Multiplicative EKF
MEKS	Multiplicative EKS

Appendix A. Covariances

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

- Blom, H.A.P. An efficient filter for abruptly changing systems. In Proceedings of the 23rd IEEE Conference on Decision and Control, Las Vegas, NV, USA, 12–14 December 1984; pp. 656–658.
- Zhan, R.; Wan, J. Passive maneuvering target tracking using 3D constant-turn model. In Proceedings of the 2006 IEEE Conference on Radar, Verona, NY, USA, 24–27 April 2006; p. 8.
- Glass, J.D.; Blair, W.D.; Bar-Shalom, Y. IMM estimators with unbiased mixing for tracking targets performing coordinated turns. In Proceedings of the 2013 IEEE Aerospace Conference, Big Sky, MT, USA, 2–9 March 2013; pp. 1–10.

4. Zhu, W.; Wang, W.; Yuan, G. An Improved Interacting Multiple Model Filtering Algorithm Based on the Cubature Kalman Filter for Maneuvering Target Tracking. *Sensors* **2016**, *16*, 805. [[CrossRef](#)] [[PubMed](#)]
5. Wann, C.-D.; Gao, J.-H. Orientation Estimation for Sensor Motion Tracking Using Interacting Multiple Model Filter. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* **2010**, *93*, 1565–1568. [[CrossRef](#)]
6. Liu, J.; Cai, B.; Wang, J.; Shangguan, W. GNSS/INS-based vehicle lane-change estimation using IMM and lane-level road map. In Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013), The Hague, The Netherlands, 6–9 October 2013; pp. 148–153.
7. Sadeghzadeh-Nokhodberiz, N.; Poshtan, J. Distributed interacting multiple filters for fault diagnosis of navigation sensors in a robotic system. *IEEE Trans. Syst. Man, Cybern. Syst.* **2016**, *47*, 1383–1393. [[CrossRef](#)]
8. Mazor, E.; Averbuch, A.; Bar-Shalom, Y.; Dayan, J. Interacting multiple model methods in target tracking: a survey. *IEEE Trans. Aerosp. Electron. Syst.* **1998**, *34*, 103–123. [[CrossRef](#)]
9. Boers, Y.; Driessen, J.N. Interacting multiple model particle filter. *IEE Proc. Radar Sonar Navig.* **2003**, *150*, 344–349. [[CrossRef](#)]
10. Ali, W.; Li, Y.; Raja, M.A.Z.; Ahmed, N. Generalized pseudo Bayesian algorithms for tracking of multiple model underwater maneuvering target. *Appl. Acoust.* **2020**, *166*, 107345. [[CrossRef](#)]
11. Nadarajah, N.; Tharmarasa, R.; McDonald, M.; Kirubarajan, T. IMM forward filtering and backward smoothing for maneuvering target tracking. *IEEE Trans. Aerosp. Electron. Syst.* **2012**, *48*, 2673–2678. [[CrossRef](#)]
12. Lopez, R.; Danes, P. Low-complexity IMM smoothing for jump Markov nonlinear systems. *IEEE Trans. Aerosp. Electron. Syst.* **2017**, *53*, 1261–1272. [[CrossRef](#)]
13. Li, K.; Ma, X.; Liu, Y.; Yuan, D. An Interacting Multiple Model Smoothing Algorithm for Nonlinear Maneuvering Target Tracking. In Proceedings of the 2019 IEEE International Conference on Signal, Information and Data Processing (ICSIDP), Chongqing, China, 11–13 December 2019; pp. 1–7.
14. Hertzberg, C.; Wagner, R.; Frese, U.; Schröder, L. Integrating generic sensor fusion algorithms with sound state representations through encapsulation of manifolds. *Inf. Fusion* **2013**, *14*, 57–77. [[CrossRef](#)]
15. Markley, F.L. Attitude error representations for Kalman filtering. *J. Guid. Control. Dyn.* **2003**, *26*, 311–317. [[CrossRef](#)]
16. Madyastha, V.; Ravindra, V.; Mallikarjunan, S.; Goyal, A. Extended Kalman Filter vs. Error State Kalman Filter for Aircraft Attitude Estimation. In Proceedings of the AIAA Guidance, Navigation, and Control Conference, American Institute of Aeronautics and Astronautics, Oregon, Portland, 8–11 August 2011.
17. van Der Laan, N.; Cohen, M.; Arsenault, J.; Forbes, J.R. The Invariant Rauch-Tung-Striebel Smoother. *IEEE Robot. Autom. Lett.* **2020**, *5*, 5067–5074. [[CrossRef](#)]
18. Himberg, H.; Motai, Y.; Bradley, A. A multiple model approach to track head orientation with delta quaternions. *IEEE Trans. Cybern.* **2012**, *43*, 90–101. [[CrossRef](#)] [[PubMed](#)]
19. Clemens, J.; Schill, K. Extended Kalman filter with manifold state representation for navigating a maneuverable melting probe. In Proceedings of the 2016 19th International Conference on Information Fusion (FUSION), Heidelberg, Germany, 5–8 July 2016; pp. 1789–1796.
20. Nakath, D.; Clemens, J.; Schill, K. Multi-Sensor Fusion and Active Perception for Autonomous Deep Space Navigation. In Proceedings of the 2018 21st International Conference on Information Fusion (FUSION), Cambridge, UK, 10–13 July 2018; pp. 2596–2605.
21. Koller, T.L.; Frese, U. State Observability through Prior Knowledge: Analysis of the Height Map Prior for Track Cycling. *Sensors* **2020**, *20*, 2438. [[CrossRef](#)] [[PubMed](#)]
22. Koller, T.L.; Frese, U. The Interacting Multiple Model Filter on Boxplus-Manifolds. In Proceedings of the 2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Karlsruhe, Germany, 14–16 September 2020.
23. Rauch, H.E.; Tung, F.; Striebel, C.T. Maximum likelihood estimates of linear dynamic systems. *AIAA J.* **1965**, *3*, 1445–1450. [[CrossRef](#)]
24. Hartley, R.; Trunpf, J.; Dai, Y.; Li, H. Rotation averaging. *Int. J. Comput. Vis.* **2013**, *103*, 267–305. [[CrossRef](#)]
25. Manton, J.H. A globally convergent numerical algorithm for computing the centre of mass on compact Lie groups. In Proceedings of the ICARCV 2004 8th Control, Automation, Robotics and Vision Conference, Kunming, China, 6–9 December 2004; Volume 3, pp. 2211–2216.
26. Singer, H. *A Concise Proof of Gaussian Smoothing*; FernUniversität in Hagen: Hagen, Germany, 2018.
27. Bar-Shalom, Y.; Li, X.R.; Kirubarajan, T. *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software*; John Wiley & Sons: New York City, NY, USA, 2004.
28. Hoffmann, P.H. A hitchhiker’s guide to automatic differentiation. *Numer. Algorithms* **2016**, *72*, 775–811. [[CrossRef](#)]
29. Post, L.; Koller, T.L. An Automatically Differentiating Extended Kalman Filter on Boxplus Manifolds. 2020. Available online: <https://github.com/TomLKoller/ADEKF> (accessed on 16 June 2021).