Gaussian MAP filtering using Kalman optimisation

Ángel F. García-Fernández, Lennart Svensson

***Abstract*—This paper deals with the update step of Gaussian MAP filtering. In this framework, we seek a Gaussian approximation to the posterior probability density function (PDF) whose mean is given by the maximum a posteriori (MAP) estimator. We propose two novel optimisation algorithms which are quite suitable for finding the MAP estimate although they can also be used to solve general optimisation problems. These are based on the design of a sequence of PDFs that become increasingly concentrated around the MAP estimate. The resulting algorithms are referred to as Kalman optimisation (KO) methods. We also provide the important relations between these KO methods and their conventional optimisation algorithms (COAs) counterparts, i.e., Newton´s and Levenberg-Marquardt algorithms. Our simulations indicate that KO methods are more robust than their COA equivalents.**

***Index Terms*—Kalman filter, MAP estimation, Bayesian nonlinear filtering, optimisation**

I. Introduction

In Bayesian filtering, we estimate the current state of a Markov process based on a sequence of measurements up to the current time. In this setting, knowledge of the probability density function (PDF) of the current state given the measurements up to the current time is of utmost importance as any optimal estimator requires it. In theory, this PDF, which is referred to as posterior PDF, can be calculated recursively in two steps: prediction and update. The problem however is that this recursion is analytically intractable in most nonlinear/non-Gaussian systems and must be approximated.

In many cases, the posterior is unimodal and, consequently, Gaussian approximations can be of high accuracy. As a result, Gaussian approximations with low computational burden draw considerable attention. In the Gaussian case, the prediction step is relatively simple as it comes down to approximating the first two moments of a random variable that undergoes a transformation [1]. The update step is more difficult and there are several options. In the next two paragraphs, we discuss two of these alternatives.

The most extensively used choice in the update step is the Kalman filter (KF). Even though the KF is most commonly known in the filtering literature as the solution to the linear/Gaussian filtering recursion, the KF can also be applied to obtain a Gaussian posterior approximation for nonlinear/non-Gaussian systems [2]. In the KF update step, the posterior mean and covariance matrix are approximated by the linear minimum mean square error (LMMSE) estimator and its mean square error matrix, respectively. The KF is usually intractable if the measurement function is nonlinear so approximations to the KF have been developed, e.g., extended KF (EKF) [3], unscented KF (UKF) [2], cubature KF (CKF) [4] or quadrature KF (QKF) [5].

The biggest problem of the KF and its approximations in a nonlinear update is that their performance is poor if the measurement noise is low enough [6]. Therefore, in this case, another kind of approximation should be sought, e.g., based on the maximum a posteriori (MAP) estimator. A Gaussian posterior PDF approximation whose mean is the MAP estimator and its covariance matrix is given by minus the inverse Hessian of the log-posterior is asymptotically optimal as the measurement noise tends to zero [7]. We refer to this approach of approximating the posterior as Gaussian MAP filtering and constitutes the focus of this paper. In the next paragraphs, we explain usual algorithms to obtain the MAP estimate, the novel algorithms we propose in this paper and the benefits of using our methods in the Gaussian MAP filtering framework.

The MAP estimate is the solution to an unconstrained optimisation problem. If the problem does not admit an analytical solution, conventional optimisation algorithms (COAs) create a sequence of vectors that should converge to the solution. Algorithms of this type are, for example, Newton's, steepest descent, Gauss-Newton or Levenberg-Marquardt [8]. The most well-known Gaussian MAP filter is the iterated EKF (IEKF), which is based on the Gauss-Newton optimisation method [9]. In [10], the Levenberg-Marquardt (LM) method [8] is used instead of Gauss-Newton to increase robustness. Other COAs for Gaussian MAP filtering are discussed in [11].

Motivated by the fact that sigma-point methods (e.g. UKF and CKF) generally outperform the EKF [2], several authors have developed iterated sigma-point methods to find the MAP estimate although with ad-hoc approaches. In [12], sigma points are only used once to approximate the (usual) KF moments. Then, the iteration to find the MAP estimate corresponds to the IEKF iteration but changing the analytical linearisation of the measurement function at the current MAP estimate, used by the IEKF, by an ad-hoc linearisation that mixes statistitical linear regression w.r.t. the prior [13] and analytical linearisation at the current MAP estimate. The iterated UKF (IUKF) in [14] requires several ad-hoc parameters and conditions. In addition, the resulting iteration has the effect of performing several corrections with the same measurement even though we only observe it once. On the contrary, this paper proposes two iterated sigma-point Gaussian MAP filters with sound mathematical foundations. This is explained in the following.

Ángel F. García-Fernández was with the Department of Signals and Systems, Chalmers University of Technology, SE-412 96 Gothenburg, Sweden. He is now with the Department of Electrical and Computer Engineering, Curtin University, Perth, WA 6102, Australia (e-mail: angel.garciafernandez@curtin.edu.au). Lennart Svensson is with the Department of Signals and Systems, Chalmers University of Technology, SE-412 96 Gothenburg, Sweden (email: lennart.svensson@chalmers.se).

Unconstrained optimisation problems can also be solved based on the following fact: if a PDF is raised to a power greater than one and renormalised, the resulting PDF becomes increasingly concentrated on the mode(s) of the original distribution as the power increases [15]. This is the foundation of some stochastic optimisation (SO) algorithms, e.g. simulated annealing [16], [17], which are not of practical use in Gaussian MAP filtering due to their high computational complexity. In this paper, we propose two novel optimisation algorithms which are based on convergent sequences of PDFs as in SO. These sequences, which are based on the consecutive application of Bayes' rule have not been reported in the literature yet. Importantly, we use a deterministic rather than a stochastic approximation to these PDFs based on the KF update [2]. This framework for optimisation is referred to as Kalman optimisation (KO).

The first KO algorithm proposed in this paper can be applied to any optimisation problem and is referred to as Newton KO (NKO). The second KO algorithm, which is called Levenberg-Marquardt KO (LMKO), is designed specifically for least squares problems. An important contribution of this paper, apart from the fact of designing NKO and LMKO, is that we show the connections between NKO and LMKO and COAs. In NKO, the mean of the sequence of PDFs is equivalent to Newton's method, but instead of using the gradient and Hessian, we use the expected gradient and Hessian, while accounting for the normalised expected square nonlinearities and noise. In LMKO, the mean of the sequence of the PDFs is equivalent to the LM method, but substituting the function and its gradient by the expected function and gradient, while accounting for both the expected square nonlinearities and prior knowledge.

We want to clarify that connections between the linear KF and Newton's method are known [18]. However, as we indicated before, we do not propose the use of COAs, such as Newton's method, to find the MAP estimate. Instead, we develop two novel algorithms in the (nonlinear) KO framework. Stochastic gradient methods do a form of averaging of the gradient over the iterations, instead of averaging at each iteration as in KO [17]. Nevertheless, they are applicable in a different context, in which we can only evaluate noisy versions of the function to optimise and its gradient. In situations where it is time consuming to evaluate the function of interest, it can be advantageous to provide a random model for the function using, for example, Gaussian processes [19]. However, these methods have a high computational burden and are not convenient for usual Gaussian MAP filtering problems, in which the functions are fast to evaluate.

In practice, the KF moments of KO are approximated using sigma-point methods [2], [4], [5]. It is quite convenient to use NKO and LMKO instead of their COAs counterparts to find the MAP because of two reasons. The first one is mathematical simplicity: NKO does not require the calculation of the Hessian, while Newton´s method does, and LMKO does not require the calculation of the gradient, while LM does. The second is robustness: sigma-point approximations of the KF update step are known to be more robust than those based on analytical linearisation [1]. These benefits are analysed thoroughly in the paper and the excellent performances of the algorithms are demonstrated via simulations in Section VII.

The rest of the paper is organised as follows. In the first part of the paper, we only deal with optimisation problems and propose two novel optimisation algorithms based on the KF update. In Section II, we provide the general ideas behind KO. Section III introduces the NKO and Section IV presents the LMKO. The second part of this paper deals with the application of NKO and LMKO to Gaussian MAP nonlinear filtering update. In Section V, we explain how to approximate the first two moments of the posterior based on the NKO and LMKO output. In Section VI, we provide a local convergence analysis of NKO and LMKO in this setting. Simulation results are provided in Section VII. Finally, conclusions are drawn in Section VIII.

II. Overview of Kalman optimisation

The objective of this section is to give a general idea of Bayes' rule optimisation (BO) and Kalman optimisation (KO). The aim is to find the global minimum $\tilde{\mathbf{x}}^{\star} \in \mathbb{R}^{n_x}$, which is assumed to be unique, of a differentiable function $f(\cdot) : \mathbb{R}^{n_x} \to \mathbb{R}$

$$\tilde{\mathbf{x}}^{\star} = \underset{\mathbf{x}\in\mathbb{R}^{n_x}}{\arg\min} f(\mathbf{x}). \tag{1}$$

If $\tilde{\mathbf{x}}^{\star}$ cannot be calculated analytically, COAs create a sequence of vectors $\tilde{\mathbf{x}}_0, \tilde{\mathbf{x}}_1, ..., \tilde{\mathbf{x}}_i$ that should converge to $\tilde{\mathbf{x}}^{\star}$ [8]. BO and KO depart from this approach and use instead a sequence of PDFs. First, we describe BO and then we proceed to explain KO.

BO starts with an initial PDF

$$p_0(\tilde{\mathbf{x}}) = \mathcal{N}(\tilde{\mathbf{x}}; \overline{\mathbf{x}}_0, \mathbf{P}_0) \tag{2}$$

where $\mathcal{N}(\tilde{\mathbf{x}}; \overline{\mathbf{x}}_0, \mathbf{P}_0)$ is the Gaussian PDF evaluated at $\tilde{\mathbf{x}}$ with mean $\overline{\mathbf{x}}_0$ and covariance matrix $\mathbf{P}_0$. BO uses a convergent sequence of PDFs $\pi_1(\cdot)$,..., $\pi_i(\cdot)$ based on the successive application of Bayes' rule. In other words, in BO, we design a sequence of likelihood functions $l_i^{\mathbf{y}_i}(\cdot)$, where $\mathbf{y}_i$ represents the ith measurement, such that the jth PDF is

$$\pi_j(\tilde{\mathbf{x}}) \propto \prod_{i=0}^{j-1} l_i^{\mathbf{y}_i}(\tilde{\mathbf{x}}) \, p_0(\tilde{\mathbf{x}}) \tag{3}$$

where $\propto$ means "proportional to" and as j tends to infinity this sequence converges to

$$c(\tilde{\mathbf{x}}) = \delta(\tilde{\mathbf{x}} - \tilde{\mathbf{x}}^{\star}) \tag{4}$$

where $\delta(\cdot)$ is the Dirac delta. This allows us to find $\tilde{\mathbf{x}}^{\star}$. By considering independent measurements given the state in (3), we can iteratively approximate this sequence easily and establish connections with COAs. In Sections III-A and IV-A, we indicate how to create such sequence of likelihoods for NKO and LMKO.

The sequence of PDFs given by (3) is usually intractable to calculate analytically so approximations must be used. A common and computationally efficient method to approximate Bayes' rule is to use the KF update [2]. The resulting method is called KO, which creates a sequence $p_1(\cdot)$,..., $p_i(\cdot)$ of Gaussian PDFs that approximates $\pi_1(\cdot)$,..., $\pi_i(\cdot)$. This sequence

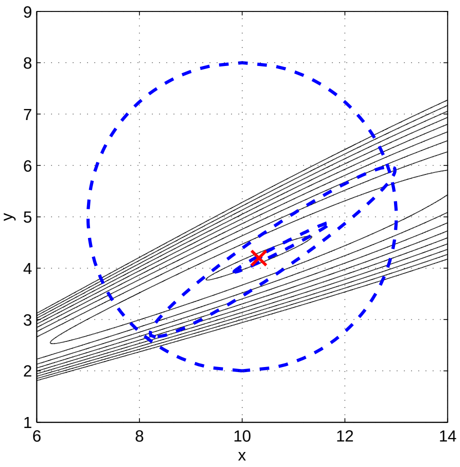


Figure 1: Illustration of KO. The minimum of the function is shown as a red cross. Ten contour plots of the function $f(\cdot)$ around the minimum are shown as black lines. The 3σ-ellipses of the PDF iteration are shown as dashed blue lines.

also converges to $c(\cdot)$ under some assumptions, which depend on the KO algorithm[1]. An illustrative KO sequence is shown in Figure 1. The initial PDF is represented as the outer 3σ-ellipse. As the iteration continues, the ellipses get smaller and smaller around $\tilde{\mathbf{x}}^{\star}$.

Due to their suitability, the main application of KO algorithms is in the update step of Gaussian MAP filtering, as will be thoroughly explained in Section V. In the update step of Bayesian filtering, we combine a prior PDF and a measurement to produce the posterior PDF. In Gaussian MAP filtering, the posterior is approximated as a Gaussian with the mean provided by the MAP estimate and a certain posterior covariance matrix. In this set-up, we want to remark that KO is mainly used to obtain the MAP estimate, which is the solution of an optimisation problem, though KO implicitly provides an approximation of the posterior covariance matrix, which will be given in Section V. This is not the covariance matrix of the ith PDF, which always tends to zero as shown in Figure 1. It is also important to highlight that the sequence of PDFs (3) is just an artificial construction to solve optimisation problems and should not be confused with the sequence of PDFs of Gaussian MAP filtering.

III. Newton Kalman optimisation

The aim is to find the minimum $\tilde{\mathbf{x}}^{\star}$ of $f(\cdot)$. As Newton's method, NKO uses the fact that

$$\nabla f\left(\tilde{\mathbf{x}}^{\star}\right)=\mathbf{0} \tag{5}$$

where $\nabla f(\cdot)$ is the gradient of $f(\cdot)$. If $f(\cdot)$ has only one stationary point, NKO should converge to the global minimum as there is only one point that meets (5). If not, it should also converge to the global minimum if it is initialized well but, in general, it could converge to a local minimum, local maximum or inflection point. Newton's method also has the same drawback [8].

This section is organised in the following manner. In Section III-A, we explain how the sequence of PDFs given by (3) is designed. In Section III-B, we explain the theoretical aspects of NKO. We include a practical NKO algorithm in Section III-C.

A. Bayes' rule optimisation

In this section, we design a sequence of PDFs as in (3) that converges to the minimum. The sequence of measurement equations is

$$\mathbf{y}_i=\nabla f\left(\tilde{\mathbf{x}}\right)+\boldsymbol{\eta}_i \tag{6}$$

where $\mathbf{y}_i \in \mathbb{R}^{n_x}$ and $\boldsymbol{\eta}_i \in \mathbb{R}^{n_x}$ is a sequence of independent zero-mean Gaussian measurement noise with covariance matrix $\beta_i^2\mathbf{I}$, $\beta_i^2>0$, which implies that $l_i^{\mathbf{y}_i}\left(\tilde{\mathbf{x}}\right) \propto \mathcal{N}\left(\mathbf{y}_i;\nabla f\left(\tilde{\mathbf{x}}\right),\beta_i^2\mathbf{I}\right)$. We select $\beta_i^2\mathbf{I}$ as the noise covariance matrix for convenience although more general approaches might be possible. This way, we can easily ensure convergence of BO and select its value properly in KO, as will be explained in Section III-B1.

By fixing the measurement to $\mathbf{y}_i=\mathbf{0}$, the sequence of likelihood functions in (3) becomes $l_i^{\mathbf{0}}\left(\tilde{\mathbf{x}}\right)\propto\mathcal{N}\left(\mathbf{0};\nabla f\left(\tilde{\mathbf{x}}\right),\beta_i^2\mathbf{I}\right)$. Setting the measurement to a value is a procedure that is not based on purely Bayesian arguments. However, as is explained in the following, it allows us to minimise $f(\cdot)$ by the sequential application of Bayes' rule, which is our objective. The ith PDF of the BO sequence (3) becomes

$$\begin{aligned}\pi_i\left(\tilde{\mathbf{x}}\right)\propto\exp\Bigg(&-1/2\left(\nabla f\left(\tilde{\mathbf{x}}\right)\right)^T\nabla f\left(\tilde{\mathbf{x}}\right)\sum_{j=0}^{i-1}\frac{1}{\beta_j^2}\\&-1/2\left(\tilde{\mathbf{x}}-\overline{\mathbf{x}}_0\right)^T\mathbf{P}_0^{-1}\left(\tilde{\mathbf{x}}-\overline{\mathbf{x}}_0\right)\Bigg).\end{aligned} \tag{7}$$

As indicated in Section II, we need to prove that this sequence is progressively more concentrated around the minimum such that it can be used for optimisation purposes. If β_j is bounded, as i increases $1/2\left(\nabla f\left(\tilde{\mathbf{x}}\right)\right)^T\nabla f\left(\tilde{\mathbf{x}}\right)\sum_{j=0}^{i-1}\frac{1}{\beta_j^2}$ gets greater and greater, the term $1/2\left(\tilde{\mathbf{x}}-\overline{\mathbf{x}}_0\right)^T\mathbf{P}_0^{-1}\left(\tilde{\mathbf{x}}-\overline{\mathbf{x}}_0\right)$ becomes negligible and (7) can be approximated as [7]

$$\begin{aligned}\pi_i\left(\tilde{\mathbf{x}}\right)\approx&\mathcal{N}\left(\tilde{\mathbf{x}};\tilde{\mathbf{x}}^{\star},\mathbf{P}_{q,i}\right)\\\tilde{\mathbf{x}}^{\star}=&\arg\min_{\tilde{\mathbf{x}}}\left(\nabla f\left(\tilde{\mathbf{x}}\right)\right)^T\nabla f\left(\tilde{\mathbf{x}}\right)\approx\arg\max_{\tilde{\mathbf{x}}}\log\pi_i\left(\tilde{\mathbf{x}}\right)\\\mathbf{P}_{\pi,i}^{-1}=&-\left.\nabla^2\log\pi_i\left(\tilde{\mathbf{x}}\right)\right|_{\tilde{\mathbf{x}}^{\star}}\\\approx&\nabla^2\left[1/2\left(\nabla f\left(\tilde{\mathbf{x}}\right)\right)^T\nabla f\left(\tilde{\mathbf{x}}\right)\right]\sum_{j=0}^{i-1}\frac{1}{\beta_j^2}\end{aligned}$$

where ∇^2 stands for the Hessian and $\left(\nabla f\left(\tilde{\mathbf{x}}\right)\right)^T\nabla f\left(\tilde{\mathbf{x}}\right)$ is minimised for $\tilde{\mathbf{x}}=\tilde{\mathbf{x}}^{\star}$ because of (5). If $\sum_{j=0}^{i-1}\frac{1}{\beta_j^2}$ tends to infinity, $\mathbf{P}_{\pi,i}\to\mathbf{0}$, so $\pi_i(\cdot)$ tends to a Dirac delta located on $\tilde{\mathbf{x}}^{\star}$ as in (4). It should be noted that convergence can be attained in one update step by letting $\beta_0^2\to 0$.

[1] The convergence properties of NKO and LMKO are addressed in Section VI.

B. Kalman optimisation

The sequence $\pi_1(\cdot)$,..., $\pi_i(\cdot)$ is intractable to calculate in general because a Bayesian update with measurement equation (6) cannot be obtained in closed-form if $\nabla f(\cdot)$ is nonlinear. Nevertheless, Bayesian updates with nonlinear functions can be approximated by the widely-used KF update, which approximates the updated PDF as Gaussian using the LMMSE estimator [2]. This is the approach followed by NKO, which uses a sequence $p_1(\cdot)$,..., $p_i(\cdot)$ of Gaussian PDFs as an approximation to $\pi_1(\cdot)$,..., $\pi_i(\cdot)$. Given $p_i(\cdot)$ with moments $\overline{\mathbf{x}}_i$ and $\mathbf{P}_i$, we approximate the corresponding updated PDF $p_{i+1}(\cdot)$ with measurement equation (6) and $\mathbf{y}_i = \mathbf{0}$ using the KF update. The first two moments $\overline{\mathbf{x}}_{i+1}$ and $\mathbf{P}_{i+1}$ of the PDF at iteration $i+1$ are calculated as [2]

$$\overline{\mathbf{x}}_{i+1} = \overline{\mathbf{x}}_i + \mathbf{\Psi}_i \mathbf{S}_i^{-1}\left(\mathbf{y}_i - \overline{\mathbf{y}}_i\right)\Big|_{\mathbf{y}_i=0} \tag{8}$$

$$\mathbf{P}_{i+1} = \mathbf{P}_i - \mathbf{\Psi}_i \mathbf{S}_i^{-1}\mathbf{\Psi}_i^T \tag{9}$$

where $\mathbf{S}_i = \mathbf{\Phi}_i + \beta_i^2 \mathbf{I}$ and the KF moments are

$$\overline{\mathbf{y}}_i = \mathsf{E}_i\left[\nabla f(\tilde{\mathbf{x}})\right] = \int \nabla f(\tilde{\mathbf{x}})\, p_i(\tilde{\mathbf{x}})\, d\tilde{\mathbf{x}} \tag{10}$$

$$\begin{aligned}\mathbf{\Phi}_i &= \mathsf{cov}_i\left[\nabla f(\tilde{\mathbf{x}})\right] \\ &= \int \left(\nabla f(\tilde{\mathbf{x}}) - \overline{\mathbf{y}}_i\right)\left(\nabla f(\tilde{\mathbf{x}}) - \overline{\mathbf{y}}_i\right)^T p_i(\tilde{\mathbf{x}})\, d\tilde{\mathbf{x}} \end{aligned}\tag{11}$$

$$\begin{aligned}\mathbf{\Psi}_i &= \mathsf{cov}_i\left[\tilde{\mathbf{x}}, \nabla f(\tilde{\mathbf{x}})\right] \\ &= \int \left(\tilde{\mathbf{x}} - \overline{\mathbf{x}}_i\right)\left(\nabla f(\tilde{\mathbf{x}}) - \overline{\mathbf{y}}_i\right)^T p_i(\tilde{\mathbf{x}})\, d\tilde{\mathbf{x}} \end{aligned}\tag{12}$$

where the subscript i in the moments indicate that they are obtained w.r.t. $p_i(\cdot)$.

1) On the selection of β_i: In principle, it seems appropriate to select β_i as low as possible as the sequence (7) converges faster. However, the KF update does not necessarily perform well if the measurement noise is too low [6]. An indicator of performance of the KF update is the Kullback-Leibler divergence (KLD) D_i of the KF approximation of the joint PDF of the state and the measurement from the true joint PDF. The lower the KLD is, the KF works better. We aim to select β_i such that it gives a small KLD and still allows for reasonably fast convergence.

Let us first write $\nabla f(\cdot)$ as

$$\nabla f(\mathbf{x}) = \mathbf{A}_i \mathbf{x} + \mathbf{b}_i + \mathbf{n}_i(\mathbf{x}) \tag{13}$$

where $\mathbf{A}_i = \mathbf{\Psi}_i^T \mathbf{P}_i^{-1}$, $\mathbf{b}_i = \overline{\mathbf{y}}_i - \mathbf{A}_i \overline{\mathbf{x}}_i$, $\mathbf{n}_i(\cdot)$ represents the nonlinearities of $\nabla f(\cdot)$ and $\mathbf{A}_i$ and $\mathbf{b}_i$ have been chosen by statistical linear regression w.r.t. $p_i(\cdot)$ [5]. Then, D_i is given by[2] [6]

$$D_i = \log\left(\left|\frac{\beta_i^2 \mathbf{I} + \mathbf{N}_i}{\beta_i^2}\right|\right)/2 \tag{14}$$

where

$$\mathbf{N}_i = \mathsf{E}_i\left[\mathbf{n}_i(\tilde{\mathbf{x}})\, \mathbf{n}_i^T(\tilde{\mathbf{x}})\right] = \mathbf{\Phi}_i - \mathbf{A}_i \mathbf{P}_i \mathbf{A}_i^T. \tag{15}$$

It is interesting to note that if $\mathbf{N}_i = \mathbf{0}$, $D_i = 0$, i.e., the KF update provides the optimal solution. As $\mathbf{N}_i$ increases, β_i^2 should increase to keep a given KLD so β_i should depend on $\mathbf{N}_i$.

We denote the eigenvalues of $\mathbf{N}_i$ as $\lambda_1, ..., \lambda_{n_x}$. Then, Equation (14) becomes

$$D_i = \frac{1}{2}\sum_{j=1}^{n_x} \log\left(\frac{\beta_i^2 + \lambda_j}{\beta_i^2}\right).$$

By choosing

$$\beta_i^2 = \Gamma_\beta \max\left(\lambda_1, ..., \lambda_{n_x}\right), \tag{16}$$

we get

$$D_i \le n_x/2 \log\left(\frac{\Gamma_\beta + 1}{\Gamma_\beta}\right),$$

so we can ensure that the KLD is below a given value. The higher Γ_β, the lower D_i so higher performance is expected at the expense of slower convergence.

2) Relation to Newton's method: In this section, we provide the relation of NKO with Newton's method. We can write [5]

$$\mathbf{S}_i = \mathbf{\Psi}_i^T \mathbf{P}_i^{-1} \mathbf{\Psi}_i + \mathbf{N}_i + \beta_i^2 \mathbf{I} \tag{17}$$

Substituting (17) into (8), we obtain

$$\overline{\mathbf{x}}_{i+1} = \overline{\mathbf{x}}_i - \mathbf{\Psi}_i\left(\mathbf{\Psi}_i^T \mathbf{P}_i^{-1}\mathbf{\Psi}_i + \mathbf{N}_i + \beta_i^2 \mathbf{I}\right)^{-1} \mathsf{E}_i\left[\nabla f(\tilde{\mathbf{x}})\right].$$

Under assumption

- A: $\mathbf{\Psi}_i^{-1}$ exists

we can write

$$\overline{\mathbf{x}}_{i+1} = \overline{\mathbf{x}}_i - \left(\mathbf{\Psi}_i^T \mathbf{P}_i^{-1} + \left(\mathbf{N}_i + \beta_i^2 \mathbf{I}\right)\mathbf{\Psi}_i^{-1}\right)^{-1} \mathsf{E}_i\left[\nabla f(\tilde{\mathbf{x}})\right].$$

Using the properties of the Fourier-Hermite series [20], it can be shown that

$$\mathbf{\Psi}_i^T \mathbf{P}_i^{-1} = \mathsf{E}_i\left[\left[\nabla\left(\nabla f(\tilde{\mathbf{x}})\right)^T\right]^T\right] = \mathsf{E}_i\left[\nabla^2 f(\tilde{\mathbf{x}})\right]. \tag{18}$$

Then, the NKO recursion can be written as

$$\overline{\mathbf{x}}_{i+1} = \overline{\mathbf{x}}_i - \left(\mathsf{E}_i\left[\nabla^2 f(\tilde{\mathbf{x}})\right] + \left(\mathbf{N}_i + \beta_i^2 \mathbf{I}\right)\mathbf{\Psi}_i^{-1}\right)^{-1} \mathsf{E}_i\left[\nabla f(\tilde{\mathbf{x}})\right]. \tag{19}$$

The NKO recursion is the same as Newton's method [8] but changing the gradient and the Hessian by expected gradient and expected Hessian and adding a term to the expected Hessian that accounts for the nonlinearities and measurement noise. It should be noted that due to (18), Assumption A is equivalent to assuming that $\left(\mathsf{E}_i\left[\nabla^2 f(\tilde{\mathbf{x}})\right]\right)^{-1}$ exists. Newton's method assumes that $\left(\nabla^2 f(\tilde{\mathbf{x}})\right)^{-1}$ exists at the iteration point so Assumption A is just a generalisation of the assumption required by Newton's.

3) EKF approximation: A usual approximation to the KF moments (10)-(12) is the EKF. It linearises the measurement function (6) around $\overline{\mathbf{x}}_i$ [3]. Taking into account that the Hessian is the Jacobian of the gradient, the EKF approximations to the KF moments become

$$\overline{\mathbf{y}}_i = \nabla f(\overline{\mathbf{x}}_i) \tag{20}$$

$$\mathbf{S}_i = \nabla^2 f(\overline{\mathbf{x}}_i)\, \mathbf{P}_i \left(\nabla^2 f(\overline{\mathbf{x}}_i)\right)^T + \beta_i^2 \mathbf{I} \tag{21}$$

$$\mathbf{\Psi}_i = \mathbf{P}_i \left(\nabla^2 f(\overline{\mathbf{x}}_i)\right)^T. \tag{22}$$

[2] There is a typo in Equation (23) of [6]. It should be divided by 2.

Using (8), the resulting iteration for the mean is

$$\overline{\mathbf{x}}_{i+1} = \overline{\mathbf{x}}_i - \left(\nabla^2 f\left(\overline{\mathbf{x}}_i\right) + \beta_i^2 \mathbf{I}\right)^{-1} \nabla f\left(\overline{\mathbf{x}}_i\right). \tag{23}$$

If $\beta_i^2 = 0$, the EKF approximation to NKO becomes Newton's method. In addition, if we select β_i^2 as indicated in Section III-B1, $\mathbf{N}_i = \mathbf{0}$ for the EKF approximation and therefore $\beta_i^2 = 0$, so the EKF approximation to NKO becomes Newton's method. In this case, substituting (20)-(22) into (9), we get that the updated covariance matrix is a zero matrix. This implies that the EKF approximation is not good enough for the covariance matrix if there are nonlinearities as we know that we have not found the optimal point with total certainty. The EKF could be applied anyway by selecting β_i^2 in a different manner, not following the fundamental procedure suggested in Section III-B1. However, the EKF approximation to the KF moments is usually poor compared to sigma point methods [2], [5]. Therefore, Newton's method can be seen as NKO in which the KF moments are approximated poorly and the noise parameter β_i^2 is selected suboptimally. A better approximation to the KF moments (10)-(12) and a proper selection of β_i^2 is expected to outperform EKF and, therefore, Newton's method. Such an approximation is explained in the following section.

C. A practical NKO algorithm

In general, the KF moments (10)-(12) do not have an analytical expression so they must be approximated. There are a vast number of algorithms that approximate the moments (10)-(12), e.g., the EKF, UKF, CKF or QKF. The UKF and CKF are usually preferred over the QKF because the number of sigma-points increases linearly with the dimension of the state while in the QKF it increases exponentially. However, their accuracy is lower than for QKF. While this might be unimportant in some cases, the UKF and CKF do not work well to perform NKO in the Gaussian MAP nonlinear filtering examples of Section VII.

It is also important to notice that the sequence of covariance matrices $\mathbf{P}_i$ gets smaller and smaller. Therefore, stability issues can arise and the updated covariance matrix might not be positive definite so a square root solution should be used [21]. As a result, we propagate a sequence of matrices $\mathbf{P}_i^{1/2}$ where $\mathbf{P}_i = \mathbf{P}_i^{1/2}\left(\mathbf{P}_i^{1/2}\right)^T$.

For the reasons mentioned above, we suggest using the square root QKF [21]. How to calculate $\overline{\mathbf{x}}_{i+1}$ and $\mathbf{P}_{i+1}^{1/2}$ based on $\overline{\mathbf{x}}_i$ and $\mathbf{P}_i^{1/2}$ is explained in the following. Let $\left\{\left(\boldsymbol{\xi}_j, w_j\right)\right\}_{j=1}^m$ denote the set of quadrature points and associated weights. We build the following matrix and quadrature points

$$\mathcal{X}_{i,j} = \mathbf{P}_i^{1/2}\boldsymbol{\xi}_j + \overline{\mathbf{x}}_i \quad j = 1, ..., m \tag{24}$$

$$\mathcal{Y}_{i,j} = \nabla f\left(\mathcal{X}_{i,j}\right) \quad j = 1, ..., m \tag{25}$$

$$\mathbf{W} = \operatorname{diag}\left(\left[\sqrt{w_1}, ..., \sqrt{w_m}\right]\right)$$

where $\operatorname{diag}(\cdot)$ stands for a diagonal matrix with the given main diagonal. The approximated KF moments are [21]

$$\overline{\mathbf{y}}_i \approx \sum_{j=1}^m w_j \mathcal{Y}_{i,j} \tag{26}$$

$$\mathbf{S}_i^{1/2} \approx \operatorname{Tria}\left(\left[\mathbf{Y}_i,\, \beta_i \mathbf{I}\right]\right) \tag{27}$$

$$\boldsymbol{\Psi}_i \approx \mathbf{X}_i \mathbf{Y}_i^T \tag{28}$$

where

$$\mathbf{X}_i = \left[\mathcal{X}_{i,1} - \overline{\mathbf{x}}_i, ..., \mathcal{X}_{i,m} - \overline{\mathbf{x}}_i\right]\mathbf{W} \tag{29}$$

$$\mathbf{Y}_i = \left[\mathcal{Y}_{i,1} - \overline{\mathbf{y}}_i, ..., \mathcal{Y}_{i,m} - \overline{\mathbf{y}}_i\right]\mathbf{W} \tag{30}$$

and $\operatorname{Tria}(\cdot)$ is a triangularisation algorithm using QR decomposition described in [21]. The square root of $\mathbf{N}_i$ can be written as

$$\mathbf{N}_i^{1/2} \approx \mathbf{Y}_i - \boldsymbol{\Psi}_i^T \mathbf{P}_i^{-1} \mathbf{X}_i \tag{31}$$

and density $p_{i+1}(\cdot)$ is characterised by [21]

$$\overline{\mathbf{x}}_{i+1} = \overline{\mathbf{x}}_i + \mathbf{K}_i\left(\mathbf{y}_i - \overline{\mathbf{y}}_i\right)\big|_{\mathbf{y}_i=0} \tag{32}$$

$$\mathbf{P}_{i+1}^{1/2} = \operatorname{Tria}\left(\left[\mathbf{X}_i - \mathbf{K}_i\mathbf{Y}_i,\, \beta_i \mathbf{K}_i\right]\right) \tag{33}$$

where $\mathbf{K}_i = \boldsymbol{\Psi}_i \mathbf{S}_i^{-1}$. We stop the iteration if the ith PDF is sufficiently concentrated around the optimal point, see Figure 1. In practice, we assess this by comparing the trace of current covariance matrix with a threshold Γ_c. Finally, the steps of the resulting algorithm can be found in Algorithm 1.

Algorithm 1 Steps of the NKO algorithm

```
Input: Initial parameters $\overline{\mathbf{x}}_0$ and $\mathbf{P}_0^{1/2}$.
Output: MAP estimate $\tilde{\mathbf{x}}^\star = \overline{\mathbf{x}}_{i^\star}$.    ▷ $i^\star$ is the number of iterations
  - Set $c = 0$, $i = 0$          ▷ $c = 1$ if convergence has been attained
  while $c = 0$ do
      - Calculate the quadrature points $\mathcal{X}_{i,1}, ..., \mathcal{X}_{i,m}$ using (24).
      - Calculate the transformed quadrature points $\mathcal{Y}_{i,1}, ..., \mathcal{Y}_{i,m}$ using (25).
      - Approximate $\overline{\mathbf{y}}_i$ using (26).
      - Calculate $\mathbf{X}_i$ and $\mathbf{Y}_i$ using (29) and (30).
      - Approximate $\boldsymbol{\Psi}_i$ and $\mathbf{N}_i^{1/2}$ using (28) and (31).
      - Select $\beta_i^2$ using (16).
      - Approximate $\mathbf{S}_i^{1/2}$ using (27).
      - Calculate $\overline{\mathbf{x}}_{i+1}$ and $\mathbf{P}_{i+1}^{1/2}$ using (32) and (33).
      if $\operatorname{tr}(\mathbf{P}_{i+1}) < \Gamma_c$ then
          - Set $c = 1$ and $i^\star = i + 1$.
      end if
      - Set $i = i + 1$.
  end while
```

IV. Levenberg-Marquardt Kalman optimisation

While NKO can handle general optimisation problems, in this section we focus on least squares problems, e.g., MAP estimation in Gaussian MAP filtering, which will be addressed in Section V. The aim is to find the minimum $\tilde{\mathbf{x}}^\star$ of the function

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{r}^T(\mathbf{x})\,\mathbf{r}(\mathbf{x}) \tag{34}$$

where $\mathbf{r}(\cdot) : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_r}$ and $n_r \geq n_x$.

This section is organised in the following way. Section IV-A addresses Bayes' rule optimisation while LMKO is explained in Section IV-B. Our main motivation for studying these least squares problems is to perform MAP estimation in the update step of Bayesian filters as described in Section V. When applied to this problem, LMKO allows for a simplified implementation and we therefore postpone the description of the practical aspects until MAP filtering has been addressed in Section V.

A. Bayes' rule optimisation

The objective is to create a sequence of PDFs that converges to the optimal point, see Section II. To do so, we create the sequence of measurements

$$\mathbf{y}_i = \mathbf{r}\left(\tilde{\mathbf{x}}\right) + \boldsymbol{\eta}_i \tag{35}$$

where $\boldsymbol{\eta}_i$ is a sequence of independent zero-mean Gaussian measurement noise with covariance matrix $\alpha_i^2\mathbf{I}$ and $\alpha_i^2 > 0$.

We use the measurement equation (35) for $\mathbf{y}_i = \mathbf{0}$ so $l_i^{\mathbf{0}}\left(\tilde{\mathbf{x}}\right) \propto \mathcal{N}\left(\mathbf{0}; \mathbf{r}\left(\tilde{\mathbf{x}}\right), \alpha_i^2\mathbf{I}\right)$. The ith PDF of the BO sequence (3) becomes

$$\begin{aligned}\pi_i\left(\tilde{\mathbf{x}}\right) \propto \exp\Bigg(&-1/2\mathbf{r}^T\left(\tilde{\mathbf{x}}\right)\mathbf{r}\left(\tilde{\mathbf{x}}\right)\sum_{j=0}^{i-1}\frac{1}{\alpha_j^2} \\ &-1/2\left(\tilde{\mathbf{x}} - \overline{\mathbf{x}}_0\right)^T\mathbf{P}_0^{-1}\left(\tilde{\mathbf{x}} - \overline{\mathbf{x}}_0\right)\Bigg).\end{aligned} \tag{36}$$

As pointed out in Section II, we need to prove that this sequence of PDFs is progressively more concentrated around the minimum such that we can use it for optimisation purposes. Using the same arguments as in Section III-A, as i increases and assuming that α_j is bounded, (36) can be approximated as Gaussian with first two moments

$$\begin{aligned}\tilde{\mathbf{x}}^\star =& \arg\min_{\tilde{\mathbf{x}}} f\left(\tilde{\mathbf{x}}\right) \\ \mathbf{P}_{\pi,i}^{-1} =& \left[\nabla\mathbf{r}^T\left(\tilde{\mathbf{x}}\right)\left(\nabla\mathbf{r}^T\left(\tilde{\mathbf{x}}\right)\right)^T + \nabla^2\mathbf{r}\left(\tilde{\mathbf{x}}\right)\mathbf{r}\left(\tilde{\mathbf{x}}\right)\right]\Big|_{\tilde{\mathbf{x}}^\star}\sum_{j=0}^{i-1}\frac{1}{\alpha_j^2}.\end{aligned}$$

If $\sum_{j=0}^{i-1}\frac{1}{\alpha_j^2}$ tends to infinity, $\mathbf{P}_{\pi,i} \to \mathbf{0}$, so $\pi_i\left(\cdot\right)$ tends to a Dirac delta located on $\tilde{\mathbf{x}}^\star$ as in (4). It should be noted that convergence can be attained in one update step by letting $\alpha_0^2 \to 0$.

B. Kalman optimisation

In KO, given $p_i\left(\cdot\right)$ with moments $\overline{\mathbf{x}}_i$ and $\mathbf{P}_i$, we approximate $p_{i+1}\left(\cdot\right)$ using the KF update with measurement (35). In this case, the KF moments are

$$\overline{\mathbf{y}}_i = \int \mathbf{r}\left(\tilde{\mathbf{x}}\right)p_i\left(\tilde{\mathbf{x}}\right)d\tilde{\mathbf{x}} \tag{37}$$

$$\boldsymbol{\Phi}_i = \int \left(\mathbf{r}\left(\tilde{\mathbf{x}}\right) - \overline{\mathbf{y}}_i\right)\left(\mathbf{r}\left(\tilde{\mathbf{x}}\right) - \overline{\mathbf{y}}_i\right)^T p_i\left(\tilde{\mathbf{x}}\right)d\tilde{\mathbf{x}} \tag{38}$$

$$\boldsymbol{\Psi}_i = \int \left(\tilde{\mathbf{x}} - \overline{\mathbf{x}}_i\right)\left(\mathbf{r}\left(\tilde{\mathbf{x}}\right) - \overline{\mathbf{y}}_i\right)^T p_i\left(\tilde{\mathbf{x}}\right)d\tilde{\mathbf{x}}. \tag{39}$$

and $\mathbf{S}_i = \boldsymbol{\Phi}_i + \alpha_i^2\mathbf{I}$. The recursion that comes from substituting (37)-(39) into (8)-(9) is referred to as LMKO for reasons that should be clear at the end of this section.

1) On the selection of α_i: In principle, α_i should be selected as β_i, see Section III-B1. That is, $\alpha_i^2 = \Gamma_\alpha \max\left(\lambda_1, ..., \lambda_{n_r}\right)$ where $\lambda_1, ..., \lambda_{n_r}$ correspond to the eigenvalues of the expected squared nonlinearities $\mathbf{N}_i$ of $\mathbf{r}\left(\cdot\right)$. The higher Γ_α, the higher accuracy is expected but convergence is slower. For LMKO, it is quite convenient to select $\Gamma_\alpha \gg 1$. Equations simplify and the connection with LM method is clearly established as will be seen in Section IV-B2. If we select $\Gamma_\alpha \gg 1$, we get

$$\begin{aligned}\alpha_i^2\mathbf{I} + \mathbf{N}_i =& \alpha_i^2\mathbf{Q}_i\mathbf{Q}_i^{-1} + \mathbf{Q}_i\boldsymbol{\Lambda}_i\mathbf{Q}_i^{-1} \\ =& \mathbf{Q}_i\left(\alpha_i^2\mathbf{I} + \boldsymbol{\Lambda}_i\right)\mathbf{Q}_i^{-1} \\ \approx& \alpha_i^2\mathbf{Q}_i\mathbf{I}\mathbf{Q}_i^{-1} = \alpha_i^2\mathbf{I}\end{aligned} \tag{40}$$

where $\mathbf{N}_i = \mathbf{Q}_i\boldsymbol{\Lambda}_i\mathbf{Q}_i^{-1}$is the eigendecomposition of $\mathbf{N}_i$ [3].

2) Relation to LM method: As done in Section III-B2 with $\nabla f\left(\cdot\right)$, we decompose $\mathbf{r}\left(\cdot\right)$ using statistical linear regression w.r.t. $p_i\left(\cdot\right)$ [5]

$$\mathbf{r}\left(\mathbf{x}\right) = \mathbf{A}_i\mathbf{x} + \mathbf{b}_i + \mathbf{n}_i\left(\mathbf{x}\right)$$

where $\mathbf{A}_i = \boldsymbol{\Psi}_i^T\mathbf{P}_i^{-1}$, $\mathbf{b}_i = \overline{\mathbf{y}}_i - \mathbf{A}_i\overline{\mathbf{x}}_i$ and $\mathbf{n}_i\left(\cdot\right)$ represents the nonlinearities of $\mathbf{r}\left(\cdot\right)$. It should be noted that, as in (18), $\mathbf{A}_i = \mathsf{E}_{p_i}\left[\mathbf{J}\left(\tilde{\mathbf{x}}\right)\right]$ where $\mathbf{J}\left(\tilde{\mathbf{x}}\right) = \left[\nabla\left(\mathbf{r}\left(\tilde{\mathbf{x}}\right)\right)^T\right]^T$ is the Jacobian of $\mathbf{r}\left(\tilde{\mathbf{x}}\right)$.

We can write the LMKO recursion as

$$\overline{\mathbf{x}}_{i+1} = \overline{\mathbf{x}}_i - \mathbf{P}_i\mathbf{A}_i^T\left(\mathbf{A}_i\mathbf{P}_i\mathbf{A}_i^T + \mathbf{N}_i + \alpha_i^2\mathbf{I}\right)^{-1}\overline{\mathbf{y}}_i$$

Applying the matrix inversion lemma, we get

$$\begin{aligned}\overline{\mathbf{x}}_{i+1} =& \overline{\mathbf{x}}_i - \left(\mathbf{P}_i^{-1} + \mathbf{A}_i^T\left(\mathbf{N}_i + \alpha_i^2\mathbf{I}\right)^{-1}\mathbf{A}_i\right)^{-1} \\ & \mathbf{A}_i^T\left(\mathbf{N}_i + \alpha_i^2\mathbf{I}\right)^{-1}\overline{\mathbf{y}}_i.\end{aligned} \tag{41}$$

If α_i^2 is chosen as indicated in Section IV-B1, approximation (40) is accurate. Therefore, (41) can be approximated as

$$\begin{aligned}\overline{\mathbf{x}}_{i+1} =& \overline{\mathbf{x}}_i - \left(\alpha_i^2\mathbf{P}_i^{-1} + \mathbf{A}_i^T\mathbf{A}_i\right)^{-1}\mathbf{A}_i^T\overline{\mathbf{y}}_i \\ =& \overline{\mathbf{x}}_i - \left(\alpha_i^2\mathbf{P}_i^{-1} + \mathsf{E}_i\left[\mathbf{J}^T\left(\tilde{\mathbf{x}}\right)\right]\mathsf{E}_i\left[\mathbf{J}\left(\tilde{\mathbf{x}}\right)\right]\right)^{-1} \\ & \mathsf{E}_i\left[\mathbf{J}^T\left(\tilde{\mathbf{x}}\right)\right]\mathsf{E}_i\left[\mathbf{r}\left(\tilde{\mathbf{x}}\right)\right].\end{aligned} \tag{42}$$

Recursion (42) boils down to the LM algorithm if we replace $\mathbf{P}_i$ by $\mathbf{I}$ and the expected values by the values evaluated at $\overline{\mathbf{x}}_i$. Thus, we can see (42) as a generalisation of the LM algorithm that takes into account the prior knowledge, which is given by $\mathbf{P}_i$.

3) EKF approximation: The EKF approximation to the moments (37)-(39) is

$$\overline{\mathbf{y}}_i = \mathbf{r}\left(\overline{\mathbf{x}}_i\right) \tag{43}$$

$$\mathbf{S}_i = \mathbf{J}_i\mathbf{P}_i\mathbf{J}_i^T + \alpha_i^2\mathbf{I} \tag{44}$$

$$\boldsymbol{\Psi}_i = \mathbf{P}_i\mathbf{J}_i^T \tag{45}$$

where $\mathbf{J}_i = \mathbf{J}\left(\overline{\mathbf{x}}_i\right)$. Substituting (43)-(45) into (8), and applying the matrix inversion lemma, we get

$$\overline{\mathbf{x}}_{i+1} = \overline{\mathbf{x}}_i - \left(\alpha_i^2\mathbf{P}_i^{-1} + \mathbf{J}_i^T\mathbf{J}_i\right)^{-1}\mathbf{J}_i^T\mathbf{r}\left(\overline{\mathbf{x}}_i\right). \tag{46}$$

If $\mathbf{P}_i = \mathbf{I}$, (46) is the LM method recursion. In general, (46) can be thought of as a generalisation of LM method that accounts for prior knowledge. If $\alpha_i^2 \to 0$, then, the recursion becomes

$$\overline{\mathbf{x}}_{i+1} = \overline{\mathbf{x}}_i - \left(\mathbf{J}_i^T\mathbf{J}_i\right)^{-1}\mathbf{J}_i^T\mathbf{r}\left(\overline{\mathbf{x}}_i\right) \tag{47}$$

which is the Gauss-Newton recursion. It can be checked that for $\alpha_i^2 \to 0$, the updated covariance matrix is a zero matrix. Recall that this also happened for the EKF approximation to NKO, see Section III-B3. LM and Gauss-Newton methods can converge to a local minimum instead of the global minimum. Because of these relations, LMKO can also converge to a local minimum instead of the global minimum.

V. Application to Gaussian MAP filtering update

In this section, we apply NKO and LMKO to the update step of Bayesian filtering using a Gaussian MAP approximation. The aim is therefore to approximate the posterior PDF given a measurement equation and a prior PDF using the MAP estimate. The resulting filtering algorithms are referred to as NKO filter (NKOF) and LMKO filter (LMKOF). As we only consider the update step, we do not include the time index of the filtering recursion to simplify notation[3]. As mentioned in the introduction, the prediction step is out of the scope of the paper. In practice, it can be done using the UKF prediction step [2] so NKOF and LMKOF can be applied with linear/nonlinear dynamics.

We write the state vector as $\mathbf{x} = \left[\mathbf{a}^T, \, \mathbf{b}^T\right]^T$, where $\mathbf{a} \in \mathbb{R}^{n_a}$, $\mathbf{b} \in \mathbb{R}^{n_b}$ and $n_x = n_a + n_b$, such that the measurement equation is

$$\mathbf{z} = \mathbf{h}(\mathbf{a}) + \boldsymbol{\xi} \tag{48}$$

where $\mathbf{h}(\cdot)$ is the measurement function and $\boldsymbol{\xi}$ is the zero-mean Gaussian measurement noise with covariance matrix $\mathbf{R}$. Equation (48) assumes additive Gaussian noise for simplicity. Nevertheless, NKO and LMKO can be in principle generalised for other types of measurement equations. We want to highlight that the "artificial" measurements to find the MAP, used in Sections II, III and IV, are represented by $\mathbf{y}_i$ while the actual measurement of the update step is $\mathbf{z}$.

The prior $p(\cdot)$ is Gaussian with mean $\overline{\mathbf{x}}$ and covariance matrix $\boldsymbol{\Sigma}$:

$$\overline{\mathbf{x}} = \left[(\overline{\mathbf{a}})^T, (\overline{\mathbf{b}})^T\right]^T \tag{49}$$

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_a & \boldsymbol{\Sigma}_{ab} \\ \boldsymbol{\Sigma}_{ab}^T & \boldsymbol{\Sigma}_b \end{bmatrix}. \tag{50}$$

The posterior $q(\cdot)$ is given by Bayes' rule

$$\begin{aligned} q(\mathbf{x}) =& \frac{l(\mathbf{z}\,|\mathbf{a})\, p(\mathbf{x})}{\int l(\mathbf{z}\,|\mathbf{a})\, p(\mathbf{x})\, d\mathbf{x}} \\ =& q_a(\mathbf{a})\, p(\mathbf{b}\,|\mathbf{a}) \end{aligned}$$

where $l(\mathbf{z}\,|\mathbf{a}) = \mathcal{N}(\mathbf{z}; \mathbf{h}(\mathbf{a}), \mathbf{R})$ is the likelihood, which is obtained using (48), and the posterior $q_a(\cdot)$ of $\mathbf{a}$ is

$$q_a(\mathbf{a}) = \frac{l(\mathbf{z}\,|\mathbf{a})\, p(\mathbf{a})}{\int l(\mathbf{z}\,|\mathbf{a})\, p(\mathbf{a})\, d\mathbf{a}}.$$

The aim is to approximate the posterior $q(\cdot)$ as a Gaussian. To do this, we approximate $q_a(\cdot)$ as a Gaussian

$$q_a(\mathbf{a}) \approx \mathcal{N}\left(\mathbf{a}; \tilde{\mathbf{a}}^\star, \tilde{\mathbf{P}}_a\right) \tag{51}$$

where $\tilde{\mathbf{a}}^\star$ is the MAP estimate of $q_a(\cdot)$ and $\tilde{\mathbf{P}}_a$ is the posterior covariance matrix approximation. The aim of the previous decomposition of the state vector is to lower the dimensionality of the optimisation problem and, therefore, the computational burden.

We want to highlight that $\tilde{\mathbf{a}}^\star$ corresponds to the final mean of a KO sequence but $\tilde{\mathbf{P}}_a$ is not the final covariance of a KO sequence. The final covariance of a KO sequence always tends to zero, see Figure 1, which means that the sequence of PDFs of KO gets more concentrated around the MAP estimate. Matrix $\tilde{\mathbf{P}}_a$ is to be determined in Section V-B. Once $q_a(\cdot)$ is approximated, we calculate $q(\cdot)$ as indicated in Section V-C.

A. Calculation of the MAP estimate

For the measurement equation given by (48), the MAP estimate of $q_a(\cdot)$ can be found by minimising

$$\begin{aligned} f(\mathbf{a}) =& \frac{1}{2}(\mathbf{z} - \mathbf{h}(\mathbf{a}))^T \mathbf{R}^{-1} (\mathbf{z} - \mathbf{h}(\mathbf{a})) \\ &+ \frac{1}{2}(\mathbf{a} - \overline{\mathbf{a}})^T \boldsymbol{\Sigma}_a^{-1} (\mathbf{a} - \overline{\mathbf{a}}). \end{aligned} \tag{52}$$

The initial PDF we use in KO is

$$p_0(\tilde{\mathbf{a}}) = \mathcal{N}(\tilde{\mathbf{a}}; \overline{\mathbf{a}}_0, \mathbf{P}_0) \tag{53}$$

where $\overline{\mathbf{a}}_0 = \overline{\mathbf{a}}$ and $\mathbf{P}_0 = \boldsymbol{\Sigma}_a$. We use this PDF as we know that the true value of $\mathbf{a}$ is in the region where (53) is high so it makes sense to use this knowledge and look for the MAP estimate $\tilde{\mathbf{a}}^\star$ also in that region.

1) NKO: The application of NKO to MAP filtering directly uses the algorithm explained in Section III-C. The gradient, which we need in NKO as indicated by (6), is

$$\nabla f(\mathbf{a}) = -\mathbf{H}(\mathbf{a})^T \mathbf{R}^{-1} (\mathbf{z} - \mathbf{h}(\mathbf{a})) + \boldsymbol{\Sigma}_a^{-1}(\mathbf{a} - \overline{\mathbf{a}})$$

where $\mathbf{H}(\mathbf{a})$ is the Jacobian of $\mathbf{h}(\cdot)$ evaluated at $\mathbf{a}$.

2) LMKO: The application of LMKO to MAP filtering requires the writing of $f(\cdot)$ as in (34). We get that,

$$\mathbf{r}(\mathbf{a}) = \mathbf{T}\left[(\mathbf{z} - \mathbf{h}(\mathbf{a}))^T, (\overline{\mathbf{a}} - \mathbf{a})^T\right]^T \tag{54}$$

where $\mathbf{T}$ is such that

$$\mathbf{T}\mathbf{T}^T = \begin{pmatrix} \mathbf{R}^{-1} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_a^{-1} \end{pmatrix}$$

so $\mathbf{T}$ can be obtain using Cholesky factorisation. We can apply LMKO directly as explained in Section IV although, for this problem, the recursion can be simplified. Instead of computing the moments (37)-(39), we use the following moments

$$\overline{\mathbf{z}}_i = \mathsf{E}_i[\mathbf{h}(\tilde{\mathbf{a}})] \tag{55}$$

$$\boldsymbol{\Phi}_i^{\mathbf{z}} = \mathsf{cov}_i[\mathbf{h}(\tilde{\mathbf{a}})] \tag{56}$$

$$\boldsymbol{\Psi}_i^{\mathbf{z}} = \mathsf{cov}_i[\tilde{\mathbf{a}}, \, \mathbf{h}(\tilde{\mathbf{a}})]. \tag{57}$$

These are the usual KF moments, have a lower dimensionality and can be easily approximated by the UT [2]. Performing a similar analysis as in Section III-B2, the expected Jacobian of $\mathbf{h}(\cdot)$ and the expected squared nonlinearities of $\mathbf{h}(\cdot)$ are

$$\mathbf{H}_i = \mathsf{E}_i\left[\left(\nabla \mathbf{h}^T(\tilde{\mathbf{a}})\right)^T\right] = \left(\boldsymbol{\Psi}_i^{\mathbf{z}}\right)^T \mathbf{P}_i \tag{58}$$

$$\mathbf{N}_i^{\mathbf{z}} = \boldsymbol{\Phi}_i^{\mathbf{z}} - \left(\boldsymbol{\Psi}_i^{\mathbf{z}}\right)^T \mathbf{P}_i^{-1} \boldsymbol{\Psi}_i^{\mathbf{z}}. \tag{59}$$

As indicated in Section IV-B1, LMKO selects α_i^2 based on the expected squared nonlinearites of $\mathbf{r}(\cdot)$. Using (54), we get

$$\mathbf{N}_i = \mathbf{T}\begin{bmatrix} \mathbf{N}_i^{\mathbf{z}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \mathbf{T}^T. \tag{60}$$

[3] It should be noted that the measurement function and measurement noise can depend on the current time step although it is not explicit in the notation.

It is shown in Appendix A that the LMKO recursion when α_i^2 is chosen as indicated in Section IV-B1, which is given by (42), can be written as

$$\begin{aligned}\overline{\mathbf{a}}_{i+1} =& \overline{\mathbf{a}} + \mathbf{L}_i \left(\mathbf{z} - \overline{\mathbf{z}}_i - \mathbf{H}_i \left(\overline{\mathbf{a}} - \overline{\mathbf{a}}_i\right)\right) \\ & - \alpha_i^2 \left(\mathbf{I} - \mathbf{L}_i \mathbf{H}_i\right) \hat{\mathbf{P}}_i \mathbf{P}_i^{-1} \left(\overline{\mathbf{a}} - \overline{\mathbf{a}}_i\right) \qquad (61)\\ \mathbf{P}_{i+1} =& \mathbf{P}_i - \mathbf{P}_i \left(\alpha_i^2 \check{\mathbf{P}}_i + \mathbf{P}_i\right)^{-1} \mathbf{P}_i \qquad (62)\end{aligned}$$

where

$$\begin{aligned}\hat{\mathbf{P}}_i =& \mathbf{P}_0 - \alpha_i^2 \mathbf{P}_0 \left(\mathbf{P}_i + \alpha_i^2 \mathbf{P}_0\right)^{-1} \mathbf{P}_0 \qquad (63)\\ \mathbf{L}_i =& \hat{\mathbf{P}}_i \mathbf{H}_i^T \left(\mathbf{R} + \mathbf{H}_i \hat{\mathbf{P}}_i \mathbf{H}_i^T\right)^{-1} \qquad (64)\\ \check{\mathbf{P}}_i =& \mathbf{P}_0 - \mathbf{P}_0 \mathbf{H}_i^T \left(\mathbf{R} + \mathbf{H}_i \mathbf{P}_0 \mathbf{H}_i^T\right)^{-1} \mathbf{H}_i \mathbf{P}_0. \qquad (65)\end{aligned}$$

As explained in Section III-C, KO algorithms should use a square root solution to improve numerical stability. Provided that the sigma-point weights are positive, the update step of a square root UKF is exactly the same as the square root QKF explained in Section III-C but using different sigma-points and weights. The resulting steps of LMKO for MAP filtering are given in Algorithm 2.

Algorithm 2 Steps of the LMKO algorithm for MAP filtering

```
Input: Initial parameters $\overline{\mathbf{a}}_0$ and $\mathbf{P}_0^{1/2}$.
Output: MAP estimate $\tilde{\mathbf{a}}^\star = \overline{\mathbf{a}}_{i^\star}$.    ▷ $i^\star$ is the number of iterations
  - Set $c = 0$, $i = 0$        ▷ $c = 1$ if convergence has been attained
  while $c = 0$ do
      - Calculate the UKF sigma-points $\mathcal{X}_{i,1}, ..., \mathcal{X}_{i,m}$ according to $p_i(\cdot)$.
      - Calculate the transformed sigma-points $\mathcal{Z}_{i,1}, ..., \mathcal{Z}_{i,m}$ using (48).
      - Approximate $\overline{\mathbf{z}}_i$ as in (26).
      - Create matrices $\mathbf{X}_i$ and $\mathbf{Z}_i$ as in (29) and (30).
      - Approximate $\mathbf{\Psi}_i^{\mathbf{z}}$ and $\left(\mathbf{N}_i^{\mathbf{z}}\right)^{1/2}$ using the square root UKF as in Section III-C.
      - Calculate $\mathbf{N}_i$ using (60) and its positive eigenvalues $\lambda_1, ..., \lambda_{n_z}$.
      - Calculate $\alpha_i^2 = \Gamma_\alpha \max\left(\lambda_1, ..., \lambda_{n_z}\right)$.
      - Calculate $\mathbf{H}_i$ using (58), $\hat{\mathbf{P}}_i$ using (63) and $\mathbf{L}_i$ using (64).
      - Calculate $\overline{\mathbf{a}}_{i+1}$ using (61).
      - Calculate $\mathbf{K}_{1,i} = \mathbf{P}_0 \mathbf{H}_i^T \left(\mathbf{R} + \mathbf{H}_i \mathbf{P}_0 \mathbf{H}_i^T\right)^{-1}$.
      - Calculate $\check{\mathbf{P}}_i^{1/2} = \mathrm{Tria}\left(\left[\mathbf{P}_0^{1/2} - \mathbf{K}_{1,i}\mathbf{H}_i\mathbf{P}_0^{1/2}, \mathbf{K}_{1,i}\mathbf{R}^{1/2}\right]\right)$ (see (65)).
      - Calculate $\mathbf{K}_{2,i} = \mathbf{P}_i \left(\alpha_i^2 \check{\mathbf{P}}_i + \mathbf{P}_i\right)^{-1}$.
      - Calculate $\mathbf{P}_{i+1}^{1/2} = \mathrm{Tria}\left(\left[\mathbf{P}_i^{1/2} - \mathbf{K}_{2,i}\mathbf{P}_i^{1/2}, \sqrt{\alpha_i^2}\mathbf{K}_{2,i}\check{\mathbf{P}}_i^{1/2}\right]\right)$ (see (62)).
      if $\mathrm{tr}\left(\mathbf{P}_{i+1}\right) < \Gamma_c$ then
          - Set $c = 1$ and $i^\star = i + 1$.
      end if
      - Set $i = i + 1$.
  end while
```

B. Approximation of the posterior covariance matrix

In this section, we explain how NKO and LMKO approximate the posterior covariance matrix $\tilde{\mathbf{P}}_a$ in (51). We recall that the posterior covariance matrix is not the same as the covariance matrix at the final step of the KO iteration as the latter always tends to zero, as will be proved in Section VI. Gaussian MAP filters usually approximate the covariance matrix of the posterior through the Hessian of $f(\cdot)$ [22] or linearisation of the measurement function evaluated at $\tilde{\mathbf{a}}^\star$ [9].

1) NKO: The NKO iteration directly provides an approximation to the expected value of the Hessian of $f(\cdot)$. The posterior covariance can be approximated as [22]

$$\begin{aligned}\tilde{\mathbf{P}}_a &= \left(-\mathsf{E}_{i^\star}\left[\nabla^2 \log\left(l\left(\mathbf{z}\left|\tilde{\mathbf{a}}\right.\right) p\left(\tilde{\mathbf{a}}\right)\right)\right]\right)^{-1} \\ &= \left(\mathsf{E}_{i^\star}\left[\nabla^2 f\left(\tilde{\mathbf{a}}\right)\right]\right)^{-1} = \mathbf{P}_{i^\star}\left(\mathbf{\Psi}_{i^\star}^T\right)^{-1} \qquad (66)\end{aligned}$$

where $i^\star$ denotes the step at which the KO iteration stops and we have used (18).

If NKO is initialised properly, it converges to a minimum but, otherwise, it can converge to a maximum or inflection point of (52). Nevertheless, if $\tilde{\mathbf{P}}_a$ is not positive definite, NKO has not converged to a minimum of $f(\cdot)$ because of the relation between $\tilde{\mathbf{P}}_a^{-1}$ and the Hessian [23]. This is quite useful because the algorithm itself indicates if it has been successful. If NKO does not locate the MAP estimate, we can repeat KO with different initial conditions or apply a usual KF update at this step. We can do the latter as the ultimate objective is to approximate the posterior. In Section VII, we use the UKF for this purpose although NKO hardly ever fails to find the MAP estimate.

2) LMKO: If we use the Jacobian evaluated at $\overline{\mathbf{a}}_i$ in (65) instead of the expected Jacobian $\mathbf{H}_i$, matrix $\check{\mathbf{P}}_{i^\star}$, where $i^\star$ denotes the final step of the iteration, is the posterior covariance matrix of the IEKF [9]. When the iteration converges, the expected Jacobian is quite similar to the Jacobian at the mean of the PDF. Thus, we can use $\tilde{\mathbf{P}}_a = \check{\mathbf{P}}_{i^\star}$.

C. Approximation of the posterior

If the state is partitioned as in (49), (50) and the measurement is of the form (48), the posterior PDF approximation can be written as [24]

$$q\left(\mathbf{x}\right) \approx \mathcal{N}\left(\mathbf{x}; \tilde{\mathbf{x}}^\star, \tilde{\mathbf{P}}\right) \qquad (67)$$

where $\tilde{\mathbf{x}}^\star = \left[\left(\tilde{\mathbf{a}}^\star\right)^T, \left(\tilde{\mathbf{b}}^\star\right)^T\right]^T$,

$$\tilde{\mathbf{P}} = \left[\begin{array}{cc} \tilde{\mathbf{P}}_a & \tilde{\mathbf{P}}_{ab} \\ \tilde{\mathbf{P}}_{ab}^T & \tilde{\mathbf{P}}_b \end{array}\right],$$

and

$$\begin{aligned}\tilde{\mathbf{b}}^\star &= \overline{\mathbf{b}} + \mathbf{\Sigma}_{ab}^T \mathbf{\Sigma}_a^{-1} \left(\tilde{\mathbf{a}} - \overline{\mathbf{a}}\right) \qquad (68)\\ \tilde{\mathbf{P}}_{ab} &= \tilde{\mathbf{P}}_a \mathbf{\Sigma}_a^{-1} \mathbf{\Sigma}_{ab} \qquad (69)\\ \tilde{\mathbf{P}}_b &= \mathbf{\Sigma}_b - \mathbf{\Sigma}_{ab}^T \mathbf{\Sigma}_a^{-1} \left(\mathbf{\Sigma}_a - \tilde{\mathbf{P}}_a\right) \mathbf{\Sigma}_a^{-1} \mathbf{\Sigma}_{ab}. \qquad (70)\end{aligned}$$

To sum up, the steps of the update phase of MAP Bayesian filtering using NKOF and LMKOF are given in Algorithm 3.

VI. Local convergence of the algorithms

This section analyses the local convergence of NKO and LMKO as $i \to \infty$. It is shown that NKO and LMKO asymptotically behave as Newton´s and Gauss-Newton recursion, respectively. This implies that these algorithms have the same properties of local convergence as Newton´s and Gauss-Newton methods [8] although globally they are expected to behave better due to the reasons mentioned previously, see for example Section III-B3.

Algorithm 3 Update step of LMKOF and NKOF

```
Input: Prior moments $\overline{\mathbf{x}}$ and $\mathbf{\Sigma}$.
Output: Posterior moments $\tilde{\mathbf{x}}^\star$ and $\tilde{\mathbf{P}}$.
- Apply LMKO/NKO to calculate the MAP estimate $\tilde{\mathbf{a}}^\star$ of $q_a(\cdot)$ with a prior given by (53).
if LMKOF then
    - Calculate $\tilde{\mathbf{P}}_a = \check{\mathbf{P}}_{i^\star}$ of $q_a(\cdot)$ using (65) where $i^\star$ is the number of iterations.
else
    - Calculate $\tilde{\mathbf{P}}_a = \mathbf{P}_{i^\star}\left(\mathbf{\Psi}_{i^\star}^T\right)^{-1}$ where $i^\star$ is the number of iterations.
end if
- Compute $\tilde{\mathbf{x}}^\star$ and $\tilde{\mathbf{P}}$ using (68), (69) and (70).
```

A. NKO

In order to analyse the convergence, we focus on the NKO recursion given by (19) and (9). Under Assumption A, it is shown in Appendix B that

$$\lim_{i\to\infty} \mathbf{P}_i = \mathbf{0} \tag{71}$$

Due to (71) and the fact that $\lim_{i\to\infty}\mathcal{N}\left(\mathbf{x};\overline{\mathbf{x}},\mathbf{P}_i\right) = \delta\left(\mathbf{x}-\overline{\mathbf{x}}\right)$ [25], for i sufficiently high

$$\mathsf{E}_i\left[\mathbf{g}\left(\mathbf{x}\right)\right] = \mathbf{g}\left(\overline{\mathbf{x}}_i\right) \tag{72}$$

where $\mathbf{g}(\cdot)$ is a function of the state. For i sufficiently high, we use (72) and obtain $\mathsf{E}_i\left[\nabla f\left(\tilde{\mathbf{x}}\right)\right] = \nabla f\left(\overline{\mathbf{x}}_i\right)$, $\mathsf{E}_i\left[\nabla^2 f\left(\tilde{\mathbf{x}}\right)\right] = \nabla^2 f\left(\overline{\mathbf{x}}_i\right)$ in (19). It can be easily checked that we also get $\mathbf{N}_i = \mathbf{0}$ in this case and therefore $\lim_{i\to\infty}\beta_i = 0$. As a result, if i is sufficiently high, the NKO recursion (19) behaves as Newton´s method so both methods have the same local convergence properties, such as rate of convergence [8].

B. LMKO

In order to analyse the convergence, we focus on the LMKO recursion applied to MAP filtering, which is given by (61) and (62). It is shown in Appendix C that

$$\lim_{i\to\infty} \mathbf{P}_i = \mathbf{0}. \tag{73}$$

Now, we calculate recursion (61) for i high enough. Because of the way we select α_i, (73) and the fact that in the limit $\mathbf{N}_i = \mathbf{0}$, $\lim_{i\to\infty}\alpha_i = 0$ and $\lim_{i\to\infty}\hat{\mathbf{P}}_i = \mathbf{P}_0$. For i sufficiently high, we use (72) and get $\mathsf{E}_i\left[\left(\nabla\mathbf{h}^T\left(\tilde{\mathbf{a}}\right)\right)^T\right] = \left(\nabla\mathbf{h}^T\left(\overline{\mathbf{a}}_i\right)\right)^T$. Therefore, the recursion (61) for i high enough is equivalent to the IEKF recursion, which is the Gauss-Newton method [9].

VII. Simulations

In this section, the performances of NKOF and LMKOF are assessed in two examples. We are especially interested in the case of low measurement noise as the posterior tends to be Gaussian centered at the MAP estimate [7]. As a result, the more accurately an algorithm finds the MAP estimate and approximates the covariance matrix of the posterior, the more accurate the posterior approximation is.

We have implemented KF-type (LMMSE-type) algorithms such as the EKF, UKF and CKF; iterated KF-type (MAP-type) algorithms such as the IEKF, (ad-hoc) IUKF [14] and iterated Levenberg–Marquardt KF[4] (ILMKF) [10]. We use the following parameters for the algorithms. The UKF uses $N_s = 2n_x+1$ sigma-points and the weight of the sigma-point located on the mean is 1/3. The IEKF uses 30 iterations. The (ad-hoc) IUKF in [14] is implemented with $\eta = 0.8$. The ILMKF uses the LM method to obtain the MAP estimate and its parameter is selected as in [26, Chap. 15]. The parameters of NKOF are: three quadrature points per dimension and $\Gamma_\beta = 0.01$. In LMKOF, $\Gamma_\alpha = 10$. Both filters use $\Gamma_c = 10^{-4}$.

The main properties of the iterated algorithms used in the paper are summarised in Table I. The entry “Behaves well with increasing dimension” indicates that the computational burden does not grow exponentially with the dimension of the state. The NKOF does not behave well as it uses a quadrature rule. The root mean square (RMS) error is obtained using 200 Monte Carlo runs, averaging over different realisations of the measurements and the target states.

A. Passive target tracking

This example deals with passive target tracking with bearing, bearing rate and Doppler rate measurements [14]. The state vector at time k is $\mathbf{x}^k = \left[x^k, \dot{x}^k, y^k, \dot{y}^k\right]^T$ where $\left[x^k, y^k\right]^T$ is the position vector and $\left[\dot{x}^k, \dot{y}^k\right]^T$ is the velocity vector. The process equation is

$$\mathbf{x}^{k+1} = \mathbf{F}\mathbf{x}^k + \mathbf{G}\mathbf{v}^k \tag{74}$$

where

$$\mathbf{F} = \mathbf{I}_2 \otimes \begin{pmatrix} 1 & \tau \\ 0 & 1 \end{pmatrix}$$

$$\mathbf{G} = \mathbf{I}_2 \otimes \begin{pmatrix} 0.5\tau^2 \\ \tau \end{pmatrix}$$

where $\mathbf{I}_m$ is the $m \times m$ identity matrix, $\otimes$ is the Kronecker product, τ is the sampling time, $\mathbf{v}^k$ is the process noise that is zero-mean Gaussian distributed with covariance matrix $\mathbf{Q}$. It should be noted that the indices of the iterations of the optimisation algorithms in Sections II, III and IV use subscripts while the time index of the state vector is a superscript. We recall that the optimisation algorithms are performed at every update step to calculate the MAP estimate and then the posterior is approximated as indicated in Section V-C.

The measurement vector $\mathbf{z}^k = \left[\theta^k, \dot{\theta}^k, \dot{f}^k\right]^T$ consist of bearing θ^k, bearing rate $\dot{\theta}^k$, and Doppler rate $\dot{f}^k$, so the measurement equation can be written as

$$\begin{aligned}
\theta^k &= \arctan\left(\frac{y^k}{x^k}\right) + \xi_\theta \\
\dot{\theta}^k &= \frac{\dot{y}^k x^k - \dot{x}^k y^k}{\left(r^k\right)^2} + \xi_{\dot{\theta}} \\
\dot{f}^k &= \frac{-\left(\dot{y}^k x^k - \dot{x}^k y^k\right)^2}{\left(r^k\right)^3 \lambda} + \xi_{\dot{f}}
\end{aligned}$$

[4] In [10], the ILMKF is called “the new iterated filter”.

Table I: Properties of the iterated filtering algorithms

Property	NKOF	LMKOF	IEKF	ILMKF	IUKF
Theoretically sound	√	√	√	√	×
Uses sigma-points	√	√	×	×	√
Derivative-free	×	√	×	×	√
Behaves well with increasing dimension	×	√	√	√	√

where λ is the wavelength, $r^k = \sqrt{(x^k)^2 + (y^k)^2}$ and ξ_θ, $\xi_{\dot{\theta}}$ and $\xi_{\dot{f}}$ are independent zero-mean Gaussian distributed noises with variances σ_θ^2, $\sigma_{\dot{\theta}}^2$ and $\sigma_{\dot{f}}^2$. With this measurement function, the partitioning in Section V is $\mathbf{a}^k = \mathbf{x}^k$ and $\mathbf{b}^k = \oslash$.

The prior PDF at time 0 is

$$p\left(\mathbf{x}^0\right) = \mathcal{N}\left(\mathbf{x}^0; \overline{\mathbf{x}}^0, \mathbf{\Sigma}^0\right). \qquad (75)$$

We consider 120 time steps, $\tau = 0.5\text{s}$, $\mathbf{Q} = \operatorname{diag}\left(\left[\left(9\text{m/s}^2\right)^2, \left(2\text{m/s}^2\right)^2\right]\right)$, $\lambda = 0.1\text{m}$, $\overline{\mathbf{x}}^0 = [800\text{m}, -50\text{m/s}, 300\text{m}, 10\text{m/s}]^T$ and

$$\mathbf{\Sigma}^0 = \begin{bmatrix} \sigma_x^2 & 0 & \rho\sigma_x\sigma_y & 0 \\ 0 & \sigma_v^2 & 0 & 0 \\ \rho\sigma_x\sigma_y & 0 & \sigma_y^2 & 0 \\ 0 & 0 & 0 & \sigma_v^2 \end{bmatrix} \qquad (76)$$

where $\sigma_x = 500\text{m}$, $\sigma_y = 200\text{m}$, $\rho = 0.95$ and $\sigma_v = 100\text{m/s}$.

We analyse three scenarios that differ in the measurement noise variances. Scenario 1: $\sigma_\theta = 2 \cdot 10^{-4}\text{rad}$, $\sigma_{\dot{\theta}} = 10^{-5}\text{rad/s}$ and $\sigma_{\dot{f}} = 0.05\text{Hz/s}$. Scenario 2: $\sigma_\theta = 2 \cdot 10^{-6}\text{rad}$, $\sigma_{\dot{\theta}} = 10^{-5}\text{rad/s}$ and $\sigma_{\dot{f}} = 0.05\text{Hz/s}$. Scenario 3: $\sigma_\theta = 2 \cdot 10^{-6}\text{rad}$, $\sigma_{\dot{\theta}} = 10^{0.1}\text{rad/s}$ and $\sigma_{\dot{f}} = 10^{-3}\text{Hz/s}$.

The RMS position error and the normalised estimation error squared (NEES) of the position in Scenario 1 are plotted in Fig. 2. NKOF, LMKOF and ILMKF have roughly the same performance, outperform the rest of the algorithms and are consistent. IEKF performs well at the beginning but it diverges. The IUKF, UKF and CKF perform better than EKF but are far from the performance of the KO algorithms developed in this paper. EKF has a high NEES and has not not been included in Fig. 2(b).

The execution times in milliseconds of the Matlab implementation of the algorithms on a Intel Core i7 laptop are: NKOF (830), LMKOF (81), IEKF (90), IUKF (1650), ILMKF (45), UKF (15), EKF (10), CKF (15). Lowest computational burden is achieved by LMMSE algorithms: EKF, CKF and UKF. Among the iterated filters, ILMKF is the one with lowest computational burden followed by LMKOF. NKOF requires more computational burden. The much higher execution time of the IUKF comes from the fact that it requires an average of 113.78 iterations to converge while NKOF requires 4.01, LMKOF 3.11 and ILMKF 2.78. In this scenario, NKOF always converges, see Section V-B.

LMKOF and NKOF are not very sensitive to what parameters have been chosen. In LMKOF, we have analysed performance with values of Γ_α between 10 and 1000. Performance is practically unaltered although, the higher Γ_α is, the more iterations are required to converge. For example, for $\Gamma_\alpha = 1000$, the average number of iterations is 5.55 instead of 3.11, which are required for $\Gamma_\alpha = 10$. For NKOF, we have analysed values of Γ_β between 0.01 and 10. Performance remains unchanged but with $\Gamma_\beta = 10$, NKOF performs 17.68 iterations on average rather than 4.01, which are done with $\Gamma_\beta = 0.01$.

In Scenario 2, ILMKF and LMKOF are the algorithms that have better performance followed by NKOF. The NKOF does not converge to a local minimum 5 times out of $120 \cdot 200$ optimisations (number of time steps multiplied by number of Monte Carlo runs) so it converges over 99.99% of the times, see Section V-B. As in Scenario 1, IEKF performs well at the beginning but it diverges soon. IUKF has a high error at the beginning but at the end of the simulation it has the same error as NKOF. Finally in Scenario 3, LMKOF, NKOF and IUKF are the algorithms that behave better while ILMKF and IEKF diverge. The errors are lower in this scenario mainly due to the much lower Doppler rate noise. NKOF does not converge 6 times so it also converges over 99.99% of the times. This scenario is analysed more deeply from an optimisation perspective in Section VII-B. For Scenario 3, we have performed several simulations with $\sigma_{\dot{\theta}} \in (0.01, 10)$ rad/s and the above-mentioned conclusions remain unaltered, i.e., ILMKF diverges while LMKOF and NKOF have the highest performance.

To sum up, in this example, LMKOF is the algorithm with highest performance on the whole followed by NKOF. ILMKF also behaves quite well despite the fact that it diverges in Scenario 3. Importantly, the increase in performance of LMKOF does not imply a much higher computational burden compared to other iterated approaches. In addition, LMKOF does not require any Jacobian calculations, which makes the approach less error prone and more convenient for practitioners than the other iterated filters.

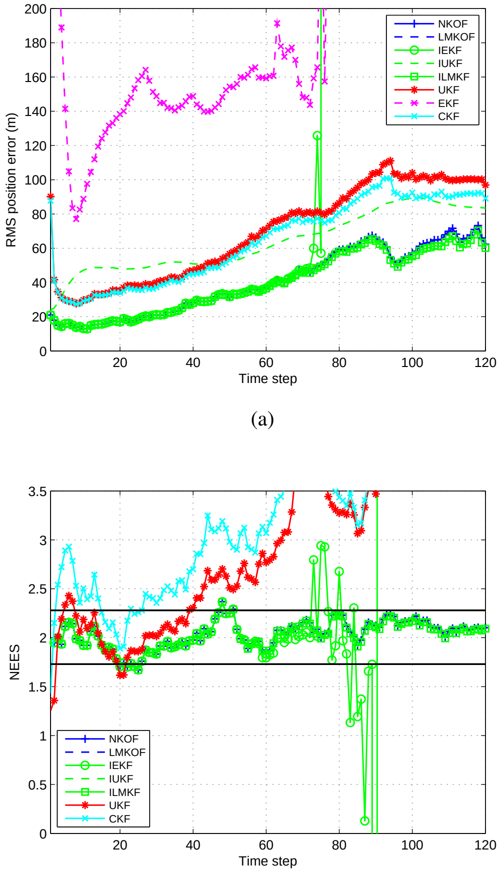
B. *Optimisation problem in passive target tracking*

In Scenario 3 of the passive target tracking example of the previous section, ILMKF and IEKF diverge while LMKOF, NKOF and IUKF perform well. This indicates that at some time step, the KO algorithms proposed in this paper find the MAP estimate accurately while Levenberg-Marquardt and Gauss-Newton methods do not. Therefore, in this section, we analyse Scenario 3 more thoroughly from a pure optimisation perspective. That is, we focus on the update phase at one time step and analyse the behaviour of the MAP filtering algorithms to minimise the objective function, which is given by (52).

The parameters of the objective function we use are

$$\overline{\mathbf{x}}^k = [437.81, -1.42, 2981.95, 8.356]^T$$

$$\mathbf{P}^k = \begin{bmatrix} 2.991 & 5.571 & 11.732 & 3.485 \\ 5.571 & 20.438 & 3.453 & 1.276 \\ 11.732 & 3.453 & 79.855 & 23.95 \\ 3.485 & 1.276 & 23.95 & 9.739 \end{bmatrix}$$



(a)

(b)

Figure 2: RMS position error (a) and NEES (b) against time for Scenario 1. Horizonal black lines indicate the 95% probability region. NKOF, LMKOF, ILMKF provide the lowest errors and are consistent.

Table II: Number of iterations and objective function value of the optimisation algorithms

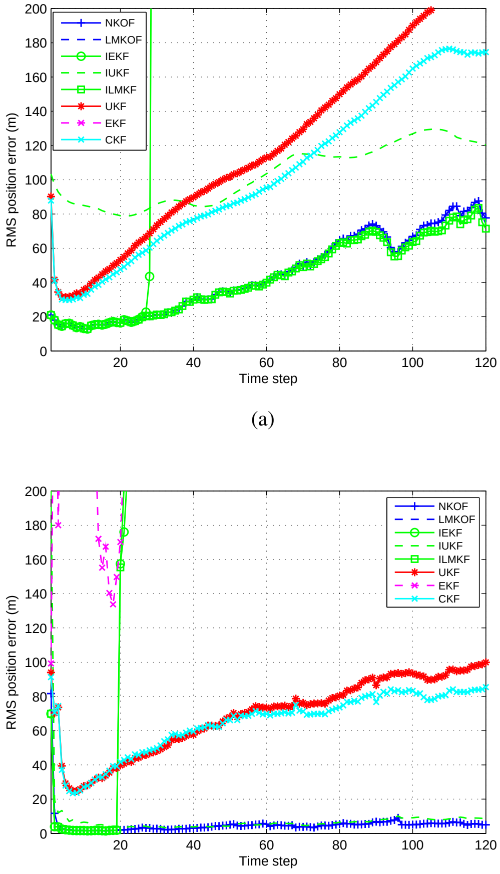
	NKOF	LMKOF	IEKF	ILMKF	IUKF
Iterations	5	8	30	6	105
Function value	8.42	8.42	$5.67 \cdot 10^4$	$5.67 \cdot 10^4$	8.55

$$\mathbf{z}^k = [1.42375, 2.30477, -0.52501]^T$$

where these parameters have been obtained from one Monte Carlo run at $k = 20$. In Table II, we show the objective function value that the optimisation algorithms attain as well as the number of iterations they require to converge. NKOF and LMKOF attain the lowest value among the analysed algorithms. IUKF provides a slightly higher value. IEKF and ILMKF converge to a point that is not the minimum.

C. Angle tracking

In this example, an airbone sensor at a constant height observes a target moving along the ground. The sensor measures the azimuth and elevation of the target. The state vector is the same as in Section VII-A. The target dynamics are modelled with a nearly constant velocity model, i.e., the dynamic equation is (74) with $\mathbf{G} = \mathbf{I}_4$ and $\mathbf{v}^k$ is zero-mean Gaussian distributed with covariance matrix:



(a)

(b)

Figure 3: RMS position error against time (a) Scenario 2 (b) Scenario 3. NKOF and LMKOF outperform a variety of widely used filters. In (a), LMKOF works better as NKOF does not always converge to the MAP. In (b), ILMKF diverges.

$$\mathbf{Q} = \sigma_u^2 \begin{pmatrix} \tau^3/3 & \tau^2/2 \\ \tau^2/2 & \tau \end{pmatrix} \otimes \mathbf{I}_2$$

where σ_u^2 is the continuous-time process noise intensity. The measurement equation is [27]

$$\mathbf{z}^k = \begin{bmatrix} \arctan\left(\left(y^k - y_0^k\right) / \left(x^k - x_0^k\right)\right) \\ \arctan\left(h / \sqrt{\left(x^k - x_0^k\right)^2 + \left(y^k - y_0^k\right)^2}\right) \end{bmatrix} + \boldsymbol{\xi}^k$$

where h is the known height of the sensor, $\left[x_0^k, y_0^k\right]^T$ represents the known position of the sensor at altitude h at time k and $\boldsymbol{\xi}^k$ is a zero-mean Gaussian measurement noise with variance $\mathbf{R} = \operatorname{diag}\left(\left[\sigma_1^2, \sigma_2^2\right]\right)$. With this measurement function, the partitioning in Section V is $\mathbf{a}^k = \left[x^k, y^k\right]^T$ and $\mathbf{b}^k = \left[\dot{x}^k, \dot{y}^k\right]^T$.

The prior PDF at time 0 is given by (75)-(76). The parameters we consider are: $\sigma_x = 1000\text{m}$, $\sigma_y = 2000\text{m}$, $\rho = 0.1$ and $\sigma_v = 2\text{m/s}$, $\sigma_u = 0.01\text{m/s}^{3/2}$, $\overline{\mathbf{x}}^0 = [15000\text{m}, -10\text{m/s}, 5000\text{m}, 10\text{m/s}]^T$, $\tau = 20\text{s}$ and $h = 10000\text{m}$. The number of steps is 48 and the ownship moves

Table III: Noise parameters for each scenario

	1	2	3	4
σ_1 (°)	$4.5 \cdot 10^{-4}$	$4.5 \cdot 10^{-4}$	4.5	4.5
σ_2 (°)	$4.5 \cdot 10^{-3}$	4.5	$4.5 \cdot 10^{-3}$	4.5

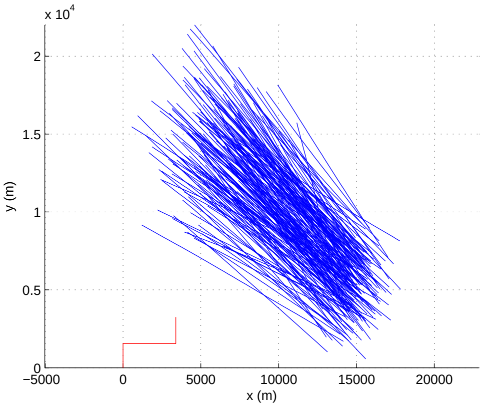


Figure 4: Scenario for the angle tracking example. The red line indicates the ownship trajectory and the blue lines indicate the target trajectory for each Monte Carlo run.

with the same parameters as in [11]. The scenario and the trajectories used to evaluate the filters' performances are shown in Figure 4. We consider four scenarios that differ in their noise parameters, which are provided in Table III. For completeness, we have also included scenarios with relatively high measurement noise to demonstrate the superiority of KF-type algorithms in this case.

The RMS position error of the algorithms averaged over all time steps are provided in Table IV. As expected, in Scenario 1, in which the measurements are rather precise, UKF and CKF have much worse performance than Gaussian MAP filters and NKO, LMKOF and ILMKF have the best performance. In Scenario 2, NKOF, LMKOF and IEKF are superior to the rest. In Scenarios 3 and 4, the UKF and CKF are the algorithms with lowest error. Among the Gaussian MAP filters, LMKOF is the best one and ILMKF does not perform well in Scenario 4. On the whole, LMKOF is the best Gaussian MAP filter followed by NKOF.

VIII. Conclusions

In this paper, we have introduced the KO framework and developed two algorithms within this framework, LMKO and NKO. Both algorithms can be implemented using low-computational complexity sigma-point methods, which makes both of them quite suitable for Gaussian MAP filtering. As demonstrated by numerical simulations, KO algorithms applied to MAP filtering are more robust than the IEKF, ILMKF and IUKF. In addition, for low measurement noise, these new filters should outperform LMMSE-type solutions such as EKF, UKF or CKF. On the whole, LMKO outperforms NKO because it is a derivative-free method, behaves well with increasing dimension and, if it converges, it converges to a local minimum. On the contrary, NKO requires the calculation of the derivatives, does not behave well with increasing dimension and can converge to a stationary point.

Table IV: RMS position error of the algorithms in each scenario

	1	2	3	4
NKOF	1.74	395.78	648.07	766.40
LMKOF	1.74	395.74	508.03	766.68
IEKF	1.97	395.75	528.93	766.70
ILMKF	1.74	403.84	508.03	1063.3
IUKF	2.01	733.13	1087.9	1515.2
UKF	31.02	423.76	322.63	757.84
CKF	28.84	437.13	328.35	757.82

We have also indicated important links between KO algorithms and Newton's and Levenberg-Marquardt methods. These relations are very important as they provide considerable insight into Kalman filtering. We have also shown that, in the limit, LMKO and NKO behave as Gauss-Newton and Newton's methods, respectively. Therefore, they have the same local convergence properties.

The analysis of the performance of the developed KO algorithms in optimisation problems different from MAP estimation is an interesting line of future research.

Appendix A

In this appendix, we prove the LMKO recursion to calculate the MAP, see Section V-A2. The procedure is quite similar to the one explained in [10]. Here, the expected Jacobian is written as $\mathbf{J}_i = -\mathbf{T}\left[\mathbf{H}_i^T, \mathbf{I}^T\right]^T$ and $\overline{\mathbf{y}}_i = \mathbf{T}\left[\left(\mathbf{z}-\overline{\mathbf{z}}_i\right)^T, \left(\overline{\mathbf{a}}-\overline{\mathbf{a}}_i\right)^T\right]^T$. The recursion for the mean, assuming that α_i^2 is selected as indicated in Section IV-B1, is

$$\begin{aligned}\overline{\mathbf{a}}_{i+1} =& \overline{\mathbf{a}}_i - \left(\alpha_i^2\mathbf{P}_i^{-1} + \mathbf{J}_i^T\mathbf{J}_i\right)^{-1}\mathbf{J}_i^T\overline{\mathbf{y}}_i \\ =& \left(\alpha_i^2\mathbf{P}_i^{-1} + \mathbf{J}_i^T\mathbf{J}_i\right)^{-1}\left[\mathbf{J}_i^T\left(-\overline{\mathbf{y}}_i + \mathbf{J}_i\overline{\mathbf{a}}_i\right) + \alpha_i^2\mathbf{P}_i^{-1}\overline{\mathbf{a}}_i\right] \\ =& \overline{\mathbf{a}} + \left(\mathbf{H}_i^T\mathbf{R}^{-1}\mathbf{H}_i + \mathbf{P}_0^{-1} + \alpha_i^2\mathbf{P}_i^{-1}\right)^{-1} \\ & \left[\mathbf{H}_i^T\mathbf{R}^{-1}\left(\mathbf{z}-\overline{\mathbf{z}}_i\right) - \mathbf{H}_i^T\mathbf{R}^{-1}\mathbf{H}_i\left(\overline{\mathbf{a}}-\overline{\mathbf{a}}_i\right)\right. \\ & \left.-\alpha_i^2\mathbf{P}_i^{-1}\left(\overline{\mathbf{a}}-\overline{\mathbf{a}}_i\right)\right] \\ =& \overline{\mathbf{a}} + \mathbf{L}_i\left(\mathbf{z}-\overline{\mathbf{z}}_i - \mathbf{H}_i\left(\overline{\mathbf{a}}-\overline{\mathbf{a}}_i\right)\right) \\ & -\alpha_i^2\left(\mathbf{I}-\mathbf{L}_i\mathbf{H}_i\right)\hat{\mathbf{P}}_i\mathbf{P}_i^{-1}\left(\overline{\mathbf{a}}-\overline{\mathbf{a}}_i\right)\end{aligned}$$

where $\mathbf{L}_i$ is given by (64) and $\hat{\mathbf{P}}_i$ by (63) and $\mathbf{P}_0 = \mathbf{\Sigma}_a$ according to (53). The recursion for the covariance matrix, assuming that α_i^2 is selected as indicated in Section IV-B1, is

$$\begin{aligned}\mathbf{P}_{i+1} =& \mathbf{P}_i - \mathbf{P}_i\mathbf{J}_i^T\left(\mathbf{J}_i\mathbf{P}_i\mathbf{J}_i^T + \alpha_i^2\mathbf{I}\right)^{-1}\mathbf{J}_i\mathbf{P}_i \\ =& \mathbf{P}_i - \left(\alpha_i^2\mathbf{P}_i^{-1} + \mathbf{J}_i^T\mathbf{J}_i\right)^{-1}\mathbf{J}_i^T\mathbf{J}_i\mathbf{P}_i \\ =& \mathbf{P}_i - \left(\alpha_i^2\mathbf{P}_i^{-1} + \mathbf{H}_i^T\mathbf{R}^{-1}\mathbf{H}_i + \mathbf{P}_0^{-1}\right)^{-1} \\ & \left(\mathbf{H}_i^T\mathbf{R}^{-1}\mathbf{H}_i + \mathbf{P}_0^{-1}\right)\mathbf{P}_i \\ =& \mathbf{P}_i - \mathbf{P}_i\left(\alpha_i^2\check{\mathbf{P}}_i + \mathbf{P}_i\right)^{-1}\mathbf{P}_i\end{aligned}$$

where $\check{\mathbf{P}}_i$ is given by (65).

Appendix B

In this appendix we show (71).

Lemma 1. *Under Assumption A, matrix* $\mathbf{\Psi}_i\mathbf{S}_i^{-1}\mathbf{\Psi}_i^T > \mathbf{0}$.

A matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ is $\mathbf{A} > \mathbf{0}$ if $\mathbf{x}^T\mathbf{A}\mathbf{x} > 0$ for $\mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$. To prove Lemma 1, we have to check that

$$\mathbf{x}^T\mathbf{\Psi}_i\mathbf{S}_i^{-1}\mathbf{\Psi}_i^T\mathbf{x} > 0 \tag{77}$$

for $\mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$ under Assumption A.

As $\mathbf{S}_i$ is a covariance matrix, it is positive definite and $\mathbf{S}_i^{-1}$ is positive definite too. Then, it has a Cholesky decomposition $\mathbf{S}_i^{-1} = \mathbf{S}_i^{-1/2}\left(\mathbf{S}_i^{-1/2}\right)^T$ where $\mathbf{S}_i^{-1/2}$ is invertible and positive definite. Then, (77) can be written as

$$\left\|\left(\mathbf{S}_i^{-1/2}\right)^T \mathbf{\Psi}_i^T \mathbf{x}\right\|^2 > 0.$$

The norm of a vector is always positive unless the vector is zero. In our case, if $\left(\mathbf{S}_i^{-1/2}\right)^T \mathbf{\Psi}_i^T \mathbf{x} = \mathbf{0}$, $\mathbf{x} = \mathbf{0}$ because $\left(\mathbf{S}_i^{-1/2}\right)^T$ is invertible and, under Assumption A, $\mathbf{\Psi}_i^T$ is invertible too. This proves Lemma 1.

Using (9), we get

$$\begin{aligned} \mathbf{x}^T \mathbf{P}_{i+1} \mathbf{x} =& \mathbf{x}^T \left(\mathbf{P}_i - \mathbf{\Psi}_i \mathbf{S}_i^{-1} \mathbf{\Psi}_i^T\right) \mathbf{x} \\ \mathbf{x}^T \mathbf{P}_{i+1} \mathbf{x} <& \mathbf{x}^T \mathbf{P}_i \mathbf{x}. \end{aligned} \tag{78}$$

We also know that

$$\mathbf{x}^T \mathbf{P}_i \mathbf{x} > 0 \, \forall \mathbf{x} \neq \mathbf{0} \tag{79}$$

because $\mathbf{P}_i$ is positive definite. Therefore, using (78) and (79), we get that

$$\lim_{i\to\infty} \mathbf{x}^T \mathbf{P}_i \mathbf{x} = \mathbf{x}^T \left(\lim_{i\to\infty} \mathbf{P}_i\right) \mathbf{x} = 0 \, \forall \mathbf{x} \in \mathbb{R}^{n_x}$$

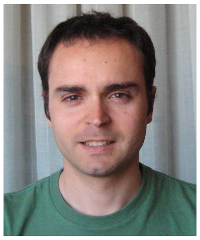
which implies that $\lim_{i\to\infty} \mathbf{P}_i = \mathbf{0}$.

Appendix C

In this appendix we show (73).

Lemma 2. *Matrix* $\mathbf{P}_i \left(\alpha_i^2 \check{\mathbf{P}}_i + \mathbf{P}_i\right)^{-1} \mathbf{P}_i > \mathbf{0}$.

The proof of Lemma 2 is as follows. As (65) is the update equation of the covariance matrix for a Kalman filter with a linear measurement [3], $\check{\mathbf{P}}_i$ is positive definitive. As $\alpha_i^2 > 0$ and $\mathbf{P}_i$ is positive definite, $\alpha_i^2 \check{\mathbf{P}}_i + \mathbf{P}_i$ is positive definite and so is its inverse. Then, we can use the same procedure as in Appendix B to finish the proof of (73).



Ángel F. García-Fernández received the telecommunication engineering degree (with honours) and the Ph.D. degree from Universidad Politécnica de Madrid, Madrid, Spain, in 2007 and 2011, respectively.
He is currently a Research Associate in the Department of Electrical and Computer Engineering at Curtin University, Perth, Australia. His main research activities and interests are in the area of Bayesian nonlinear inference and radar imaging.

Lennart Svensson was born in Älvängen, Sweden in 1976. He received the M.S. degree in electrical engineering in 1999 and the Ph.D. degree in 2004, both from Chalmers University of Technology, Gothenburg, Sweden.
He is currently Associate Professor at the Signal Processing group, again at Chalmers University of Technology. His research interests include Bayesian inference in general, and nonlinear filtering and tracking in particular.

References

[1] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, “A new method for the nonlinear transformation of means and covariances in filters and estimators,” *IEEE Transactions on Automatic Control*, vol. 45, no. 3, pp. 477–482, Mar. 2000.

[2] S. J. Julier and J. K. Uhlmann, “Unscented filtering and nonlinear estimation,” *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004.

[3] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice-Hall, 1993.

[4] I. Arasaratnam and S. Haykin, “Cubature Kalman filters,” *IEEE Transactions on Automatic Control*, vol. 54, no. 6, pp. 1254–1269, June 2009.

[5] I. Arasaratnam, S. Haykin, and R. Elliott, “Discrete-time nonlinear filtering algorithms using Gauss-Hermite quadrature,” *Proceedings of the IEEE*, vol. 95, no. 5, pp. 953–977, May 2007.

[6] M. R. Morelande and A. F. García-Fernández, “Analysis of Kalman filter approximations for nonlinear measurements,” *IEEE Transactions on Signal Processing*, vol. 61, no. 22, pp. 5477–5484, Nov. 2013.

[7] A. M. Walker, “On the asymptotic behaviour of posterior distributions,” *Journal of the Royal Statistical Society. Series B (Methodological).*, vol. 31, no. 1, pp. 80–88, 1969.

[8] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer, 1999.

[9] B. Bell and F. Cathey, “The iterated Kalman filter update as a Gauss-Newton method,” *IEEE Transactions on Automatic Control*, vol. 38, no. 2, pp. 294–297, Feb. 1993.

[10] R. L. Bellaire, E. W. Kamen, and S. M. Zabin, “A new nonlinear iterated filter with applications to target tracking,” in *Proc. SPIE 2561, Signal and Data Processing of Small Targets*, 1995.

[11] M. Fatemi, L. Svensson, L. Hammarstrand, and M. Morelande, “A study of MAP estimation techniques for nonlinear filtering,” in *15th International Conference on Information Fusion*, 2012, pp. 1058–1065.

[12] G. Sibley, G. Sukhatme, and L. Matthies, “The iterated sigma point Kalman filter with applications to long range stereo,” in *Proceedings of Robotics: Science and Systems*, 2006.

[13] T. Lefebvre, H. Bruyninckx, and J. De Schuller, “Comment on "a new method for the nonlinear transformation of means and covariances in filters and estimators" [and authors’ reply],” *IEEE Transactions on Automatic Control*, vol. 47, no. 8, pp. 1406–1409, Aug. 2002.

[14] R. Zhan and J. Wan, “Iterated unscented Kalman filter for passive target tracking,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 3, pp. 1155–1163, July 2007.

[15] C.-R. Hwang, “Laplace’s method revisited: weak convergence of probability measures,” *The Annals of Probability*, vol. 8, no. 6, pp. 1177–1182, Dec. 1980.

[16] A. Doucet and X. Wang, “Monte Carlo methods for signal processing: a review in the statistical signal processing context,” *IEEE Signal Processing Magazine*, vol. 22, no. 6, pp. 152–170, 2005.

[17] J. C. Spall, *Introduction to stochastic search and optimization*. John Wiley & Sons, 2003.

[18] J. Humpherys and J. West, “Kalman filtering with Newton’s method [lecture notes],” *IEEE Control Systems Magazine*, vol. 30, no. 6, pp. 101–106, Dec 2010.

[19] M. A. Osborne, R. Garnett, and S. J. Roberts, “Gaussian processes for global optimization,” in *International Conference on Learning and Intelligent Optimizatio*, 2009.

[20] J. Sarmavuori and S. Särkkä, “Fourier-Hermite Kalman filter,” *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1511–1515, June 2012.

[21] I. Arasaratnam and S. Haykin, “Square-root quadrature Kalman filtering,” *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2589–2593, June 2008.

[22] A. Gelman, J. B. Carlin, H. S. Stern, and D. B.Rubin, *Bayesian Data Analysis*. Chapman & Hall/CRC, 2004.

[23] T. M. Apostol, *Calculus. Volume II.* John Wiley & Sons, 1969.

[24] F. Beutler, M. F. Huber, and U. D. Hanebeck, “Gaussian filtering using state decomposition methods,” in *12th International Conference on Information Fusion*, July 2009, pp. 579–586.

[25] R. N. Bracewell, *The Fourier transform and its applications*. McGraw-Hill, 2000.

[26] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipies in C: The Art of Scientific Computing*. Cambridge University Press, 1992.

[27] S. Challa, M. R. Morelande, D. Musicki, and R. J. Evans, *Fundamentals of Object Tracking*. Cambridge University Press, 2011.