

MSE-optimal measurement dimension reduction in Gaussian filtering*

Greiff, Marcus; Robertsson, Anders; Berntorp, Karl

TR2020-124 August 28, 2020

Abstract

We present a framework for measurement dimension reduction in Gaussian filtering, defined in terms of a linear operator acting on the measurement vector. This operator is optimized to minimize the Cramer–Rao bound of the estimate’s mean squared error (MSE), yielding a measurement subspace from which elements minimally worsen the filter MSE performance, as compared to filtering with the original measurements. This is demonstrated with Kalman filtering in a linear Gaussian setting and various non-linear Gaussian filters with an on-line adaption of the operator. The proposed method improves computational time in exchange for a quantifiable and sometimes negligibly worsened MSE of the estimate

Conference on Control Technology and Applications (CCTA)

© 2020 MERL. This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of Mitsubishi Electric Research Laboratories, Inc.; an acknowledgment of the authors and individual contributions to the work; and all applicable portions of the copyright notice. Copying, reproduction, or republishing for any other purpose shall require a license with payment of fee to Mitsubishi Electric Research Laboratories, Inc. All rights reserved.

MSE-optimal measurement dimension reduction in Gaussian filtering*

Marcus Greiff¹, Anders Robertsson¹ and Karl Berntorp²

Abstract—We present a framework for measurement dimension reduction in Gaussian filtering, defined in terms of a linear operator acting on the measurement vector. This operator is optimized to minimize the Cramér–Rao bound of the estimate’s mean squared error (MSE), yielding a measurement subspace from which elements minimally worsen the filter MSE performance, as compared to filtering with the original measurements. This is demonstrated with Kalman filtering in a linear Gaussian setting and various non-linear Gaussian filters with an on-line adaption of the operator. The proposed method improves computational time in exchange for a quantifiable and sometimes negligibly worsened MSE of the estimate.

I. INTRODUCTION

In numerous filtering applications, there is a relative abundance of measurement information and a need to achieve a high estimation accuracy, often quantified in terms of an empirically measured mean-squared error (MSE) of the estimate. One such example is in global navigation satellite system (GNSS) positioning, where measurements on multiple carrier-frequency bands from multiple satellites in multiple constellations yield information on the state of the receiver [1], [2]. In standard GNSS implementations, it is not uncommon to be dealing with hundreds of measurements per time-step. This results in problems related to computational complexity, often solved sub-optimally by heuristically removing measurements to facilitate real-time filter execution. Applications that suffer from similar problems include RSSI-localization [3] and Gaussian Process (GP) learning [4].

Motivated by such high-dimensional estimation applications, we propose a systematic approach for combining the acquired measurements in a lower-dimensional vector space, so as to minimally degrade estimate performance when using minimum MSE (MMSE) estimators. For generic estimation problems, the Cramér–Rao Bound (CRB) serves as a lower bound on the variance of any unbiased state-estimate, cf. [5] and [6]. Consequently, we explore filtering with linear combinations of the acquired measurements and study its effect on the CRB. We then minimize the CRB of the posterior state-estimate MSE over the set of linear operators used to project the measurements. When applied to filtering problems to reduce measurement dimensionality, it

yields a *quantifiable* and *controllable* performance degradation, which is often negligible, in exchange for a substantial computational speedup. Hence, the proposed approach serves as a valuable tool in real-time filter implementations. The ideas are tailored for Gaussian filters, but have further applicability to Rao-Blackwellized Particle filtering (RBPF) approaches in [7] or the Kalman mixture model filtering [8].

1) *Contributions*: We present a way of computing a linear combination of the measurements, such that the CRB of the state-estimate MSE is minimized in filtering with the lower-dimensional projected measurements. We provide an efficient way of evaluating the gradient of a objective function in the CRB, facilitating a wide range of optimization algorithms.

In the case of linear Kalman filtering (KF), the CRB can be computed recursively and is tight. Here we give an example with an offline version of the optimization, providing a quantifiable performance decrease for a significantly reduced complexity in the online measurement updates.

We then consider a wider class of nonlinear filters, here restricted to the Gaussian Approximate density filters (ADFs), and extend the ideas by providing an algorithm for the on-line adaption of the projection operator in this setting.

2) *Problem*: We consider discrete-time systems,

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{w}_k \in \mathbb{R}^N, \quad (1a)$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{e}_k \in \mathbb{R}^M, \quad (1b)$$

where the process noise and measurement noise are Gaussian distributed with zero mean, respectively, with $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$, $\mathbf{e}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$, 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 find a linear map $\Psi_k : \mathbb{R}^M \rightarrow \mathbb{R}^{\bar{M}}$ such that filtering with the measurements $\tilde{\mathbf{y}}_k = \Psi_k(\mathbf{y}_k)$ minimally degrades estimate MSE performance for fixed \bar{M} .

3) *Notation*: We let uppercase and bold font variables \mathbf{Y} denote matrices and refer to an element of this matrix by $[\mathbf{Y}]_{ij}$. Similarly, vectors are denoted with lower case bold font \mathbf{y} with elements y_i . All signals are considered in discrete time with a sampling period of $h > 0$ [s] where, $\mathbf{y}_k = \mathbf{y}(hk)$, and we denote $\mathbf{y}_{0:k} = \{\mathbf{y}_0, \dots, \mathbf{y}_k\}$. We will be dealing with Gaussian distributions, here denoted $\mathcal{N}(\mathbf{x}_k; \mathbf{m}_k^x, \Sigma_k^{xx})$, where the mean is denoted by \mathbf{m}_k^x and covariance by Σ_k^{xx} . Furthermore, we take the sub-indexation $a|b$ to denote conditioning of a variable at a time-step $k = a$ on information up until and including $k = b$.

4) *Outline*: The paper gives a brief overview of the nonlinear Gaussian filters in Section II and summarizes relevant results on the CRB in Section III, before introducing the projection optimization scheme in Section IV. The claims are verified by examples in Section V, and the conclusions in Section VI closes the paper, with proofs in the Appendix.

*Marcus Greiff performed parts of this work as an intern at MERL.

¹The authors are members of the LCCC Linnaeus Center and the ELLIIT Excellence Center at Lund University, and affiliated with Swedish Science Foundation (SSF) project "Semantic mapping and visual navigation for smart robots" (RIT15-0038). Department of Automatic Control, Lund University, SE-221 00 Lund, Sweden. {marcus.greiff, andersro}@control.lth.se

²The author is with Mitsubishi Electric Research Laboratories (MERL), 02139 Cambridge, MA, USA. berntorp@merl.com

II. GAUSSIAN APPROXIMATE DENSITY FILTERS

The general idea of the ADF is to approximate the state-distribution with some predefined distribution, often chosen for computational tractability. This is particularly the case in the Gaussian filters, where the posterior distribution of the state-estimate at time step $k-1$ is approximated as

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

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

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

and the joint distribution is approximated based on (1b), as

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

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

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

where,

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

The difference among all of the Gaussian ADFs lie in the way the distribution in (3) and the joint distribution in (6) are approximated. In the event of linear flow and measurement equations (1), the state-distribution will be Gaussian at all times, and the conditional can be computed exactly. The above equations in (6) then result in the familiar KF measurement update, which is the optimal MMSE estimator for the filtering problem in the linear setting. If the flow or measurement functions are non-linear, a first order Taylor expansion about the estimate mean results in the Extended Kalman filter (EKF). This filter comes in many versions, with iteratively refined approximations in the (IEKF), and higher order approximations in the second-order EKF (SOEKF) [9]. Another option is to approximate the joint distribution in (6) using the unscented transform (UT) resulting in the Unscented Kalman filter (UKF) [10], or by other direct approximations of the moment integrals

$$\begin{aligned} \mathbf{m}^y &= \int \mathbf{h}(\mathbf{x}) \mathcal{N}(\mathbf{x}; \mathbf{m}^x, \Sigma^{xx}) d\mathbf{x}, \quad (7) \\ \Sigma^{yy} &= \int (\mathbf{m}^y - \mathbf{h}(\mathbf{x})) (\mathbf{m}^y - \mathbf{h}(\mathbf{x}))^\top \mathcal{N}(\mathbf{x}; \mathbf{m}^x, \Sigma^{xx}) d\mathbf{x}, \\ \Sigma^{xy} &= \int (\mathbf{m}^x - \mathbf{x}) (\mathbf{m}^y - \mathbf{h}(\mathbf{x}))^\top \mathcal{N}(\mathbf{x}; \mathbf{m}^x, \Sigma^{xx}) d\mathbf{x}, \end{aligned}$$

as done in the large family of the linear regression Kalman filters (LRKFs) [11]. This includes various Gaussian cubature rules, of which the more common are the spherical-radial cubature rules, resulting in the Cubature Kalman filter (CKF) [12], the Gauss-Hermite cubature rules resulting in the (GHKF) [13], and the less common MC-based approach in the randomized Unscented Kalman filter (RUKF) [14].

In all of these Gaussian ADFs, we need to invert $\Sigma_{k|k-1}^{yy}$ to compute the conditional distribution in (6), which may

become computationally cumbersome for high-dimensional measurement vectors \mathbf{y}_k . This is especially true if the filters are to be used in RBPF schemes or in other mixture model KF approaches, where multiple filters are executed in parallel. Consequently, we attempt to address the problem of reducing the complexity of the measurement update in (6) by considering the Fisher information matrix (FIM) in the context of the measurement equation in (1b). This approach applies to all of the aforementioned Gaussian ADFs.

III. REVIEW OF THE CRAMÉR-RAO BOUND

The CRB serves as a valuable tool when evaluating filter performance, as it provides a lower bound on the variance of any unbiased estimate through the inverse of the Fisher Information Matrix (FIM), here denoted by \mathcal{I} . The CRB is derived for the samples from multi-variate Gaussian distributions in [5], for more general nonlinear Gaussian filtering problems in [15], with an excellent contemporary survey paper on various forms of Bayesian CRBs in [6].

In this paper, we will primarily consider the CRB of the MSE of the filter estimate, as given in Remark 1.

Remark 1: For any estimate $\hat{\mathbf{x}}_k$ of \mathbf{x}_k , we have

$$\mathbb{E}[\|\hat{\mathbf{x}}_k - \mathbf{x}_k\|_2^2] = \text{Tr}(\text{Cov}[\hat{\mathbf{x}}_k]) \geq \text{Tr}(\mathcal{I}[\mathbf{x}_k]^{-1}). \quad (8)$$

To evaluate this bound, we need to compute FIM in the context of the filtering problem at hand, which we can do in two different ways. The first only considers the measurement at a single time-step with no prior information, here referred to as the memoryless FIM. This essentially treats the measurement in an least-squares setting, where we have no process noise or state dynamics and no prior information at the time of computing the FIM. From [5],

$$\mathcal{I}[\mathbf{x}_k] = \mathbf{H}_k^\top \mathbf{R}_k^{-1} \mathbf{H}_k, \quad \mathbf{H}_k = \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}}. \quad (9)$$

However, in the context of (1), we some dynamics and process noise which will both affect the CRB. Therefore, the FIM could alternatively be computed recursively [6]. If the flow and measurement functions are linear in \mathbf{x} , then

$$\mathbf{F}_k = \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}}, \quad \mathbf{H}_k = \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}}, \quad (10)$$

are constant, the FIM at a time-step k takes the form

$$\mathcal{I}[\mathbf{x}_k] = [\mathbf{Q}_k + \mathbf{F}_k \mathcal{I}[\mathbf{x}_{k-1}]^{-1} \mathbf{F}_k]^{-1} + \mathbf{H}_k^\top \mathbf{R}_k^{-1} \mathbf{H}_k, \quad (11)$$

as shown in [6], [15]. The associated CRB is commonly referred to as the joint unconditional Bayesian CRB (JU-BCRB), and the celebrated KF yields a tight CRB in this case, being the MMSE estimator for this particular class of problems. In the general non-linear context, there exist many CRBs of various tightness, and these will depend on and have to be computed with respect to a specific state-trajectory. For simplicity, we will use a first-order Taylor expansion about the state-estimate mean in evaluating the JU-BCRB in the nonlinear setting, referring to the prior information associated with \mathbf{x}_{k-1} by $\mathcal{P}_k := [\mathbf{Q}_k + \mathbf{F}_k \mathcal{I}[\mathbf{x}_{k-1}]^{-1} \mathbf{F}_k]^{-1}$, such that the FIM in (11) can be written compactly as

$$\mathcal{I}[\mathbf{x}_k] = \mathcal{P}_k + \mathbf{H}_k^\top \mathbf{R}_k^{-1} \mathbf{H}_k. \quad (12)$$

Knowing how to compute the MSE CRB in (8), and knowing how to recursively evaluate the FIM in (11), we can study how certain operations on the measurement affects the theoretical filter MSE-performance. This allows us to address the measurement reduction problem posed in Section I.

IV. COMPUTING MSE-OPTIMAL PROJECTIONS

The general idea is to consider the effect on the MSE CRB when filtering with a lower-dimensional vector $\tilde{\mathbf{y}}_k = \Psi_k(\mathbf{y}_k) \in \mathbb{R}^{\tilde{M}}$ for some $\tilde{M} \leq M$. Now, if we constrain this map to be linear, with $\Psi_k \in \mathbb{R}^{\tilde{M} \times M}$, then

$$\tilde{\mathbf{y}}_k = \Psi_k \mathbf{y}_k \sim \mathcal{N}(\Psi_k \mathbf{h}(\mathbf{x}_k), \Psi_k^\top \mathbf{R}_k \Psi_k). \quad (13)$$

Remark 2: A linear map is mainly chosen for computational tractability and will be shown to be a sufficiently large class of maps in the considered numerical examples.

Starting with the memory-less FIM in (9), computed with respect to the state vector with no prior information, provided the distribution of the projected measurements in (13), is

$$\mathcal{I}[\mathbf{x}_k; \Psi_k] = \mathbf{H}_k^\top \Psi_k^\top (\Psi_k^\top \mathbf{R}_k \Psi_k)^{-1} \Psi_k \mathbf{H}_k. \quad (14)$$

As for the FIM in (11), incorporating prior information from the filtering history through \mathcal{P}_k , we instead get

$$\mathcal{I}[\mathbf{x}_k; \Psi_k] = \mathcal{P}_k + \mathbf{H}_k^\top \Psi_k^\top (\Psi_k^\top \mathbf{R}_k \Psi_k)^{-1} \Psi_k \mathbf{H}_k. \quad (15)$$

Remark 3: Equation (15) can be shown by arguments similar to those in [6]. Intuitively, as $\mathcal{I}[\mathbf{x}_{k-1}]$, \mathbf{f} and \mathbf{Q}_k are unaffected by Ψ_k , \mathcal{P}_k will in no way be affected by a projection Ψ_k acting on \mathbf{y}_k . The FIM will be a sum of this prior information, and the new information from the projected measurement, here the same as in (14).

From (15), we can express the CRB of the MSE as a function of the projection operator using (8) and (14), as

$$J(\Psi_k) = \text{Tr}((\mathcal{P}_k + \mathbf{H}_k^\top \Psi_k^\top (\Psi_k^\top \mathbf{R}_k \Psi_k)^{-1} \Psi_k \mathbf{H}_k)^{-1}) \quad (16)$$

and study the non-convex optimization problem

$$\min_{\Psi_k \in \mathbb{R}^{\tilde{M} \times M}} J(\Psi_k). \quad (17)$$

Remark 4: The function $J(\Psi)$ is scale invariant in the memory-less case (if we let $\mathcal{P}_k = \mathbf{0}$), in the sense that $J(\alpha\Psi) = J(\beta\Psi)$ for all $\beta \neq 0$, $\alpha \neq 0$. In this case, the linear operator can be normalized to keep the magnitude of the projected measurements constant.

Remark 5: The function $J(\Psi)$ will never be smaller than the identity projection $J(\mathbf{I})$, as the linear combination of measurements cannot contribute any new information.

The objective function also has some beneficial structure that can be exploited to compute its gradient with respect to the projection operator analytically, thereby facilitating a wide range of efficient gradient-based optimization methods.

Proposition 1: The partial derivative of the JU-BCRB trace when using the projected measurements $\tilde{\mathbf{y}} = \Psi \mathbf{y}$, taken with respect to the projection operator Ψ , is given by

$$\frac{\partial \text{Tr}((\mathcal{I}[\mathbf{x}; \Psi])^{-1})}{\partial \Psi} = -2\mathbf{U}\mathbf{L}\mathbf{L}^{-2}\mathbf{L}^\top \mathbf{V}^\top \quad (18)$$

where

$$\mathbf{Y} = \mathbf{H}^\top \Psi^\top (\Psi \mathbf{R} \Psi^\top)^{-1} \Psi \mathbf{H}, \quad (19a)$$

$$\mathbf{U} = (\Psi \mathbf{R} \Psi^\top)^{-1} \Psi \mathbf{H}, \quad (19b)$$

$$\mathbf{V} = \mathbf{H} - \mathbf{R} \Psi^\top \mathbf{U}, \quad (19c)$$

and $\mathbf{L}\mathbf{L}^\top$ is the Schur-decomposition of $\mathbf{Y} + \mathcal{P}_k$.

Proof: The proof is summarized in the Appendix. ■

Proposition 1 is quite appealing, as we can construct various gradient-based optimization schemes to solve (17), the simplest being a gradient-descent. Denoting the projection operator at iteration n and time k by Ψ_k^n , we can iterate

$$\Psi_k^{n+1} = \Psi_k^n - \gamma \frac{\partial \text{Tr}((\mathcal{I}[\mathbf{x}; \Psi_k^n])^{-1})}{\partial \Psi}, \quad (20)$$

with $\gamma > 0$ is chosen by a backtracking line-search [16].

Remark 6: We cannot give any global optimality guarantees for any optimization scheme since the problem is non-convex, but it is easy to check the quality of a locally optimal solution. If the ratio $J(\Psi)/J(\mathbf{I}) \rightarrow 1$, then we can expect the same MSE CRB when using the projected measurements as when using the original measurements.

A. Implications of the filtering problem

If using the projected measurements in the filtering context, we get a joint Gaussian for the measurement update

$$\begin{bmatrix} \hat{\mathbf{x}}_{k|k-1} \\ \tilde{\mathbf{y}}_{k|k-1} \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} \mathbf{m}_{k|k-1}^x \\ \Psi \mathbf{m}_{k|k-1}^y \end{bmatrix}, \begin{bmatrix} \Sigma_{k|k-1}^{xx} & \Sigma_{k|k-1}^{xy} \Psi_k^\top \\ \Psi_k \Sigma_{k|k-1}^{yx} & \Psi_k \Sigma_{k|k-1}^{yy} \Psi_k^\top \end{bmatrix} \right),$$

and given a measurement $\tilde{\mathbf{y}}_k = \Psi_k \mathbf{y}_k$, we approximate

$$p(\hat{\mathbf{x}}_k | \tilde{\mathbf{y}}_{0:k}) \approx \mathcal{N}(\mathbf{m}_{k|k}^x, \Sigma_{k|k}^{xx}), \quad (21)$$

where the conditional distribution is given by

$$\mathbf{K}_k = \Sigma_{k|k-1}^{xy} \Psi_k^\top (\Psi_k \Sigma_{k|k-1}^{yy} \Psi_k^\top)^{-1}, \quad (22a)$$

$$\mathbf{m}_{k|k}^x = \mathbf{m}_{k|k-1}^x + \mathbf{K}_k (\tilde{\mathbf{y}}_k - \Psi_k \mathbf{m}_{k|k-1}^y), \quad (22b)$$

$$\Sigma_{k|k}^{xx} = \Sigma_{k|k-1}^{xx} - \mathbf{K}_k \Psi_k \Sigma_{k|k-1}^{yx}. \quad (22c)$$

Here we see that all of the means and covariances in the original filter formulation in (6) re-appear in (22), meaning that we can use any of the approximations discussed in Section II, and use the projection with any of the outlined filters therein. Furthermore, as $\Psi_k \Sigma_{k|k-1}^{yy} \Psi_k^\top \in \mathbb{R}^{\tilde{M} \times \tilde{M}}$ and $\Sigma_{k|k-1}^{xy} \Psi_k^\top = (\Psi_k \Sigma_{k|k-1}^{yx})^\top \in \mathbb{R}^{M \times \tilde{M}}$, the matrices we effectively use in the filter update can potentially be made significantly smaller, if we for some $\tilde{M} \ll M$ can find projections for which $J(\Psi)/J(\mathbf{I}) \rightarrow 1$. However, when considering on-line adaption of the projection operator, the overall numerical complexity of the projection scheme will depend on the convergence rate of the optimization program, which in turn depends on properties of the flow and measurement functions of the underlying model. It is interesting to note that in the linear case if we know the stationary covariance \mathcal{P}_k for large $k \rightarrow \infty$, the operator Ψ_k can be optimized offline before executing the filter.

V. NUMERICAL EXAMPLES

To demonstrate the utility of the projection scheme and verify its correctness, we give four separate examples. In Section V-A, the expression for the gradient of the CRB trace in Proposition 1 is verified, giving some intuition regarding the rather arcane optimization program. We then apply this optimal projection computation to the problem of linear Kalman filtering in Section V-B, where it is used to reduce the dimension of the measurement vector at a negligible cost in the empirically computed estimate MSE. In Section V-C, we proceed to show how the application of this scheme improves the numerical complexity of the generic Gaussian ADF update as a function of the dimension of the measurement vector. Finally, in Section V-D we study the impact of the measurement projection scheme on the estimate MSE in the context of nonlinear Gaussian filtering.

A. Illustration of the optimization scheme

To give some intuition regarding the objective function used in the numerical optimization, we randomize a dense matrix $\mathbf{H}_k \in \mathbb{R}^{5 \times 3}$, a positive definite covariance matrix $\mathbf{R}_k = \mathbf{R}_k^\top \in \mathbb{R}^{5 \times 5}$, and a random measurement projection $\Psi_k \in \mathbb{R}^{3 \times 5}$. The level sets of the scalar objective function $J(\Psi_k)$ are plotted when varying two of the elements of Ψ_k (all others are kept constant), and the gradient directions are computed using Proposition 1 with the result shown in Fig. 1.

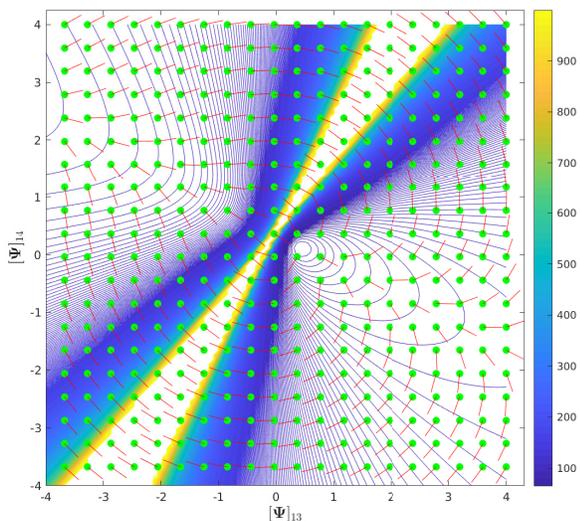


Fig. 1. Level sets of the objective $J(\Psi)$ and its gradient (here normalized).

It is clear that the gradient direction is orthogonal to the level sets, and we also note a substantial variability in the objective function over the plotted domain. It is intuitively obvious that we can make the estimate CRB MSE arbitrarily large by setting $\Psi_k = \mathbf{0}$, but it is less intuitive that the quality of the estimate is so contingent of individual elements of the linear operator. Recall that we are only varying two dimensions of a 5×3 -dimensional linear operator in Fig. 1.

B. Offline projection computation in linear Kalman filtering

In this example, we consider a linear time-invariant (LTI) system defined by the three-dimensional triple integrator, $\ddot{\mathbf{p}} = \ddot{\mathbf{v}} = \dot{\mathbf{a}} = \mathbf{u} + d\beta \in \mathbb{R}^3$, with a state vector $\mathbf{x} = [\mathbf{p}^\top, \mathbf{v}^\top, \mathbf{a}^\top]^\top$ discretized at a time-step of $h = 0.01$ [s] using zero-order hold and driven by an input $\mathbf{u} = 5[\cos(4ht), \cos(5ht), \cos(2ht)]^\top$ and Wiener distribution $d\beta$. The measurement equation is characterized by a set of N points $\{\mathbf{p}_i \in \mathbb{R}^3; i = 1, \dots, N\}$ to which distances $d_i = \|\mathbf{p}_i - \mathbf{p}\|_2$ are measured. We then form distance-difference measurements, known as time-difference of arrival measurements in the Ultra Wideband (UWB) literature or single difference measurements in the GNSS literature. The m^{th} element of $\mathbf{h}(\mathbf{x})$ is defined by $h_m(\mathbf{x}) = d_i(\mathbf{x}) - d_j(\mathbf{x})$, including all combinations (i, j) with $i > j$.

To properly illustrate the ideas in Section IV, we start by considering an LTI context where the CRB should be tight. Using a slightly simplified measurement model, where

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{x}_k \quad \text{where} \quad \mathbf{H}_k = \left. \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}_0} \quad \forall k. \quad (23)$$

We let \mathbf{R}_k be a random dense matrix satisfying $0.1\mathbf{I} \prec \mathbf{R}_k \prec 0.3\mathbf{I}$ and use a process noise defined by $\mathbf{Q}_k = 0.01\mathbf{I}$, and randomize a set of points $N = 6$ points, with $\dim(\mathbf{y}_k) = N(N-1)/2 = 15$ measurements per time-step.

Given this time-invariant measurement model, we compute a projection operator offline which is to be applied when solving the filtering problem. For this purpose, we use the memory-less FIM in the optimization problem, and compare three different projection operators, as listed below;

- (A) Here we use all available measurements, letting $\Psi_A = \mathbf{I} \in \mathbb{R}^{\tilde{M} \times \tilde{M}}$, where then $(\tilde{M}, M) = (15, 15)$.
- (B) Here we use the first five measurements, letting $\Psi_B = [\mathbf{I}, \mathbf{0}, \mathbf{0}] \in \mathbb{R}^{\tilde{M} \times \tilde{M}}$, where then $(\tilde{M}, M) = (5, 15)$.
- (C) Here we choose $\Psi_C \in \mathbb{R}^{\tilde{M} \times \tilde{M}}$ solving (17) by iterating (20), here constrained to $(\tilde{M}, M) = (3, 15)$.

Having solved the optimization program by iterating (20), we get the following cost ratios from the three different maps;

$$(A) \quad J(\Psi_A)/J(\mathbf{I}) = 1.000, \quad (24a)$$

$$(B) \quad J(\Psi_B)/J(\mathbf{I}) = 2.437, \quad (24b)$$

$$(C) \quad J(\Psi_C)/J(\mathbf{I}) = 1.001. \quad (24c)$$

Consequently, if the filtering problem using an unbiased MMSE estimator such as the KF, then we should expect near identical performance when using the original filter (A) using Ψ_A as when using the projected measurements in (C) using Ψ_C , and the strategy in (B) using Ψ_B can be expected to perform roughly a factor 2.5 worse in terms of MSE.

To verify these claims, we give a simulation study where a standard discrete-time KF is executed with the projected measurements, using each of the maps Ψ_A , Ψ_B , and Ψ_C . We perform $N_{MC} = 10^3$ Monte-Carlo (MC) runs and denote a state trajectory of simulation n by \mathbf{x}_k^n . In Fig. 2, the empirically computed MSE of the estimate is compared to

an average posterior estimate covariance (APC),

$$MSE(\mathbf{x}_k) = \frac{1}{N_{MC}} \sum_{n=1}^{N_{MC}} \|\mathbf{x}_k^n - \hat{\mathbf{x}}_{k|k}^n\|_2^2,$$

$$APC(\Sigma_k^{\mathbf{x}\mathbf{x}}) = \frac{1}{N_{MC}} \sum_{n=1}^{N_{MC}} \Sigma_{k|k}^{\mathbf{x}\mathbf{x},n},$$

the latter included to check consistency of the state-estimates.

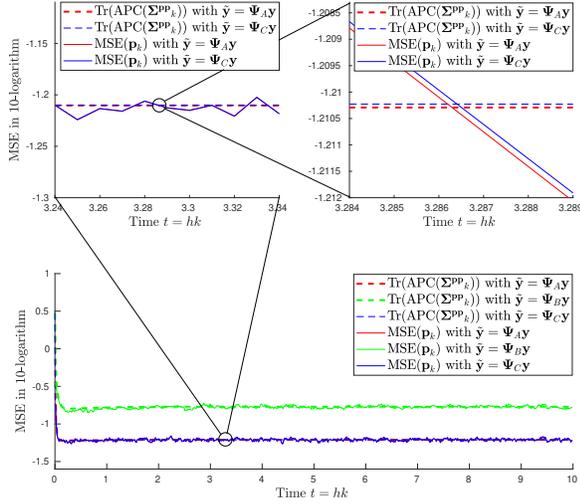


Fig. 2. Empirically computed positional MSE plotted against the trace of the positional APC from the MC-executions in the example in Section V-B. *Bottom*: APC trace and MSE of the positional states, \mathbf{p} , using the operators in (A)-(C). *Top, left*: Enlarged view of the MSE corresponding to the cases (A) and (C) around 3.3 [s]. *Top, right*: Further enlarged view of the MSE corresponding to the cases (A) and (C) around 3.3 [s].

Here we start by noting that the filters are consistent, i.e., the MSE is approximately equal to the APC-trace for the three cases (A)-(C). Furthermore, we see that the MSE-performance in (A) and (C) are virtually the same, given that the computed CRBs $J(\Psi)$ in (24) are near-identical for Ψ_A and Ψ_C . However, simply choosing the first five measurements with Ψ_B results in a loss of information and a degraded MSE performance (see Fig 2). Here, the ratio of the MSE when using (B) as compared to (A) is $10^{-0.8}/10^{-1.2} \approx 2.5$, approximately equal to the ratio in (24). Similarly, the ratio between the MSE when using (C) as compared to (A) is $10^{-1.21}/10^{-1.2103} \approx 1.001$, again comparable to (24).

Using Ψ_C clearly results in a negligible performance decrease when compared to using the full measurement vector in Ψ_A . However, instead of performing the Kalman filter updates by inverting $\Sigma_{k|k-1}^{\mathbf{y}\mathbf{y}} \in \mathbb{R}^{15 \times 15}$, we now require an inversion of the projected measurement covariance,

$$\Sigma_{k|k-1}^{\tilde{\mathbf{y}}\tilde{\mathbf{y}}} = \Psi_C(\mathbf{H}_k \Sigma_{k|k-1}^{\mathbf{x}\mathbf{x}} \tilde{\mathbf{H}}_k + \mathbf{R}_k) \Psi_C^T \in \mathbb{R}^{3 \times 3},$$

which results in a computational speedup when running the filter with the projection operator Ψ_C as opposed to using the original measurements with Ψ_A .

This example demonstrates the utility of the optimization scheme in a linear setting. Importantly, it shows that it is

possible to perform an offline optimization of Ψ prior to the filtering, in this case resulting in a computational speed-up of the Kalman filter at a negligible decrease in the MSE of the estimate. Exactly how much faster the algorithm becomes depends greatly on M and \tilde{M} , as we shall see next.

C. Computational complexity

We now consider the impact of the measurement projections on computational complexity for problems similar to the three-dimensional triple integrator in the previous example. A large contributor to the numerical complexity in the measurement update of the ADF comes from the need to invert of the predicted measurement covariance in (6), scaling roughly with $O(M^3)$. When applying the linear operator to the measurement as in (22), we need to perform a few additional matrix multiplications in order to compute the joint distribution of the predicted state and predicted projected measurement. However, the complexity of the inversion will instead be capped at $O(\tilde{M}^3)$ with \tilde{M} being an integer that we are free to choose. In this example, we measure the computational time required to perform a measurement update with M increasing from 10 to 100, using (6) and (22) respectively. Here, $\tilde{M} = 3$ as in the previous example, and the mean computational time over 10^7 updates are shown in Fig 3.

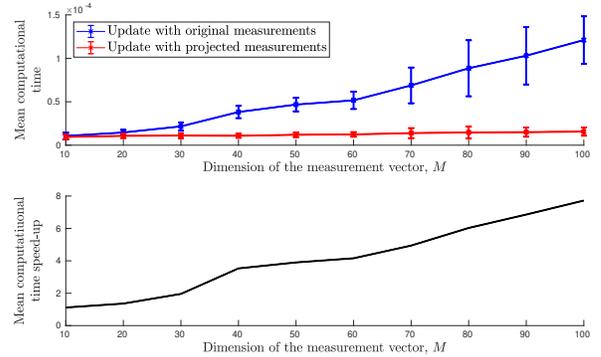


Fig. 3. *Top*: Mean computational time taken to perform a measurement with a 1σ -interval from 10^7 updates as a function of the measurement dimension M . Without the measurement projection using (6) (blue) and with the measurement projection (22) (red). *Bottom*: Relative speedup in the mean computational time when using the measurement projection.

Based on this, we note that the proposed measurement reduction greatly reduces the computational complexity of the measurement update, and in the previous example, we should expect an 8-fold computational speedup of the measurement update for a quantifiable MSE performance decrease when using upwards of $M = 100$ measurements. However, this is for the offline computation of Ψ . In the non-linear setting, additional computation will be required to adapt Ψ_k .

D. Online projection adaption in nonlinear filtering

To demonstrate the online adaption of the projection operator, we consider a simple example with a two-dimensional

non-linear pendulum. The process is defined by a two-dimensional state-space and discretized at $h = 0.01$, where

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{w}_k = \begin{bmatrix} x_{1,k} + hx_{2,k} \\ x_{2,k} + hg \sin(x_{1,k}) \end{bmatrix} + \mathbf{w}_k.$$

Here we simply let $g = 9.81$ and assume zero-mean Gaussian noise $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$ where $\mathbf{Q} = \text{diag}(10^{-5}, 10^{-1})$. However, in contrast to the typical pendulum measurement models, we consider an abundance of measurement information with non-physical sensors. This is mainly done to demonstrate that the projection optimization can be run online for highly nonlinear measurement models. Let

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{e}_k = \begin{bmatrix} \sin(x_{1,k} + 0.1) \\ \sin(x_{1,k} + 0.2) \\ \vdots \\ \sin(x_{1,k} + 0.9) \\ x_{2,k} \end{bmatrix} + \mathbf{e}_k \in \mathbb{R}^{10},$$

where the measurement noise \mathbf{e}_k is Gaussian zero mean $\mathbf{e}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ with the covariance \mathbf{R} being a random, dense positive definite matrix satisfying $0.1\mathbf{I} \prec \mathbf{R} = \mathbf{R}^\top \prec 0.3\mathbf{I}$.

Given this model, we compare the state estimates using a large collection of ADFs, with and without the application of a projection operator to the measurements. Here, the measurement model is 10-dimensional, and we let $M = 2$ and perform the optimization using the gradient-based approach in Proposition 1, updating the projection operator with one gradient descent step per time-step. We consider

- EKF - As in [17], see Chapter 5.2.
- SOEKF - In [9], using analytical Hessian evaluations.
- UKF - In [10], with parameters $\alpha = 1, \kappa = 0, \beta = 2$.
- CKF - In [12], with a 3rd-order spherical cubature rule.
- GHKF - As in [17], see Chapter 6.3, here we use the 4th-order Gauss-Hermite integral approximation.
- RUKF - As in [14], approximating the moments by a stochastic integration rule [18], here $N = 20$ iterations.

The system is initialized at $\mathbf{x}_0 = (2, -1)^\top$, and the filters are initialized with substantial initial errors, with a mean $\mathbf{m}_{0|0}^x = (0, 0)^\top$ and covariance $\Sigma_{0|0}^{xx} = 10\mathbf{I}$. The projection operator is initialized at $k = 0$ by solving the non-convex optimization problem in (17) using an interior point method, with the FIM evaluated about the initial estimate mean. Then, Ψ_k is subsequently updated online by a single gradient step per time-step of the filter using 1. Before discussing the MC-results, we present the state estimate for a one simulation with the RUKF ($N = 20$) using the projected measurements and online adaption of Ψ_k , is shown in Fig. 4.

The CRB ratio changes in time by the proposed gradient descent algorithm. However, this change is very slight, and the cost ratio is 1 down to the fourth decimal at all times (third subplot). However, to keep the ratio low, the operator Ψ_k changes significantly in time (fourth subplot).

This demonstrates that the algorithm works as intended for the RUKF, but to validate it for all of the listed filters, we need to compute the MSE empirically with and without the measurement reduction. To this end, the filters are run on the

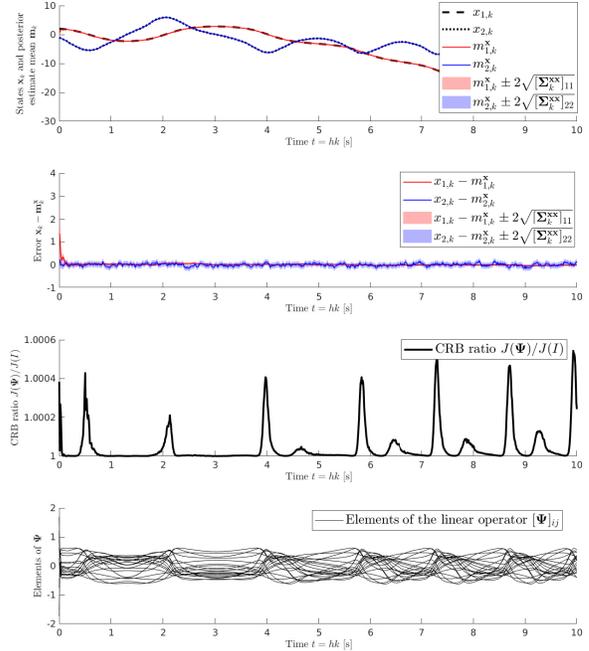


Fig. 4. One of the simulations from the MC-executions in Section V-D, here for the RUKF. *Top*: State trajectories of the pendulum (black) with the means of the state-estimate (blue/red) with the estimated 2σ -interval. *Top, center*: State estimate error (blue/red) with the estimated 2σ -interval. *Bottom, center*: CRB ratio $J(\Psi_k)/J(\mathbf{I})$, the relative increase in the lower bound on the estimate MSE when using Ψ_k as opposed to using the full measurement vector. *Bottom*: Ψ_k , updated in time by Proposition 1.

exact same measurement data over $N_{MC} = 10^2$ MC-runs, the resulting empirically computed MSE of the state estimate is plotted as a function of the time-step k , as shown in Fig. 5.

Here we note a substantial difference in the transients between the filters using Ψ_k , and their original counterparts utilizing the full 10-dimensional measurement vector. This is mainly due to the substantial initial error in the estimate mean, about which Ψ_0 is computed. However, as the estimate improves, the projection operator adapts accordingly, and for large k there is an extremely small difference between the MSE of the estimates computed with the projection operator (using a two-dimensional projected measurement) and the filters using the full 10-dimensional measurement vector.

When studying the estimates more closely, we see that the SOEKF, RUKF, and GHKF are slightly superior to the other filters for larger k , and especially superior in the transient phase. And we also note that the filtering with the projected two-dimensional measurements results in an empirical MSE which is extremely close to the corresponding filters using the full 10-dimensional measurement vector for large k . In other words, the projection operator is adapted as intended, and the number of measurements used in the update has been reduced by a factor of 5 for a slight increase in MSE.

This example serves to show the wide applicability of the method, and that it can be applied to non-linear Gaussian filtering. However, from a computational point of view, it

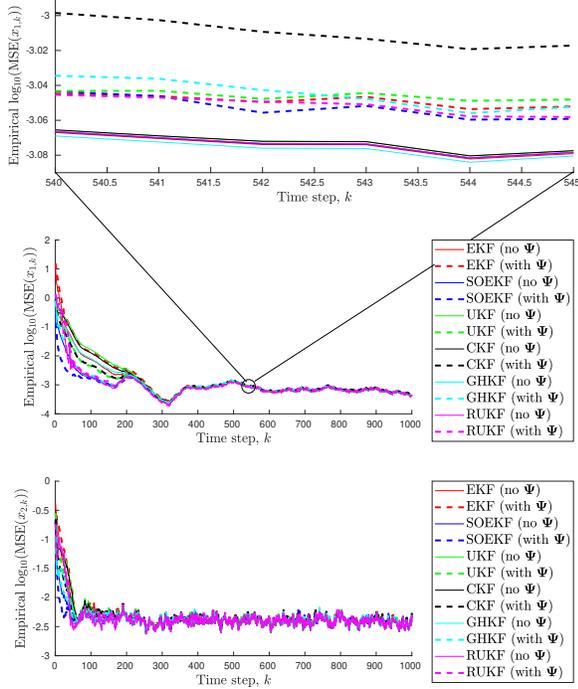


Fig. 5. Results of an MC-study with the various non-linear Gaussian ADFs in Section V-D, showing the empirically computed MSE of the estimate using the original filter (full) and the same filter with projected measurements (dashed). *Top*: Enlarged view of the MSE of the angular state at $k = 540$. *Center*: MSE of the angular state as estimated by the filters. *Bottom*: MSE of the angular velocity state as estimated by the filters.

will mainly make sense to use it for very-high-dimensional measurement models, or in mixture model Kalman filtering and RBPFs, where the same projection operator can be used for all particles. In such a setting, the computational gains in reducing the complexity of the measurement update far exceed the additional computational burden of updating the linear map every time-step for large M .

VI. CONCLUSIONS

In this paper, we consider the problem of filtering using Gaussian ADFs with high-dimensional measurement equations subject to computational constraints. We propose a structured way of determining linear operators, whose application to the measurements minimally increase the Cramér Rao bound of the MSE of the state-estimate while reducing the computational complexity of the measurement updates.

The choice of \tilde{M} needs to be done with care. A lower bound on M can be computed for which the system states become observable through the projected measurements. For the example in Section V-B, this implies that $\tilde{M} \geq 3$. Beyond this, the selection of \tilde{M} should be chosen low, so as to yield a sufficiently small degradation of the MSE CRB.

In the linear case, and for high-dimensional measurement problems, we can typically reduce the computational complexity of the measurement update by several factors, with a quantifiable decrease in MSE performance. However, the feasibility of this approach depends greatly on the system

at hand and the structure of the measurement equation. Nonetheless, it is a valuable tool when for real-time implementation of KFs subject to computational constraints.

We also presented a method for on-line adaption of the linear map in the event of non-linear measurement equations. This was subsequently illustrated by a simulation example with the EKF, SOEKF, UKF, CKF, GHKF, and RUKF. However, the method is likely best exploited for Kalman mixture model filtering or RBPFs. Utilizing the scheme in such a context will be the subject of future research. In addition, the cost function and the resulting optimization methods should be studied further, and the method's sensitivity to modelling errors in the measurement equation should be investigated.

VII. ACKNOWLEDGEMENT

The authors would like to thank the reviewers for constructive comments that helped to improve the paper's quality.

APPENDIX

For matrices \mathbf{X} and \mathbf{Y} of compatible dimensions,

$$\partial(\mathbf{X} + \mathbf{Y}) = \partial\mathbf{X} + \partial\mathbf{Y}, \quad (25)$$

$$\partial(\mathbf{X}\mathbf{Y}) = (\partial\mathbf{X})\mathbf{Y} + \mathbf{X}(\partial\mathbf{Y}), \quad (26)$$

and if the matrix \mathbf{X} is square with eigenvalues and eigenvectors $\lambda_j(\mathbf{X})\mathbf{v}_j = \mathbf{X}\mathbf{v}_j$, we have the identity

$$\partial(\lambda_j(\mathbf{X})) = \mathbf{v}_j^\top \partial(\mathbf{X})\mathbf{v}_j. \quad (27)$$

In deriving the matrix derivatives of the CRB, we will make use of a rectangular matrix \mathbf{J}^{mn} , in which all the entries are zero, except for a single element which is set to one. This matrix is defined as the outer product of two unit vectors on the form $\mathbf{e}_m^M \in \mathbb{R}^{M \times 1}$ with the m^{th} entry set to 1, such that

$$\mathbf{J}^{mn} = \mathbf{e}_m^M (\mathbf{e}_n^N)^\top \in \mathbb{R}^{M \times N}. \quad (28)$$

With the unit vectors and a matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$, where the m^{th} row is $\mathbf{A}_{m,:}$ and the n^{th} column is $\mathbf{A}_{:,n}$, we have

$$(\mathbf{e}_m^M)^\top \mathbf{A} = \mathbf{A}_{m,:} \in \mathbb{R}^{1 \times N}, \quad \mathbf{A} \mathbf{e}_n^N = \mathbf{A}_{:,n} \in \mathbb{R}^{M \times 1}. \quad (29a)$$

Furthermore, with this singleton matrix notation,

$$\frac{\partial(\mathbf{X}\mathbf{A})}{\partial[\mathbf{X}]_{mn}} = \mathbf{J}^{mn} \mathbf{A}, \quad \frac{\partial(\mathbf{A}^\top \mathbf{X}^\top)}{\partial[\mathbf{X}]_{mn}} = \mathbf{A}^\top (\mathbf{J}^{mn})^\top, \quad (30a)$$

which, using (26) and (30), yields

$$\frac{\partial \mathbf{X} \mathbf{R} \mathbf{X}^\top}{\partial[\mathbf{X}]_{mn}} = \mathbf{X} \mathbf{R} (\mathbf{J}^{mn})^\top + \mathbf{J}^{mn} \mathbf{R} \mathbf{X}^\top, \quad (31)$$

which holds under the assumption that $\mathbf{R}^\top = \mathbf{R}$. Furthermore, we have the following well known inverse identity

$$\frac{\partial \mathbf{Y}^{-1}}{\partial[\mathbf{X}]_{mn}} = -\mathbf{Y}^{-1} \frac{\partial \mathbf{Y}}{\partial[\mathbf{X}]_{mn}} \mathbf{Y}^{-1}. \quad (32)$$

See [19] for additional details on the above identities.

Lemma 1: For any $0 \prec \mathbf{Y}(\Psi) = \mathbf{Y}(\Psi)^\top \in \mathbb{R}^{K \times K}$,

$$\text{Tr}(\mathbf{Y}(\Psi)^{-1}) = \sum_j^K \frac{1}{\lambda_j(\mathbf{Y}(\Psi))}. \quad (33)$$

Proof: If $0 \prec \mathbf{Y}(\Psi) = \mathbf{Y}(\Psi)^\top$, there exists an Schur-decomposition $\mathbf{Y}(\Psi) = \mathbf{L}\mathbf{\Lambda}\mathbf{L}^\top$ where $\mathbf{\Lambda}$ is a diagonal matrix with entries $\lambda_j(\mathbf{Y}(\Psi))$ on its diagonal and $\mathbf{L}^\top = \mathbf{L}^{-1}$, whereby $\text{Tr}(\mathbf{Y}(\Psi)^{-1}) = \text{Tr}(\mathbf{\Lambda}^{-1}\mathbf{L}^\top\mathbf{L}) = \text{Tr}(\mathbf{\Lambda}^{-1})$. ■

Lemma 2: Let $\Psi \in \mathbb{R}^{M \times N}$ be a real-valued rectangular matrix with elements $[\Psi]_{mn}$. Let $0 \prec \mathbf{Y}(\Psi) \in \mathbb{R}^{K \times K}$, with eigenvalues $\lambda_j(\mathbf{Y}(\Psi))$ and eigenvectors \mathbf{v}_j , then

$$\frac{\partial \text{Tr}([\mathbf{Y}(\Psi)]^{-1})}{\partial [\Psi]_{mn}} = - \sum_{j=1}^K \frac{1}{\lambda_j(\mathbf{Y}(\Psi))^2} \mathbf{v}_j^\top \left(\frac{\partial \mathbf{Y}(\Psi)}{\partial [\Psi]_{mn}} \right) \mathbf{v}_j.$$

Proof: Omitting the argument Ψ , application of Lemma 1 and the matrix identities, yields

$$\frac{\partial \text{Tr}(\mathbf{Y}^{-1})}{\partial [\Psi]_{mn}} = \frac{\partial}{\partial [\Psi]_{mn}} \left(\sum_{j=1}^K \frac{1}{\lambda_j(\mathbf{Y})} \right) \quad (34a)$$

$$\stackrel{(25)}{=} \sum_{j=1}^K \frac{\partial}{\partial [\Psi]_{mn}} \frac{1}{\lambda_j(\mathbf{Y})} \quad (34b)$$

$$\stackrel{(32)}{=} - \sum_{j=1}^K \frac{1}{\lambda_j(\mathbf{Y})^2} \frac{\partial \lambda_j(\mathbf{Y})}{\partial [\Psi]_{mn}} \quad (34c)$$

$$\stackrel{(27)}{=} - \sum_{j=1}^K \frac{1}{\lambda_j(\mathbf{Y})^2} \mathbf{v}_j^\top \left(\frac{\partial \mathbf{Y}}{\partial [\Psi]_{mn}} \right) \mathbf{v}_j \quad (34d)$$

Lemma 3: If $0 \prec \mathbf{R} \in \mathbb{R}^{N \times N}$, with rectangular matrices $\Psi \in \mathbb{R}^{N \times M}$, and $\mathbf{H} \in \mathbb{R}^{M \times K}$ then

$$\frac{\partial (\mathbf{Y})}{\partial [\Psi]_{mn}} = \mathbf{U}^\top \mathbf{J}^{mn} \mathbf{V} + (\mathbf{U}^\top \mathbf{J}^{mn} \mathbf{V})^\top \quad (35)$$

where

$$\mathbf{Y} = \mathbf{H}^\top \Psi^\top (\Psi \mathbf{R} \Psi^\top)^{-1} \mathbf{P} \mathbf{H} \quad (36a)$$

$$\mathbf{U} = (\Psi \mathbf{R} \Psi^\top)^{-1} \Psi \mathbf{H}, \quad (36b)$$

$$\mathbf{V} = \mathbf{H} - \mathbf{R} \Psi^\top \mathbf{U}. \quad (36c)$$

Proof: Follows from application of the identities (26), (32), (36b), (30), (31) and (36c) to (35) in that order. ■

Proof: To prove Proposition 1, we first derive the equations for the partial derivatives with respect to single elements $[\Psi]_{mn}$ of Ψ and then generalize it to the full matrix derivative on the form in (18). By application of Lemma 2 and 3, we obtain the second and third equality, such that

$$\frac{\partial \text{Tr}((\mathbf{Y})^{-1})}{\partial [\Psi]_{mn}} \stackrel{(34a)}{=} - \sum_{j=1}^K \frac{1}{\lambda_j(\mathbf{Y})^2} \mathbf{v}_j^\top \left(\frac{\partial \mathbf{Y}}{\partial [\Psi]_{mn}} \right) \mathbf{v}_j$$

$$\stackrel{(3)}{=} - \sum_{j=1}^K \frac{1}{\lambda_j(\mathbf{Y})^2} \mathbf{v}_j^\top \left(\mathbf{U}^\top \mathbf{J}^{mn} \mathbf{V} + (\mathbf{U}^\top \mathbf{J}^{mn} \mathbf{V})^\top \right) \mathbf{v}_j$$

$$\stackrel{(28)}{=} - \sum_{j=1}^K \frac{1}{\lambda_j(\mathbf{Y})^2} \mathbf{v}_j^\top \left(\mathbf{U}^\top \mathbf{e}_m^M (\mathbf{e}_n^N)^\top \mathbf{V} + (\mathbf{U}^\top \mathbf{e}_m^M (\mathbf{e}_n^N)^\top \mathbf{V})^\top \right) \quad (35)$$

$$\stackrel{(29)}{=} - \sum_{j=1}^K \frac{1}{\lambda_j(\mathbf{Y})^2} \mathbf{v}_j^\top \left((\mathbf{U}_{m,:})^\top \mathbf{V}_{n,:} + ((\mathbf{U}_{m,:})^\top \mathbf{V}_{n,:})^\top \right)$$

$$= -2 \sum_{j=1}^K [\mathbf{U}_{m,:} \mathbf{v}_j] \frac{1}{\lambda_j(\mathbf{Y})^2} [\mathbf{v}_j^\top \mathbf{V}_{n,:}^\top] := D_{mn}$$

where D_{mn} is recognized as the element at the m^{th} row and n^{th} column of $\mathbf{D} = -2\mathbf{U}\mathbf{L}\mathbf{\Lambda}^{-2}\mathbf{L}^\top\mathbf{V}^\top$. Thus,

$$\frac{\partial \text{Tr}(\mathbf{Y}^{-1})}{\partial \Psi} = \mathbf{D} = -2\mathbf{U}\mathbf{L}\mathbf{\Lambda}^{-2}\mathbf{L}^\top\mathbf{V}^\top, \quad (37)$$

which concludes the proof. Since \mathcal{P}_k does not depend on Ψ , the proof is trivially extendable to the case where $\mathcal{P}_k \neq \mathbf{0}$. ■

REFERENCES

- [1] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, *Global Positioning System: theory and practice*. Springer-Verlag, 1997.
- [2] S. Zhao, X. Cui, F. Guan, and M. Lu, "A Kalman filter-based short baseline RTK algorithm for single-frequency combination of GPS and BDS," *Sensors*, vol. 14, no. 8, pp. 15 415–15 433, 2014.
- [3] A. S. Paul and E. A. Wan, "RSSI-based indoor localization and tracking using sigma-point Kalman smoothers," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 5, pp. 860–873, 2009.
- [4] A. Banerjee, D. B. Dunson, and S. T. Tokdar, "Efficient Gaussian process regression for large datasets," *Biometrika*, vol. 100, no. 1, pp. 75–89, 2012.
- [5] S. M. Kay, *Fundamentals of statistical signal processing*. Prentice Hall PTR, 1993.
- [6] C. Fritsche, E. Özkan, L. Svensson, and F. Gustafsson, "A fresh look at Bayesian Cramér-Rao bounds for discrete-time nonlinear filtering," in *17th International Conference on Information Fusion (FUSION)*. IEEE, 2014, pp. 1–8.
- [7] T. Schön, 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.
- [8] K. Berntorp, A. Weiss, and S. Di Cairano, "GNSS ambiguity resolution by adaptive mixture Kalman filter," in *21st International Conference on Information Fusion*, Cambridge, UK, Jul. 2018.
- [9] M. Roth and F. Gustafsson, "An efficient implementation of the second order extended Kalman filter," in *14th International Conference on Information Fusion*. IEEE, 2011, pp. 1–6.
- [10] E. A. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373)*. Ieee, 2000, pp. 153–158.
- [11] 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.
- [12] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Transactions on automatic control*, vol. 54, no. 6, pp. 1254–1269, 2009.
- [13] K. Ito and K. Xiong, "Gaussian filters for nonlinear filtering problems," *IEEE transactions on automatic control*, vol. 45, no. 5, pp. 910–927, 2000.
- [14] J. Duník, O. Straka, and M. Šimandl, "The development of a randomised unscented Kalman filter," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 8–13, 2011.
- [15] P. Tichavsky, C. H. Muravchik, and A. Nehorai, "Posterior Cramér-Rao bounds for discrete-time nonlinear filtering," *IEEE Transactions on signal processing*, vol. 46, no. 5, pp. 1386–1396, 1998.
- [16] A. Wächter and L. T. Biegler, "Line search filter methods for nonlinear programming: Local convergence," *SIAM Journal on Optimization*, vol. 16, no. 1, pp. 32–48, 2005.
- [17] S. Särkkä, *Bayesian filtering and smoothing*. Cambridge University Press, 2013, vol. 3.
- [18] A. Genz and J. Monahan, "Stochastic integration rules for infinite regions," *SIAM journal on scientific computing*, vol. 19, no. 2, pp. 426–439, 1998.
- [19] K. B. Petersen, M. S. Pedersen *et al.*, "The matrix cookbook," *Technical University of Denmark*, vol. 7, no. 15, p. 510, 2008.