

Exploiting linear substructure in linear regression Kalman filters

Greiff, Marcus; Robertsson, Anders; Berntorp, Karl

TR2020-171 December 16, 2020

Abstract

We exploit knowledge of linear substructure in the linear-regression Kalman filters (LRKFs) to simplify the problem of moment matching. The theoretical results yield quantifiable and significant computational speedups at no cost of estimation accuracy, assuming partially linear estimation models. The results apply to any symmetrical LRKF, and reductions in computational complexity are stated as a function of the cubature rule, the number of linear and nonlinear states in the estimation model respectively. The implications for the filtering problem are illustrated by several numerical examples.

IEEE Conference on Decision and Control (CDC)

Exploiting linear substructure in linear regression Kalman filters

Marcus Greiff¹, Anders Robertsson¹ and Karl Berntorp².

Abstract—We exploit knowledge of linear substructure in the linear-regression Kalman filters (LRKFs) to simplify the problem of moment matching. The theoretical results yield quantifiable and significant computational speedups at no cost of estimation accuracy, assuming partially linear estimation models. The results apply to any symmetrical LRKF, and reductions in computational complexity are stated as a function of the cubature rule, the number of linear and nonlinear states in the estimation model respectively. The implications for the filtering problem are illustrated by several numerical examples.

I. INTRODUCTION

In this paper, we explore the incorporation of known linear substructure in the linear-regression Kalman filters (LRKFs) summarised in [1], [2]. For future reference, we refer to such filters as partially linear LRKFs, or PL-LRKFs for short. Many physical systems arising from Newtonian mechanics have some partially linear substructure in the dynamical equations, and a great number of systems have a partially linear measurement model. For state estimation with such systems, the large family of particle filters (PFs) [3], [4] quickly become computationally intractable with the number of states that are to be estimated. This has historically been a major motivation for the development of Rao-Blackwellized particle filters (RBPfFs) [5], [6], which assume a particle distribution in the nonlinear states and a Gaussian distribution in the linear states. However, due to stemming from the particle filtering framework, such filters also tend to be computationally cumbersome for large numbers of states, which is why a vast majority of nonlinear estimation applications still employ Gaussian approximate density filters (ADFs), such as the extended Kalman filter (EKF) [7].

For problems where the estimate distribution is likely to be uni-modal or the RBPfF-variants are deemed computationally intractable, alternatives to the EKFs include the LRKFs. These are also Gaussian ADFs, but use various cubature rules in order to evaluate a set of moment integrals, instead of approximating the nonlinear functions by Taylor expansions, as done in the EKF. By using high-order cubature rules, these filters are often favored over the EKF for their estimation accuracy, but much like the PFs, the LRKFs do not exploit linear substructure in the nonlinear estimation models.

Consequently, we analyze the problem of moment matching, that is, computing the first two moments of the joint distribution of input and output of a nonlinear function, for

partially linear functions using various cubature rules. Specifically, we focus on the spherical cubature rule (SC) used in the cubature Kalman filter (CKF) [8], [9]; the unscented transform (UT) used in the unscented Kalman filter (UKF) in [10], [11]; the Gauss-Hermite cubature rule (GHC) used in the GHKF in [7]; and the stochastic integration rule (SIR) used in the randomized unscented Kalman filter (RUKF) in [12], [13]. However, the results apply to all symmetric LRKFs of which the aforementioned filters are but a subset. As such, this work distinguishes itself from the relevant prior work in [14], [15] in two main respects. Firstly, in its generality: we are considering all symmetric LRKFs in terms of the cubature point set, and not specific cubature rules such as the RB-UKF in [14]. Secondly, in that we are not using the conditionally linear structure [15], but a partially linear structure in the otherwise nonlinear equations.

1) *Contributions*: Apart from the numerical results, the main theoretical contributions of this paper are as follows:

- (i) An expression for the moments of the joint distribution of input and output to a nonlinear function with a linear substructure, given a generic cubature rule defined by a set of integration points and weights.
- (ii) Conditions under which the joint distribution can be evaluated with a number of integration points that scale with the number of nonlinear states instead of with the total number of states.
- (iii) Conditions under which the square root factorization in the LRKFs only needs to be computed partially.
- (iv) Proofs that the conditions in (ii) and (iii) hold for the cubature rules in the CKF, UKF and GHKF.

2) *Overview*: We start by a brief review of the LRKFs in Section II, followed by the definition of the moment matching problem in Section III. The main results are given in Section IV, and their implications for the filtering problem are subsequently illustrated in Section V. Finally, numerical results are given in VI, and Section VII closes the paper.

3) *Notation*: In the following, we let $e_i^N \in \mathbb{R}^N$ denote a unit vector with the i^{th} element set to 1, and all other elements set to zero. The vector $\mathbf{1}_N \in \mathbb{R}^N$ is a column vector of ones, the matrix $\mathbf{0}_{N \times M} \in \mathbb{R}^{N \times M}$ is a zero matrix, and $\mathbf{I}_N \in \mathbb{R}^{N \times N}$ is the identity matrix. Vectors are denoted by bold font \mathbf{x} , and sets with calligraphic font \mathcal{S} with $|\mathcal{S}|$ denoting set cardinality. Here, $\mathcal{N}(\mathbf{x}|\mathbf{m}^x, \mathbf{P}^{xx})$ denotes a Gaussian probability density function over \mathbf{x} with mean \mathbf{m}^x and covariance \mathbf{P}^{xx} , closely following the notation in [7]. The sub-indexation $(\cdot)_k$ indicates a variable at a time-step k , and the notation $(\cdot)_{a|b}$ indicates a variable at a time $k = a$ conditioned on information up until and including $k = b$.

¹Department of Automatic Control, Lund University, Lund, Sweden. {marcus.greiff, anders.robertsson}@control.lth.se

²Control and Dynamical Systems, Mitsubishi Electric Research Laboratories, Cambridge, MA, 02139, USA. berntorp@merl.com

Finally, we take \otimes to denote the usual Kronecker product, and let \star denote a redundant entry in a symmetric matrix.

II. PRELIMINARIES

We consider a discrete-time systems on the form,

$$\mathbf{x}_{k+1} = \mathbf{F}(\mathbf{x}_k, \mathbf{q}_k) \in \mathbb{R}^X, \quad (1a)$$

$$\mathbf{y}_k = \mathbf{H}(\mathbf{x}_k, \mathbf{r}_k) \in \mathbb{R}^Y, \quad (1b)$$

where the process noise and measurement noise are Gaussian distributed, with $\mathbf{q}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$, $\mathbf{r}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$, and $\mathbf{0} \prec \mathbf{Q}_k = \mathbf{Q}_k^\top$, $\mathbf{0} \prec \mathbf{R}_k = \mathbf{R}_k^\top$. The objective is to recursively estimate the state \mathbf{x}_k given $\mathbf{y}_{0:k}$ using the LRKFs. In the generic Gaussian approximate density filters (ADFs), of which the LRKFs are a subset, the distribution of the state-estimate at time step $k-1$ is approximated by a Gaussian,

$$p(\mathbf{x}_{k-1} | \mathbf{y}_{0:k-1}) \approx \mathcal{N}(\mathbf{x}_{k-1} | \mathbf{m}_{k-1}^x, \mathbf{P}_{k-1}^{xx}). \quad (2)$$

Given this approximation, the estimate distribution is propagated through the dynamics in (1a), yielding a prediction

$$p(\mathbf{x}_k | \mathbf{y}_{0:k-1}) \approx \mathcal{N}(\mathbf{x}_k | \mathbf{m}_{k|k-1}^x, \mathbf{P}_{k|k-1}^{xx}), \quad (3)$$

and the joint distribution of the predicted state and measurement is approximated based on (1b), as

$$\mathcal{N}\left(\begin{bmatrix} \mathbf{x}_{k|k-1} \\ \mathbf{y}_{k|k-1} \end{bmatrix} \middle| \begin{bmatrix} \mathbf{m}_{k|k-1}^x \\ \mathbf{m}_{k|k-1}^y \end{bmatrix}, \begin{bmatrix} \mathbf{P}_{k|k-1}^{xx} & \mathbf{P}_{k|k-1}^{xy} \\ \mathbf{P}_{k|k-1}^{yx} & \mathbf{P}_{k|k-1}^{yy} \end{bmatrix}\right). \quad (4)$$

Upon receiving a measurement $\mathbf{y}_{k|k-1} = \mathbf{y}_k$, we evaluate

$$p(\mathbf{x}_k | \mathbf{y}_{0:k}) = \mathcal{N}(\mathbf{x}_k | \mathbf{m}_{k|k}^x, \mathbf{P}_{k|k}^{xx}), \quad (5)$$

using Bayes' rule, where for the multi-variate Gaussian case,

$$\begin{aligned} \mathbf{m}_{k|k}^x &= \mathbf{m}_{k|k-1}^x + \mathbf{P}_{k|k-1}^{xy} (\mathbf{P}_{k|k-1}^{yy})^{-1} (\mathbf{y}_k - \mathbf{m}_{k|k-1}^y), \\ \mathbf{P}_{k|k}^{xx} &= \mathbf{P}_{k|k-1}^{xx} - \mathbf{P}_{k|k-1}^{xy} (\mathbf{P}_{k|k-1}^{yy})^{-1} \mathbf{P}_{k|k-1}^{yx}. \end{aligned} \quad (6)$$

The difference among all of the Gaussian ADFs lies in the way the prediction in (3) and the joint distribution in (4) are approximated. In the event of linear flow and measurement equations (1), the state-distribution will be Gaussian at all times, and the conditional distribution can be computed exactly. The above equations in (3), (4), and (6) then result in the familiar Kalman filter. However, if (1) is nonlinear, many forms of approximations can be considered in (3) and (4). Here, direct approximation of the associated moment integrals, results in the large family of the linear regression Kalman filters (LRKFs) [1]. This is commonly referred to as moment matching, and we will now analyze how knowledge of linear substructure in (1) can be leveraged to simplify the numerical evaluation of these moment integrals in the context of the aforementioned SC, UT, GHC and SIR schemes.

III. THE PROBLEM OF MOMENT MATCHING

In the following, we simplify the notation by temporarily dropping the time indexation and only considering the computation of the first and second moments of the joint distribution of input and output to a nonlinear function with linear substructure. To investigate potential simplifications in the moment matching, consider a function $\mathbf{G} : \mathbb{R}^X \rightarrow \mathbb{R}^Y$,

operating on a state $\mathbf{x} \in \mathbb{R}^X$, which can be partitioned into a linear part $\mathbf{l} \in \mathbb{R}^L$ and a nonlinear part $\mathbf{z} \in \mathbb{R}^Z$,

$$\mathcal{N}(\mathbf{x} | \mathbf{m}^x, \mathbf{P}^{xx}) = \mathcal{N}\left(\begin{bmatrix} \mathbf{z} \\ \mathbf{l} \end{bmatrix} \middle| \begin{bmatrix} \mathbf{m}^z \\ \mathbf{m}^l \end{bmatrix}, \begin{bmatrix} \mathbf{P}^{zz} & \mathbf{P}^{zl} \\ \mathbf{P}^{lz} & \mathbf{P}^{ll} \end{bmatrix}\right), \quad (7)$$

where the \mathbf{G} has some linear substructure on the form

$$\mathbf{y} = \mathbf{G}(\mathbf{x}) \triangleq \begin{bmatrix} \mathbf{g}(\mathbf{z}) \\ \mathbf{A}\mathbf{x} \end{bmatrix}. \quad (8)$$

In essence, only knowing the structure in (7) and (8), we seek a simplified moment-matching of the joint density

$$\mathcal{N}\left(\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \middle| \begin{bmatrix} \mathbf{m}^x \\ \mathbf{m}^y \end{bmatrix}, \begin{bmatrix} \mathbf{P}^{xx} & \mathbf{P}^{xy} \\ \mathbf{P}^{yx} & \mathbf{P}^{yy} \end{bmatrix}\right), \quad (9)$$

by approximate evaluation of the moment integrals, cf. [16],

$$\mathbf{m}^y = \int_{\mathbb{R}^X} \mathbf{G}(\mathbf{x}) \mathcal{N}(\mathbf{x} | \mathbf{m}^x, \mathbf{P}^{xx}) d\mathbf{x}, \quad (10)$$

$$\mathbf{P}^{xy} = \int_{\mathbb{R}^X} (\mathbf{x} - \mathbf{m}^x) (\mathbf{G}(\mathbf{x}) - \mathbf{m}^y)^\top \mathcal{N}(\mathbf{x} | \mathbf{m}^x, \mathbf{P}^{xx}) d\mathbf{x},$$

$$\mathbf{P}^{yy} = \int_{\mathbb{R}^X} (\mathbf{G}(\mathbf{x}) - \mathbf{m}^y) (\mathbf{G}(\mathbf{x}) - \mathbf{m}^y)^\top \mathcal{N}(\mathbf{x} | \mathbf{m}^x, \mathbf{P}^{xx}) d\mathbf{x}.$$

The question is how to leverage knowledge of the linear substructure in (7) when evaluating (10), and what implications this has for the resulting filtering problem in Section II.

A. Approximate moment matching

Moving forward, we assume a non-degenerate distribution over \mathbf{x} , with $\mathbf{0} \prec \mathbf{P}^{xx} = \mathbf{P}^{xx\top}$, such that there exists

$$\mathbf{P}^{xx} = \mathbf{L}^{xx} \mathbf{L}^{xx\top}, \quad \mathbf{L}^{xx} = \begin{bmatrix} \mathbf{L}^{zz} & \mathbf{0} \\ \mathbf{L}^{lz} & \mathbf{L}^{ll} \end{bmatrix}, \quad (11)$$

where \mathbf{L}^{xx} , \mathbf{L}^{zz} , \mathbf{L}^{ll} are all lower-triangular. Given this, there are many ways of approximating the moment integrals. However, all of the prior mentioned schemes first take a coordinate transform $\boldsymbol{\xi} = (\mathbf{L}^{xx})^{-1}(\mathbf{x} - \mathbf{m}^x)$, and then approximate the moment integrals in a finite sum

$$\begin{aligned} \int_{\mathbb{R}^X} \mathbf{G}(\mathbf{x}) \mathcal{N}(\mathbf{x} | \mathbf{m}^x, \mathbf{P}^{xx}) d\mathbf{x} &= \int_{\mathbb{R}^X} \mathbf{G}(\mathbf{m}^x + \mathbf{L}^{xx} \boldsymbol{\xi}) \mathcal{N}(\boldsymbol{\xi} | \mathbf{0}, \mathbf{I}) d\boldsymbol{\xi} \\ &\approx \sum_{i=1}^{C(X)} w^{(i)} \mathbf{G}(\mathbf{m}^x + \mathbf{L}^{xx} \boldsymbol{\xi}^{(i)}) \end{aligned} \quad (12)$$

by a total of $C(X)$ pairs of weights and integration points $\mathcal{P} = \{(w^{(i)}, \boldsymbol{\xi}^{(i)})\}_{i=1}^{C(X)}$, where the cardinality $C(X) = |\mathcal{P}|$ increases with $X = \dim(\mathbf{x})$. In this notation, the LRKF moment approximations can be written on the form

$$\mathcal{X}^{(i)} = \mathbf{m}^x + \mathbf{L}^{xx} \boldsymbol{\xi}^{(i)} \quad i \in \{1, \dots, |\mathcal{P}|\}, \quad (13a)$$

$$\mathcal{Y}^{(i)} = \mathbf{G}(\mathcal{X}^{(i)}) \quad i \in \{1, \dots, |\mathcal{P}|\}, \quad (13b)$$

$$\mathbf{m}^y \approx \sum_{i=1}^{|\mathcal{P}|} w^{(i)} \mathcal{Y}^{(i)}, \quad (13c)$$

$$\mathbf{P}^{xy} \approx \sum_{i=1}^{|\mathcal{P}|} w^{(i)} (\mathcal{X}^{(i)} - \mathbf{m}^x) (\mathcal{Y}^{(i)} - \mathbf{m}^y)^\top, \quad (13d)$$

$$\mathbf{P}^{yy} \approx \sum_{i=1}^{|\mathcal{P}|} w^{(i)} (\mathcal{Y}^{(i)} - \mathbf{m}^y) (\mathcal{Y}^{(i)} - \mathbf{m}^y)^\top. \quad (13e)$$

To proceed with the analysis, we start with some additional definitions. Let $\mathbf{N} = [\mathbf{I}_Z \ \mathbf{0}_{Z \times L}]$ and $\bar{\mathbf{N}} = [\mathbf{0}_{L \times Z} \ \mathbf{I}_L]$, such that $\mathbf{N}\mathbf{x} = \mathbf{z}$ and $\bar{\mathbf{N}}\mathbf{x} = \mathbf{l}$. For simplicity, in analyzing the cubature rules in the context of the linear substructure in (8), we further categorize \mathcal{P} into three categories; central (c), linear (l), and nonlinear (z). The points where $\boldsymbol{\xi} = \mathbf{0}$, we refer to as central points with a sub-index $(\cdot)_c$; if $\boldsymbol{\xi}$ is at the origin in the dimensions corresponding to the input of the nonlinear function \mathbf{g} used to define \mathbf{G} in (8), we refer to these points as the linear points with a sub-index $(\cdot)_l$, as they only differ from the origin in the linear dimensions of the state $\mathbf{l} \subseteq \mathbf{x} \in \mathbb{R}^X$; if $\boldsymbol{\xi}$ differs from the origin in the input to \mathbf{g} , the points and weights are sub-indexed $(\cdot)_z$. In the following, we let a denote a sub-index c, z , or l , and let

$$\mathcal{P}_c = \{(w, \boldsymbol{\xi}) \in \mathcal{P} | \boldsymbol{\xi} = \mathbf{0}\}, \quad (14a)$$

$$\mathcal{P}_l = \{(w, \boldsymbol{\xi}) \in \mathcal{P} \setminus \mathcal{P}_c | \mathbf{N}\boldsymbol{\xi} = \mathbf{0}\}, \quad (14b)$$

$$\mathcal{P}_z = \mathcal{P} \setminus (\mathcal{P}_c \cup \mathcal{P}_l). \quad (14c)$$

Here, for any of the point sets \mathcal{P}_a , we let

$$\begin{cases} w_a^{(i)} = w^{(i)}, \\ \Xi_a^{(i)} = \boldsymbol{\xi}^{(i)}, \\ \mathcal{X}_a^{(i)} = \mathbf{m}^{\mathbf{x}} + \mathbf{L}^{\mathbf{x}\mathbf{x}} \boldsymbol{\xi}^{(i)} \\ \mathcal{Z}_a^{(i)} = \mathbf{N} \mathcal{X}_a^{(i)} \end{cases} \quad \forall (w^{(i)}, \boldsymbol{\xi}^{(i)}) \in \mathcal{P}_a, \quad (15)$$

and define associated matrices

$$\mathbf{w}_a = [w_a^{(1)} \ \dots \ w_a^{(|\mathcal{P}_a|)}] \in \mathbb{R}^{1 \times |\mathcal{P}_a|}, \quad (16a)$$

$$\boldsymbol{\Xi}_a = [\Xi_a^{(1)} \ \dots \ \Xi_a^{(|\mathcal{P}_a|)}] \in \mathbb{R}^{X \times |\mathcal{P}_a|}, \quad (16b)$$

$$\boldsymbol{\mathcal{X}}_a = [\mathcal{X}_a^{(1)} \ \dots \ \mathcal{X}_a^{(|\mathcal{P}_a|)}] \in \mathbb{R}^{X \times |\mathcal{P}_a|}, \quad (16c)$$

$$\boldsymbol{\mathcal{Z}}_a = [\mathcal{Z}_a^{(1)} \ \dots \ \mathcal{Z}_a^{(|\mathcal{P}_a|)}] \in \mathbb{R}^{Z \times |\mathcal{P}_a|}. \quad (16d)$$

We also define the combined matrices,

$$\mathbf{w} = [\mathbf{w}_c \ \mathbf{w}_z \ \mathbf{w}_l], \quad \mathcal{W} = \text{diag}(\mathbf{w}), \quad (17a)$$

$$\boldsymbol{\Xi} = [\boldsymbol{\Xi}_c \ \boldsymbol{\Xi}_z \ \boldsymbol{\Xi}_l], \quad \mathcal{W}_z = \text{diag}(\mathbf{w}_z), \quad (17b)$$

$$\boldsymbol{\mathcal{X}} = [\boldsymbol{\mathcal{X}}_c \ \boldsymbol{\mathcal{X}}_z \ \boldsymbol{\mathcal{X}}_l], \quad \boldsymbol{\mathcal{Z}} = [\boldsymbol{\mathcal{Z}}_c \ \boldsymbol{\mathcal{Z}}_z \ \boldsymbol{\mathcal{Z}}_l]. \quad (17c)$$

These definitions are illustrated in Figure 1, to which we shall return in the next section.

IV. MAIN RESULT

We now state some basic properties common to the prior mentioned cubature rules, which will be verified for the SCR, UT, GHCR and SIR, but certainly encompassing more of the LRFs. We will then proceed to use these properties in the simplification of the evaluation of the joint distributions.

Assumption 1. *Symmetry, such that $\forall (w^{(i)}, \boldsymbol{\xi}^{(i)}) \in \mathcal{P}_z \cup \mathcal{P}_l$, $\exists (w^{(j)}, \boldsymbol{\xi}^{(j)}) \in \mathcal{P}_z \cup \mathcal{P}_l$ s.t. $(w^{(i)}, \boldsymbol{\xi}^{(i)}) = (w^{(j)}, -\boldsymbol{\xi}^{(j)})$.*

Assumption 2. *Consistency in the first moment for linear maps, with $\sum_{(w, \boldsymbol{\xi}) \in \mathcal{P}} w = 1$.*

Assumption 3. *Consistency in the second moment for linear maps, with $\boldsymbol{\Xi} \mathcal{W} \boldsymbol{\Xi}^\top = \mathbf{I}$.*

A. The spherical cubature rule

The spherical cubature rule used in the CKF, originally presented in [8], scales as $|\mathcal{P}| = C(X) = 2X$, and is defined by a set of points where $\mathcal{P}_c = \emptyset$ implying that $\mathcal{P} = \mathcal{P}_z \cup \mathcal{P}_l$, and the weights and integration points are given by

$$\mathbf{w} = (2X)^{-1} \mathbf{1}_{2X}^\top, \quad \boldsymbol{\Xi} = \sqrt{X} [\mathbf{I} \ -\mathbf{I}]. \quad (18)$$

Remark 1. *Assumption 1 clearly holds. As $\sum_{(w, \boldsymbol{\xi}) \in \mathcal{P}} w = 2X(2X)^{-1} = 1$, and $\boldsymbol{\Xi} \mathcal{W} \boldsymbol{\Xi}^\top = (2X)^{-1} (X\mathbf{I} + X\mathbf{I}) = \mathbf{I}$, showing that both Assumptions 2 and 3 also hold.*

B. The unscented transform

The unscented transform, used to define the celebrated UKF filter originally presented in [10], scales as $|\mathcal{P}| = C(X) = 2X + 1$. This transform is determined by a length scale parameter $\lambda = \alpha^2(X + \kappa) - X$, for some constant $\alpha, \kappa > 0$. The UT has one central integration point, with $\mathcal{P}_c = (w_c^{(1)}, \boldsymbol{\xi}_c^{(1)}) = (\lambda(\lambda + X)^{-1}, \mathbf{0})$, and \mathcal{P} is given by

$$\mathbf{w} = \frac{1}{\lambda + X} [\lambda \ \frac{1}{2} \mathbf{1}_{2X}^\top], \quad \boldsymbol{\Xi} = \sqrt{(\lambda + X)} [\mathbf{0}_{X \times 1} \ \mathbf{I}_X \ -\mathbf{I}_X]$$

Remark 2. *For every point in $\mathcal{P}_z \cup \mathcal{P}_l$, Assumption 1 holds. Furthermore, $\sum_{(w, \boldsymbol{\xi}) \in \mathcal{P}} w = \lambda(\lambda + X)^{-1} + 2X(2(\lambda + X))^{-1} = (\lambda + X)(\lambda + X)^{-1} = 1$, thus Assumption 2 holds. Finally, as $\boldsymbol{\Xi} \mathcal{W} \boldsymbol{\Xi}^\top = \lambda(\lambda + X)^{-1} \mathbf{0}_{X \times X} + (2(\lambda + X))^{-1} ((\lambda + X)\mathbf{I}_X + (\lambda + X)\mathbf{I}_X) = \mathbf{I}_X$, Assumption 3 holds.*

C. The Gauss-Hermite cubature rule

Another form of cubature is the GHC, first presented in [17] and used to construct the first GHKF in [16], comprehensively summarized in [7]. Here, $H_p(x)$ denotes the probabilistic Hermite polynomial of order p , as

$$H_0 = 1, \quad H_1 = x, \quad H_{p+1}(x) = xH_p(x) - pH_{p-1}(x), \quad (19)$$

and r_p^i denotes the i^{th} root of $H_p(x)$. For each root, we compute an associated number

$$\alpha_p^i = p!(pH_{p-1}(r_p^i))^{-2}. \quad (20)$$

In the one-dimensional quadrature, r_p^i and α_p^i would form the set of integration points and weights. However, this can easily be extended to a multi-dimensional cubature, then with the set of weights and points defined as

$$\mathcal{P} = \left\{ \left(\prod_{j=1}^X \alpha_p^{k_j}, \sum_{j=1}^X r_p^{k_j} \mathbf{e}_j^X \right) \mid k_j \in \{1, \dots, p\} \forall j \in \{1, \dots, X\} \right\}.$$

Thus, the number of integration points scales exponentially with the dimension of the state, with $C(X) = p^X$.

Remark 3. *The Hermite polynomials are symmetric, that is, for every $r_p^i \neq 0$, $\exists r_p^j$ such that $r_p^i = -r_p^j$ and since $H_p(x) = (-1)^p H_p(-x)$, we have that $\alpha_p^i = \alpha_p^j$ for each such pair of roots. Thus, for every point in $\mathcal{P}_z \cup \mathcal{P}_l$, Assumption 1 clearly holds. To see that Assumption 3 is satisfied, let $\mathbf{g}(\mathbf{x}) = \mathbf{x}$ and take $\mathbf{m}^{\mathbf{x}} = \mathbf{0}$ and $\mathbf{P}^{\mathbf{x}\mathbf{x}} = \mathbf{I}$. In this case, $\mathbf{P}^{\mathbf{y}\mathbf{y}} = \mathbf{I}$, and, since the p^{th} -order Gauss-Hermite cubature rule is exact for polynomials of order p [16], and the second moment in this case is a second order polynomial in \mathbf{x} with a Gaussian weight function, we have that $\boldsymbol{\Xi} \mathcal{W} \boldsymbol{\Xi}^\top = \mathbf{I}$ for any GHC of order $p \geq 2$. Similarly, Assumption 2 holds for all $p \geq 1$.*

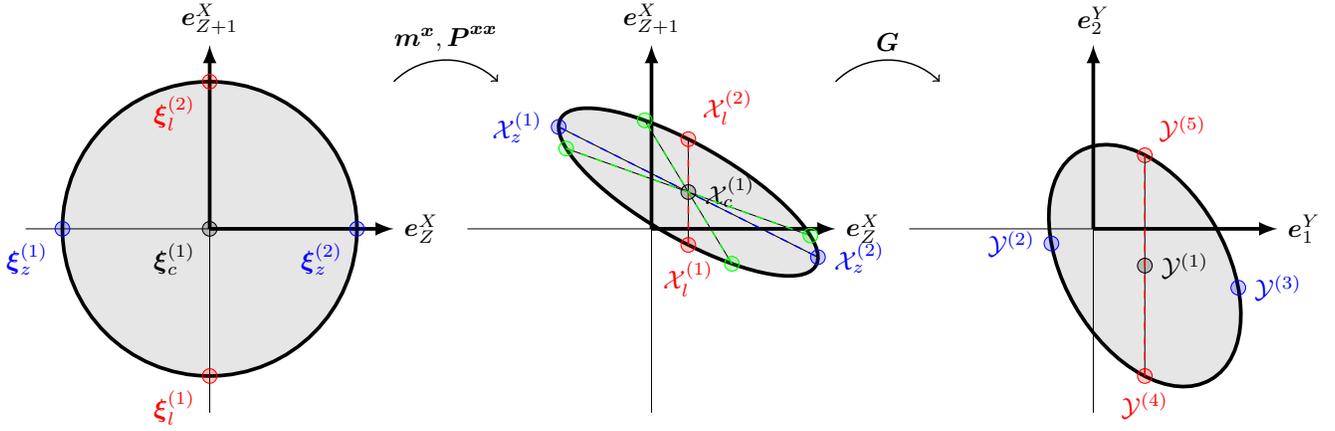


Fig. 1. Illustration of the set Ξ partitioned into the sets $\mathcal{P}_c, \mathcal{P}_z, \mathcal{P}_l$ for the unscented transform (left). Resulting integration points \mathcal{X} , here computed with a Cholesky decomposition L^{xx} (blue/red), and also with a symmetric square root factorization (green).

D. Exploiting the linear substructure

We will now attempt to exploit the linear substructure in (8) under the assumption that conditions 1, 2 and 3 all hold. The key idea here is to partition the state as done in (7), and then use the lower-triangular Cholesky decomposition in the change of variables in (12), and subsequent evaluation of the set \mathcal{X} , instead of a symmetric square root factorization, otherwise commonly used in the LRKFs. This special structure ensures that for any element in the set $\mathcal{P}_c \cup \mathcal{P}_l$, we have that $\mathbf{N}\mathcal{Y}^{(i)} = \mathbf{N}\mathbf{G}(\mathcal{X}^{(i)}) = \mathbf{g}(\mathcal{Z}^{(i)}) = \mathbf{g}(\mathbf{m}^z)$. This property does not hold for the symmetric square root factorization, as illustrated in Figure 1, and with it in mind, we can proceed by attacking the expressions in (13) using Assumptions 1, 2, and 3. The final result is stated in Theorem 1 and the Corollaries 1-3, with proof ideas in the Appendix and complete proofs in an extended paper [18].

Theorem 1. For the state in (7), its joint distribution with output of the structured nonlinear function in (8),

$$\mathcal{N}\left(\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \middle| \begin{bmatrix} \mathbf{m}^x \\ \mathbf{m}^y \end{bmatrix}, \begin{bmatrix} \mathbf{P}^{xx} & \mathbf{P}^{xy} \\ \mathbf{P}^{yx} & \mathbf{P}^{yy} \end{bmatrix}\right), \quad (21)$$

computed using a cubature rule defined by a set of points and weights $\mathcal{P} = \{(w^{(i)}, \xi^{(i)})\}_{i=1}^{C(X)}$ satisfying assumptions 1, 2, and 3, the moments of the joint distribution is given by

$$\begin{aligned} \mathbf{m}^y &= \begin{bmatrix} w_{cl}\mathbf{g}(\mathbf{m}^z) + \mathcal{G}_z \mathbf{w}_z^\top \\ \mathbf{A} \mathbf{m}^x \end{bmatrix}, & (22) \\ \mathbf{P}^{xy} &= \begin{bmatrix} L^{xx} \Xi_z \mathcal{W}_z \mathcal{G}_z^\top & \mathbf{P}^{xx} \mathbf{A}^\top \\ \mathbf{P}^{yx} & \end{bmatrix}, \\ \mathbf{P}^{yy} &= \begin{bmatrix} (\mathcal{G}_z + \mathbf{C}_z) \mathcal{W}_z (\mathcal{G}_z + \mathbf{C}_z)^\top + w_{cl} \mathbf{u}_l \mathbf{u}_l^\top & \star \\ \mathbf{A} L^{xx} \Xi_z \mathcal{W}_z \mathcal{G}_z^\top & \mathbf{A} \mathbf{P}^{xx} \mathbf{A}^\top \end{bmatrix}, \end{aligned}$$

with $\mathcal{P}_z, \mathbf{w}_z, \mathcal{W}_z, \Xi_z$ given in (14)-(17), and

$$w_{cl} = 1 - \mathbf{w}_z \mathbf{1}_{|\mathcal{P}_z|} \quad (23a)$$

$$\mathcal{G}_z = [\mathbf{g}(\mathcal{Z}^{(1)}) \quad \dots \quad \mathbf{g}(\mathcal{Z}^{(|\mathcal{P}_z|)})] \quad (23b)$$

$$\mathbf{u} = -w_{cl} \mathbf{g}(\mathbf{m}^z) - \mathcal{G}_z \mathbf{w}_z^\top \quad (23c)$$

$$\mathbf{u}_l = \mathbf{g}(\mathbf{m}^z) + \mathbf{u} \quad (23d)$$

$$\mathbf{C}_z = \mathbf{u} \mathbf{1}_{|\mathcal{P}_z|}^\top \quad (23e)$$

Corollary 1. If, $\forall (w^{(i)}, \xi^{(i)}) \in \mathcal{P}_z \cup \mathcal{P}_l, \exists (w^{(j)}, \xi^{(j)}) \in \mathcal{P}_z \cup \mathcal{P}_l$ s.t. $(w^{(i)}, \mathbf{N}\xi^{(i)}, \bar{\mathbf{N}}\xi^{(i)}) = (w^{(j)}, \mathbf{N}\xi^{(j)}, -\bar{\mathbf{N}}\xi^{(j)})$, which is clearly satisfied in the SC, UT, and GHC, then Theorem 1 only requires evaluation of the nonlinear function \mathbf{g} in $C(Z)$ points instead of evaluating the function \mathbf{G} in $C(X)$ points as done in the original cubature rules.

Corollary 2. If $\bar{\mathbf{N}}\Xi_z = \mathbf{0}$, as is the case with the SC and UT, then the joint distribution (22) in Theorem 1 can be evaluated with a partial column-wise Cholesky decomposition of \mathbf{P}^{xx} . Indeed, only the first Z columns of L^{xx} are needed.

Corollary 3. The results for the substructure in (7) can be generalized to structured nonlinear functions on the form

$$\mathbf{y} = \begin{bmatrix} \mathbf{A}_1 \mathbf{x} + \mathbf{g}(z) \\ \mathbf{A}_2 \mathbf{x} \end{bmatrix}. \quad (24)$$

V. IMPLICATIONS FOR THE FILTERING PROBLEM

To illustrate the implications for the filtering problem, we start by defining the regular LRFK filters in the context of the notation in Section II. Here, we will use a given cubature rule in approximating the moment integrals, both for the flow function $\mathbf{F} : \mathbb{R}^X \rightarrow \mathbb{R}^X$ and the measurement function $\mathbf{H} : \mathbb{R}^X \rightarrow \mathbb{R}^Y$. As such, the linear and nonlinear states may not be the same for the two functions. Consequently, we assume that the functions have an associated orthogonal permutation matrix, here denoted \mathbf{T}_F and \mathbf{T}_H respectively, such that

$$\begin{bmatrix} \bar{\mathbf{z}} \\ \bar{\mathbf{l}} \end{bmatrix} = \bar{\mathbf{x}} = \mathbf{T}_F \mathbf{x}, \quad (25)$$

yields a permuted state vector where the $\bar{\mathbf{z}}$ states are nonlinear in \mathbf{F} , and the $\bar{\mathbf{l}}$ are linear in \mathbf{F} . If the state is Gaussian distributed with $\mathcal{N}(\mathbf{x} | \mathbf{m}^x, \mathbf{P}^{xx})$, then this results in a change of both the mean and covariance, with $\mathcal{N}(\bar{\mathbf{x}} | \bar{\mathbf{m}}^{\bar{\mathbf{x}}}, \mathbf{P}^{\bar{\mathbf{x}}\bar{\mathbf{x}}}) = \mathcal{N}(\bar{\mathbf{x}} | \mathbf{T}_F \mathbf{m}^x, \mathbf{T}_F \mathbf{P}^{xx} (\mathbf{T}_F)^\top)$. For future reference, we let the transformation of the moments of the state \mathbf{x} by a transformation \mathbf{T}_F be denoted $\{\bar{\mathbf{m}}^{\bar{\mathbf{x}}}, \mathbf{P}^{\bar{\mathbf{x}}\bar{\mathbf{x}}}\} = \text{permute}(\mathbf{T}_F, \mathbf{m}^x, \mathbf{P}^{xx})$. Similarly, we will have a different set of points, \mathcal{P} , for the flow and measurement equations, and these are simply defined as

\mathcal{P}_F and \mathcal{P}_H respectively. Furthermore, we denote the full Cholesky factorization of a matrix by $\mathbf{L}^{xx} = \text{chol}(\mathbf{P}^{xx})$, and the Cholesky-Crout algorithm computing the first Z columns of the square-root factorization as $\{\mathbf{L}^{zz}, \mathbf{L}^{zl}\} = \text{chol_crout}(\mathbf{P}^{xx})$. The generic LRKF is given in Algorithm 1, to be compared with the PL-LRKF in Algorithm 2.

Algorithm 1 The generic LRKF

```

1: Initialize  $\mathcal{N}(\hat{\mathbf{x}}_0 | \mathbf{m}_0^{\hat{x}}, \mathbf{P}_0^{\hat{x}}), \mathcal{P}_F, \mathcal{P}_H$ 
2: for  $k = 1$  to  $K$  do
    // Time update
3:  $\mathbf{L}_{k-1}^{\hat{x}\hat{x}} = \text{chol}(\mathbf{P}_{k-1}^{\hat{x}\hat{x}})$ 
4: Compute  $\mathcal{X}_{k-1}$  by (15) using  $\mathbf{m}_{k-1}^{\hat{x}}, \mathbf{L}_{k-1}^{\hat{x}\hat{x}}, \mathcal{P}_F$ 
5: Evaluate  $\mathbf{m}_{k|k-1}^{\hat{x}}$  and  $\mathbf{P}_{k|k-1}^{\hat{x}\hat{x}}$  using (13) for (1a)
    // Measurement update
6:  $\mathbf{L}_{k|k-1}^{\hat{x}\hat{x}} = \text{chol}(\mathbf{P}_{k|k-1}^{\hat{x}\hat{x}})$ 
7: Compute  $\mathcal{X}_{k|k-1}$  by (15) using  $\mathbf{m}_{k|k-1}^{\hat{x}}, \mathbf{L}_{k|k-1}^{\hat{x}\hat{x}}, \mathcal{P}_H$ 
8: Evaluate  $\{\mathbf{m}_{k|k-1}^{\hat{y}}, \mathbf{P}_{k|k-1}^{\hat{y}\hat{y}}, \mathbf{P}_{k|k-1}^{\hat{y}\hat{x}}\}$  using (13) for the
    function in (1b) with the point set in  $\mathcal{P}_H$ 
9: Evaluate  $\{\mathbf{m}_k^{\hat{x}}, \mathbf{P}_k^{\hat{x}\hat{x}}\}$  using (6)
10: end for

```

Algorithm 2 The generic PL-LRKF

```

1: Initialize  $\mathcal{N}(\hat{\mathbf{x}}_0 | \mathbf{m}_0^{\hat{x}}, \mathbf{P}_0^{\hat{x}}), \mathcal{P}_F, \mathbf{T}_F, \mathcal{P}_H, \mathbf{T}_H$ 
2: for  $k = 1$  to  $K$  do
    // Time update
3:  $\{\mathbf{m}_{k-1}^{\hat{x}}, \mathbf{P}_{k-1}^{\hat{x}\hat{x}}\} = \text{permute}(\mathbf{T}_F, \mathbf{m}_{k-1}^{\hat{x}}, \mathbf{P}_{k-1}^{\hat{x}\hat{x}})$ 
4:  $\{\mathbf{L}_{k-1}^{\hat{z}\hat{z}}, \mathbf{L}_{k-1}^{\hat{z}\hat{l}}\} = \text{chol\_crout}(\mathbf{P}_{k-1}^{\hat{x}\hat{x}})$ 
5: Compute  $\mathcal{Z}_{k-1}$  by (15) using  $\mathbf{m}_{k-1}^{\hat{z}}, \mathbf{L}_{k-1}^{\hat{z}\hat{z}}, \mathcal{P}_F$ 
6: Evaluate  $\mathbf{m}_{k|k-1}^{\hat{x}}$  and  $\mathbf{P}_{k|k-1}^{\hat{x}\hat{x}}$  by Theorem 1 with (1a)
7:  $\{\mathbf{m}_{k|k-1}^{\hat{x}}, \mathbf{P}_{k|k-1}^{\hat{x}\hat{x}}\} = \text{permute}(\mathbf{T}_F^\top, \mathbf{m}_{k|k-1}^{\hat{x}}, \mathbf{P}_{k|k-1}^{\hat{x}\hat{x}})$ 
    // Measurement update
8:  $\{\mathbf{m}_{k|k-1}^{\hat{x}}, \mathbf{P}_{k|k-1}^{\hat{x}\hat{x}}\} = \text{permute}(\mathbf{T}_H, \mathbf{m}_{k|k-1}^{\hat{x}}, \mathbf{P}_{k|k-1}^{\hat{x}\hat{x}})$ 
9:  $\{\mathbf{L}_{k|k-1}^{\hat{z}\hat{z}}, \mathbf{L}_{k|k-1}^{\hat{z}\hat{l}}\} = \text{chol\_crout}(\mathbf{P}_{k|k-1}^{\hat{x}\hat{x}})$ 
10: Compute  $\mathcal{Z}_{k|k-1}$  by (15) using  $\mathbf{m}_{k|k-1}^{\hat{z}}, \mathbf{L}_{k|k-1}^{\hat{z}\hat{z}}, \mathcal{P}_H$ 
11: Evaluate  $\{\mathbf{m}_{k|k-1}^{\hat{y}}, \mathbf{P}_{k|k-1}^{\hat{y}\hat{y}}, \mathbf{P}_{k|k-1}^{\hat{y}\hat{x}}\}$  using Theorem 1
    for the function in (1b) with the point set  $\mathcal{P}_H$ 
12: Evaluate  $\{\mathbf{m}_k^{\hat{x}}, \mathbf{P}_k^{\hat{x}\hat{x}}\}$  using (6)
13:  $\{\mathbf{m}_k^{\hat{x}}, \mathbf{P}_k^{\hat{x}\hat{x}}\} = \text{permute}(\mathbf{T}_H^\top, \mathbf{m}_k^{\hat{x}}, \mathbf{P}_k^{\hat{x}\hat{x}})$ 
14: end for

```

VI. NUMERICAL RESULTS

A. Complexity analysis

Clearly, the permutations \mathbf{T} used in the PL-LRKF only imply a change in indexing, and little if any extra computational cost. Consequently, the main difference in the two algorithms follows from Corollary 1 and 2. Executing the PL-LRKF vs the LRKF should imply a significant reduction in the computational complexity if $Z_F = \dim(\bar{\mathbf{z}}_k) \leq X$ and $Z_H = \dim(\bar{\mathbf{z}}_{k|k-1}) \leq X$. Here, depending on the filter used and its associate scaling $C(X)$, the number of function evaluations required to execute the filter will be reduced by a factor $(C(Z_F) + C(Z_H))(2C(X))^{-1}$. This potentially yields an extreme reduction in complexity for filters like the GHKF where $C(X)$ is exponential. In addition, Theorem 1 only requires evaluation of the nonlinear part of the flow and measurement function, and not repeated evaluations of

the full functions \mathbf{F} and \mathbf{H} , further reducing the numerical complexity. Finally, if the conditions in Corollary 2 are met, significant computational gains will be made as there will be no need for evaluating the full Cholesky decomposition.

To illustrate the implications of exploiting the linear substructure in the moment matching, we form a nonlinear function $\mathbf{G}(\mathbf{x})$ defined as in (8), here with random dense matrix $\mathbf{A} \in \mathbb{R}^{L \times X}$ and let $\mathbf{g}(\mathbf{z}) = \mathbf{z} + \|\mathbf{z}\|_2 \mathbf{1}_Z \in \mathbb{R}^Z$. We execute the moment approximations using:

- (A) The original cubature rules defined in Section IV;
- (B) The corresponding PL cubature rules in Theorem 1.

A set of 10^6 moment approximations are done for various pairs of (Z, L) . In order to check the correctness of Theorem 1, we compare the moments by a metric $\Delta(\mathbf{m}^y) = \mathbb{E}[\|\mathbf{m}^{y,A} - \mathbf{m}^{y,B}\|_2]$, where $\mathbf{m}^{y,A}$ and $\mathbf{m}^{y,B}$ denotes a moment, here the output mean, as computed by (A) and (B), respectively. In addition, we let t_A and t_B denote the mean computational time with (A) and (B), respectively. The result is shown in Tables I, II, and III. Note that many of the fields in Table III could not be filled, as the system ran out of memory when storing all of the integration points. In all of the tested moment approximations, $\Delta(\mathbf{m}^y), \Delta(\mathbf{P}^{xy}), \Delta(\mathbf{P}^{yy})$ were all zero down to numerical precision. This shows the significant computational gains that can be made in using Theorem 1 to exploit knowledge of linear substructure, and also indicates that the commonly dismissed GHKF can be used for high-dimensional state estimation provided the number of non-linear states Z is relatively small.

TABLE I
MEAN COMPUTATIONAL TIME AND RELATIVE SPEED OF (A) CKF AND (B) THE PL-CKF IN 10^6 MOMENT APPROXIMATIONS.

(Z/L)	t_A	t_B	t_A/t_B
(3/10)	$1.54 \cdot 10^{-4}$	$7.64 \cdot 10^{-5}$	2.01
(3/100)	$4.21 \cdot 10^{-3}$	$4.51 \cdot 10^{-4}$	9.34
(3/1000)	1.50	$1.30 \cdot 10^{-1}$	11.66
(50/100)	$9.43 \cdot 10^{-3}$	$3.23 \cdot 10^{-3}$	2.91
(50/1000)	1.69	$1.15 \cdot 10^{-1}$	11.51

TABLE II
MEAN COMPUTATIONAL TIME AND RELATIVE SPEED OF (A) UKF AND (B) THE PL-UKF IN 10^6 MOMENT APPROXIMATIONS.

(Z/L)	t_A	t_B	t_A/t_B
(3/10)	$1.51 \cdot 10^{-4}$	$7.88 \cdot 10^{-5}$	2.00
(3/100)	$4.15 \cdot 10^{-3}$	$4.23 \cdot 10^{-4}$	9.80
(3/1000)	1.57	$1.20 \cdot 10^{-1}$	13.06
(50/100)	$9.86 \cdot 10^{-3}$	$2.29 \cdot 10^{-3}$	4.30
(50/1000)	1.77	$1.39 \cdot 10^{-1}$	12.78

TABLE III
MEAN COMPUTATIONAL TIME AND RELATIVE SPEED OF (A) GHCR AND (B) THE PL-GHCR IN 10^6 MOMENT APPROXIMATIONS.

(Z/L)	t_A	t_B	t_A/t_B
(3/3)	$5.11 \cdot 10^{-3}$	$1.24 \cdot 10^{-4}$	$4.12 \cdot 10^1$
(3/4)	$1.82 \cdot 10^{-2}$	$1.12 \cdot 10^{-4}$	$1.62 \cdot 10^2$
(3/5)	$2.12 \cdot 10^{-1}$	$1.23 \cdot 10^{-4}$	$1.73 \cdot 10^3$
(3/10)	-	$1.70 \cdot 10^{-4}$	-
(3/100)	-	$2.50 \cdot 10^{-3}$	-

B. Application example

Next, we give an example where Algorithm 2 can be of particular use. Imagine a scenario where N agents with positions $\mathbf{p}_k^i \in \mathbb{R}^3$, $i = 1, \dots, N$, velocities $\mathbf{v}_k^i \in \mathbb{R}^3$ and accelerations $\mathbf{a}_k^i \in \mathbb{R}^3$ in the vicinity of a base station located at $\mathbf{p}^B = \mathbf{0} \in \mathbb{R}^3$. The agents could here be ground or aerial vehicles, and for the sake of generality, suppose their motions are described by a discrete time Singer model, defined as

$$\mathbf{x}_{k+1}^i = \mathbf{A}_k^i \mathbf{x}_k^i + \mathbf{q}_k^i, \quad \mathbf{q}_k^i \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k^i), \quad (26)$$

where \mathbf{A}_k^i and \mathbf{Q}_k^i are linear maps, stated explicitly in [19].

Each of the agents estimate its own states independently based on local information. This could be done using a wide variety of algorithms, and we assume this local estimate can be represented in its first two moments, as $\mathcal{N}(\mathbf{x}_k^i | \mathbf{m}_k^{\hat{\mathbf{x}}^i}, \mathbf{P}_k^{\hat{\mathbf{x}}^i})$, and that this information is occasionally transmitted to a base station. The base station subsequently fuses this information together with two-dimensional bearing angle measurements taken of each agent. In the base station, the state vector therefore takes the form $\mathbf{x}_k = [(\mathbf{x}_k^1)^\top, \dots, (\mathbf{x}_k^N)^\top]^\top$, with the estimation model

$$\mathbf{x}_{k+1} = \mathbf{A}_k \mathbf{x}_k + \mathbf{q}_k, \quad \mathbf{q}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k), \quad (27)$$

where $\mathbf{A}_k = \mathbf{I}_N \otimes \mathbf{A}_k^i$ and $\mathbf{Q}_k = \mathbf{I}_N \otimes \mathbf{Q}_k^i$. The angular measurement model is defined as $\mathbf{y}_\alpha = \boldsymbol{\alpha}(\mathbf{p}_k^i) + \mathbf{r}_{\alpha,k}^i$, with

$$\boldsymbol{\alpha}(\mathbf{p}_k^i) = \begin{bmatrix} \arctan\left(\frac{(\mathbf{p}_k^i)^\top \mathbf{e}_3^3}{(\mathbf{p}_k^i)^\top \mathbf{e}_1^3}\right) \\ \arctan\left(\frac{\sqrt{((\mathbf{p}_k^i)^\top \mathbf{e}_1^3)^2 + ((\mathbf{p}_k^i)^\top \mathbf{e}_2^3)^2}}{(\mathbf{p}_k^i)^\top \mathbf{e}_3^3}\right) \end{bmatrix}, \quad (28)$$

and the noise $\mathbf{r}_{\alpha,k}^i \sim \mathcal{N}(\mathbf{0}, \sigma_\alpha^2 \mathbf{I}_2)$. Thus, by letting

$$\mathbf{p}_k = \begin{bmatrix} \mathbf{p}_k^1 \\ \vdots \\ \mathbf{p}_k^N \end{bmatrix}, \quad \mathbf{h}(\mathbf{p}_k) = \begin{bmatrix} \boldsymbol{\alpha}(\mathbf{p}_k^1) \\ \vdots \\ \boldsymbol{\alpha}(\mathbf{p}_k^N) \end{bmatrix}, \quad \mathbf{R}_{\alpha,k} = \begin{bmatrix} \mathbf{P}_k^{\hat{\mathbf{x}}^1} & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{P}_k^{\hat{\mathbf{x}}^N} \end{bmatrix}$$

and $\mathbf{R}_{\alpha,k} = \sigma_\alpha^2 \mathbf{I}_{2N}$, the combined measurement model of all agents' relative angles can be written

$$\mathbf{y}_k = \mathbf{H}(\mathbf{x}_k) + \mathbf{r}_k = \begin{bmatrix} \mathbf{h}(\mathbf{p}_k) \\ \mathbf{x}_k \end{bmatrix} + \begin{bmatrix} \mathbf{r}_{\alpha,k} \\ \mathbf{r}_{x,k} \end{bmatrix}, \quad (29)$$

where the transformation

$$\mathbf{T}_H = \begin{bmatrix} \mathbf{I}_N \otimes \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_{3 \times 6} \\ \mathbf{0}_{6 \times 3} & \mathbf{I}_6 \end{bmatrix} \end{bmatrix}, \quad (30)$$

transforms the state \mathbf{x}_k into the form in (7), where the non-linear states (here the positions) are stacked in the first $3N$ elements of the transformed state vector $\tilde{\mathbf{x}}_k = \mathbf{T}_H \mathbf{x}_k$. With the dynamics in (27) and the partially linear measurement equation in (29), we consider a problem where $N = 10$ (that is, $Z = 30$ and $L = 60$), and execute

- (A) a CKF, that is, Algorithm 1 with point sets \mathcal{P}_F and \mathcal{P}_H defined according to the SCR in Section IV-A.
- (B) a PL-CKF, that is, Algorithm 2 with point sets \mathcal{P}_F and \mathcal{P}_H defined according to the SCR in Section IV-A, using $\mathbf{T}_F = \mathbf{I}$ and \mathbf{T}_H defined in (30).

The resulting mean estimate error of the PL-CKF is shown in terms of the positions, velocities and accelerations of all agents together with the 95%-confidence interval in Figure 2. Furthermore, as we compute the same joint distribution in the PL-CKF and the CKF down to numerical precision, we get the same estimates with the two algorithms. This is shown by comparing the posterior estimate means, $\mathbf{m}_{k|k}^{\mathbf{x},A}$ for the CKF and $\mathbf{m}_{k|k}^{\mathbf{x},B}$ for the PL-CKF, here in l_2 -norm in time, as depicted in Figure 3. This demonstrates that Algorithm 2 indeed works as intended, and offers further numerical verification of the main result in Theorem 1.

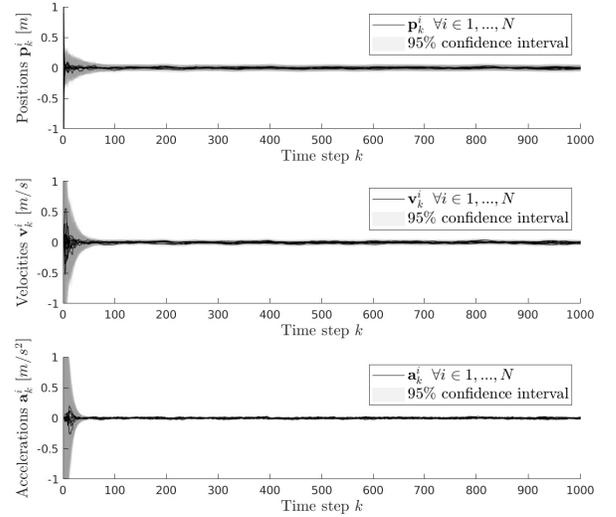


Fig. 2. Mean estimate error in positions (top), velocities (center) and accelerations (bottom), when running the PL-CKF for $N = 10$ agents.

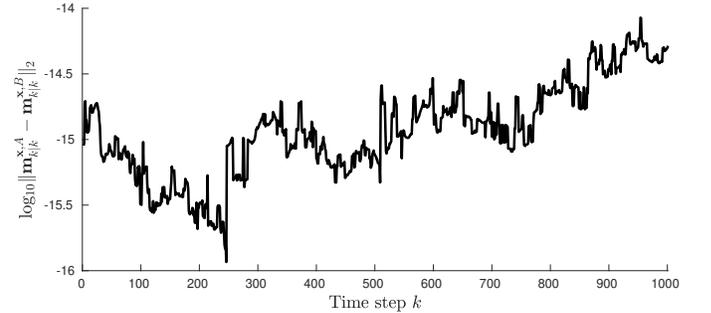


Fig. 3. Difference in the first moment of the state estimate distributions in the 10-logarithm and l_2 -norm when running the CKF and the PL-CKF on the same synthetic data with $N = 10$ agents.

VII. CONCLUSIONS

In this paper, we examine the implications of known linear substructure in the LRKFs, which can be leveraged to decrease the computational complexity of the filters significantly. We have stated exactly how the evaluation of the moment matching approximations in (13) simplifies when there exists a known linear substructure in Theorem 1, provided the cubature rule satisfies Assumptions 1-3. In addition, we have shown that we need not use the entire set of points \mathcal{P}_z in evaluating these integrals provided additional symmetry assumption, and that the number of function evaluations then

scales as $C(Z)$ instead of $C(X)$. We have given a condition under which only a partial Cholesky decomposition needs to be computed in the moment matching. Furthermore, we have demonstrated that the above assumptions are satisfied for the cubature rules used in the CKF, UKF, and GHKF. In doing so, we have also given modified versions of the original algorithms which exploit this linear substructure in Algorithm 2, and demonstrated its efficacy on a distributed filtering example. The results can easily be generalized to smoothing problems, but this falls outside the scope of the paper and is left as a topic for future research. On a final note, known linear substructure should be exploited in filtering applications whenever possible, and any implementation of an LRKF should be done using the moment approximations in Theorem 1 in order to conserve computational resources.

VIII. ACKNOWLEDGEMENT

The authors would like to thank the reviewers for helpful comments which helped to improve the quality of the paper.

IX. APPENDIX

Proof of Theorem 1. The proof follows from simple algebra invoking assumptions 1, 2, and 3. It can be found in its entirety in the extended version of the paper [18]. The main proof idea is to show that the first moment can be written

$$\mathbf{m}^y \approx \sum_{i=1}^{|\mathcal{P}|} w^{(i)} \mathcal{Y}^{(i)} = \begin{bmatrix} w_{cl} \mathbf{g}(\mathbf{m}^z) + \sum_{i=1}^{|\mathcal{P}_z|} w_z^{(i)} \mathbf{g}(\mathcal{Z}_z^{(i)}) \\ \mathbf{A} \mathbf{m}^x \end{bmatrix}, \quad (31)$$

using the assumption of symmetry, and consistency in the first moment. Inserting this in the LRKF-approximation of the moment integrals in (13d) and (13d), as well as utilizing the assumption of consistency in the second moments along with the definitions in Theorem 1 yields the desired result. \square

Proof of Corollary 1. This remark holds trivially in the case of the SC and UT, but is more complicated when $\mathbf{N}\Xi_z$ no longer contains unique columns. However, this can also be shown algebraically, done in its entirety in the extended version of the paper [18]. The key here is to let \mathcal{P}_z^u denote the set of points corresponding to the unique columns in $\mathbf{N}\Xi_z$, with the corresponding linear parts set to zero and the weights of all multiples of the the same point summed,

$$\mathcal{S} = \{\nu = \mathbf{N}\xi \mid (w, \xi) \in \mathcal{P}_z\}, \quad (32)$$

$$\mathcal{P}_z^u = \left\{ (w^u, \xi^u) \mid w^u = \sum_{(w, \xi) \in \mathcal{P}_z, \mathbf{N}\xi = \nu} w, \quad \xi^u = \begin{bmatrix} \nu \\ \mathbf{0}_{L \times 1} \end{bmatrix}, \nu \in \mathcal{S} \right\},$$

simple algebraic developments utilizing the assumption in Corollary 2 shows that the result of Theorem 1 holds when evaluated with \mathcal{P}_z^u instead of \mathcal{P}_z , concluding the proof. \square

Proof of Corollary 2. If $\bar{\mathbf{N}}\Xi_z = \mathbf{0}$,

$$\mathbf{L}^{xx} \Xi_z = \begin{bmatrix} \mathbf{L}^{zz} & \mathbf{0} \\ \mathbf{L}^{lz} & \mathbf{L}^{ll} \end{bmatrix} \begin{bmatrix} \mathbf{N} \\ \bar{\mathbf{N}} \end{bmatrix} \Xi_z = \begin{bmatrix} \mathbf{L}^{zz} \\ \mathbf{L}^{lz} \end{bmatrix} \mathbf{N}\Xi_z. \quad (33)$$

Proof of Corollary 3. Here, we can let the number of rows of \mathbf{A}_1 and \mathbf{A}_2 in (24) be N_1 and N_2 respectively. Let $\mathbf{A}^\top = [\mathbf{A}_1^\top, \mathbf{A}_2^\top]$ and define a function with output \mathbf{o} , and a map

$$\mathbf{o} = \begin{bmatrix} \mathbf{g}(z) \\ \mathbf{A} \mathbf{x} \end{bmatrix}, \quad \mathbf{M} = \begin{bmatrix} \mathbf{I}_{N_1} & \mathbf{I}_{N_1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{N_2} \end{bmatrix}. \quad (34)$$

such that with the structure of the function defining \mathbf{y} in (24), we have that $\mathbf{y} = \mathbf{M}\mathbf{o}$. Then, the joint distribution $p(\mathbf{x}, \mathbf{o})$ can simply be computed using Theorem 1 to find $\mathbf{m}^{\mathbf{o}}, \mathbf{P}^{\mathbf{x}\mathbf{o}}, \mathbf{P}^{\mathbf{o}\mathbf{o}}$, followed by computation of the joint distribution $p(\mathbf{x}, \mathbf{y})$ in terms of $\mathbf{m}^{\mathbf{y}}, \mathbf{P}^{\mathbf{x}\mathbf{y}}, \mathbf{P}^{\mathbf{y}\mathbf{y}}$ where then

$$\mathbf{m}^{\mathbf{y}} = \mathbf{M} \mathbf{m}^{\mathbf{o}}, \mathbf{P}^{\mathbf{x}\mathbf{y}} = \mathbf{P}^{\mathbf{x}\mathbf{o}} \mathbf{M}^\top, \mathbf{P}^{\mathbf{y}\mathbf{y}} = \mathbf{M} \mathbf{P}^{\mathbf{o}\mathbf{o}} \mathbf{M}^\top. \quad (35)$$

\square

REFERENCES

- [1] J. Steinbring and U. D. Hanebeck, "LRKF revisited: The smart sampling Kalman filter (S2KF)," *Journal of Advances in Information Fusion*, vol. 9, no. 2, pp. 106–123, 2014.
- [2] G. Kurz and U. D. Hanebeck, "Linear regression Kalman filtering based on hyperspherical deterministic sampling," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 2017, pp. 977–983.
- [3] N. J. Gordon, D. J. Salmond, and A. F. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," in *IEE proceedings (radar and signal processing)*, vol. 140, no. 2, 1993, pp. 107–113.
- [4] A. Smith, *Sequential Monte Carlo methods in practice*. Springer Science & Business Media, 2013.
- [5] T. Schon, F. Gustafsson, and P.-J. Nordlund, "Marginalized particle filters for mixed linear/nonlinear state-space models," *IEEE Transactions on signal processing*, vol. 53, no. 7, pp. 2279–2289, 2005.
- [6] T. B. Schon, R. Karlsson, and F. Gustafsson, "The marginalized particle filter in practice," in *2006 IEEE Aerospace Conference*. IEEE, 2006, pp. 11–pp.
- [7] S. Särkkä, *Bayesian filtering and smoothing*. Cambridge University Press, 2013, vol. 3.
- [8] I. Arasaratnam, "Cubature Kalman filtering theory & applications," Ph.D. dissertation, McMaster University, 2009.
- [9] B. Jia, M. Xin, and Y. Cheng, "High-degree cubature Kalman filter," *Automatica*, vol. 49, no. 2, pp. 510–518, 2013.
- [10] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in *Signal processing, sensor fusion, and target recognition VI*, vol. 3068. International Society for Optics and Photonics, 1997, pp. 182–194.
- [11] E. A. Wan, R. Van Der Merwe, and S. Haykin, "The unscented Kalman filter," *Kalman filtering and neural networks*, vol. 5, no. 2007, pp. 221–280, 2001.
- [12] J. Dunik, O. Straka, and M. Šimandl, "The development of a randomised unscented Kalman filter," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 8–13, 2011.
- [13] —, "Stochastic integration filter," *IEEE Transactions on Automatic Control*, vol. 58, no. 6, pp. 1561–1566, 2013.
- [14] M. R. Morelande and B. Moran, "An unscented transformation for conditionally linear models," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP*, vol. 3. IEEE, 2007, pp. III–1417.
- [15] F. Beutler, M. F. Huber, and U. D. Hanebeck, "Gaussian filtering using state decomposition methods," in *2009 12th International Conference on Information Fusion*. IEEE, 2009, pp. 579–586.
- [16] K. Ito, "Gaussian filter for nonlinear filtering problems," in *Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No. 00CH37187)*, vol. 2. IEEE, 2000, pp. 1218–1223.
- [17] G. H. Golub and J. H. Welsch, "Calculation of Gauss quadrature rules," *Mathematics of computation*, vol. 23, no. 106, pp. 221–230, 1969.
- [18] M. Greiff, A. Robertsson, and K. Berntorp, "Exploiting linear substructure in linear regression kalman filters (extended version)," *arXiv preprint arXiv:1809.05509*, 2020.
- [19] R. A. Singer, "Estimating optimal tracking filter performance for manned maneuvering targets," *IEEE Transactions on Aerospace and Electronic Systems*, no. 4, pp. 473–483, 1970.