Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Real-Time Estimation of Parameter Maps

Maik Gentsch *,[1] **Rudibert King** *

* *Technische Universität Berlin, Chair of Measurement and Control, Straße des 17. Juni 135, 10623 Berlin, Germany*

Abstract: System parameters might have a distinct operating point dependency that is unknown. Nonlinear state observers or Kalman Filters can be applied to estimate such parameters in real-time, revealing the unknown parameter value in the vicinity of the current operating point. Commonly, these methods are prone to forget the revealed dependence continuously when a different operating point is approached. This paper provides a procedure to preserve past estimates and reveal the hidden parameter map during operation of the system. Parameter dependencies are approximated via adjustable interpolants. In particular, ready-to-use formulae for piecewise linear and cubic Hermite interpolants are provided. An existing approach as well as a newly derived approach to embed these interpolants within an Unscented Kalman Filter are presented and discussed. While the first approach utilizes the parameter map estimation directly within the Kalman Filter scheme, the new approach expands the Kalman Filter steps by a recursive map adaption scheme and is thereby far less computationally expensive. Both methods are compared and validated via numerical simulations, where a superior performance is achieved compared to the standard parameter estimation within the Kalman Filter approach.

Keywords: Model-based supervision, Parameter estimation, Unscented Kalman Filter, Interpolation

1. INTRODUCTION

In model-based supervision and adaptive control schemes, real-time estimation of parameters of a physical model that describes the objective to be monitored or controlled is an essential element. If the system comprises a dynamic behavior, a coupled real-time estimation of parameters and dynamic model states can be applied in the framework of observers or Kalman Filters. Commonly, in order to achieve a smooth parameter estimation over time, the parameters are assumed to vary much slower than the model states. Consequently, the estimator may not be capable of adapting the parameters fast enough during transients if a hidden operating point dependency of the parameters exists. For real systems, this is quite often the case, while it is rather difficult to model the correct operating point dependency without conducting additional, suitable experiments prior to the actual application. Furthermore, these dependencies, which are referred to as parameter maps in what follows, may be affected by system faults or aging that needs to be monitored.

This paper contributes to real-time estimation of such parameter maps. Here, the term *real-time* is stressed to denote a constant computational complexity and memory requirement. Therefore, the presented real-time method is formulated in a recursive manner and operates on fixed-sized arrays. To provide a unified concept for arbitrarily shaped parameter maps, interpolants are used to describe the parameter dependence on the operating point. When new measurements arise, the interpolants are adjusted in real-time using time-variant grid vectors, which are the actual parameters of the scheme proposed.

Such an approach has been presented by Höckerdal et al. (2011) already. There, a linear interpolant is embedded into an *Extended Kalman Filter* that performs a joint estimation of dynamic system states and grid vectors. While this method is well suited to track time-variant parameter maps, it is accompanied by a high computational cost for a large number of grid points (flexible interpolant). Therefore, in this contribution, an alternative method is derived that performs the estimation of parameters and parameter maps in sequence.

The paper is structured as follows: In Section 2, the concept of real-time state and parameter estimation within the framework of Kalman Filters is summarized to introduce the notation. The idea of preserving parameter estimates with adjustable interpolants is discussed in Section 3. The existing and the new approach to embed these interpolants into the Kalman Filter scheme for the purpose of real-time parameter map estimation are presented in Section 4. Then, a numerical simulation is conducted that aims to validate and emphasize the superior performance of the proposed scheme. Results are presented in Section 5.

In the subsequent sections, the $(\boldsymbol{v})_j$ or $(\mathbf{M})_{jl}$ notation is used to denote the j-th entry of a vector $\boldsymbol{v}$ or the j-th row and l-th column entry of a matrix $\mathbf{M}$, respectively.

[1] Corresponding author: maik.gentsch@tu-berlin.de

2. REAL-TIME ESTIMATION OF STATES AND PARAMETERS APPLYING KALMAN FILTERS

The approaches to map estimation proposed below can be incorporated in any state observer or filter scheme. However, for the investigation below, an *Unscented Kalman Filter* (UKF) is chosen, which is briefly introduced in the present section.

The objective of the applied UKF is to provide an estimate $\hat{\boldsymbol{x}}_k$ of the true states $\boldsymbol{x}_k$ of a discrete-time, nonlinear system

$$\begin{cases} \boldsymbol{x}_k = \boldsymbol{f}\left(\boldsymbol{x}_{k-1},\, \boldsymbol{u}_{k-1},\, \boldsymbol{\theta},\, \boldsymbol{r}^x_{k-1},\, k\right), \\ \boldsymbol{y}_k = \boldsymbol{g}\left(\boldsymbol{x}_k,\, \boldsymbol{u}_k,\, \boldsymbol{\theta},\, k\right) + \boldsymbol{r}^y_k . \end{cases} \tag{1}$$

Here, $\boldsymbol{y}_k \in \mathbb{R}^{n_y}$, $\boldsymbol{x}_k \in \mathbb{R}^{n_x}$, $\boldsymbol{u}_k \in \mathbb{R}^{n_u}$, and $\boldsymbol{\theta} \in \mathbb{R}^{n_\theta}$ are the measurable outputs, the states, the inputs, and the parameters of the model, respectively. A variable with an index k denotes a discrete-time quantity; e.g., $\boldsymbol{x}_k = \boldsymbol{x}(t = t_k)$. The system noise $\boldsymbol{r}^x_k \in \mathbb{R}^{n_x}$ and the additive measurement noise $\boldsymbol{r}^y_k \in \mathbb{R}^{n_y}$ are stochastic, zero-mean, uncorrelated, discrete-time signals with time-variant covariance matrices $\mathbf{R}^x_k \in \mathbb{R}^{n_x \times n_x}$ and $\mathbf{R}^y_k \in \mathbb{R}^{n_y \times n_y}$, respectively. Applying the expectation operator $E\{\}$,

$$\begin{aligned} E\{\, \boldsymbol{r}^x_k \boldsymbol{r}^{x\,T}_l \,\} &= \mathbf{R}^x_k\, \delta_{kl}\,, \qquad & E\{\, \boldsymbol{r}^y_k \boldsymbol{r}^{y\,T}_l \,\} &= \mathbf{R}^y_k\, \delta_{kl}\,, \\ E\{\, \boldsymbol{r}^x_k \boldsymbol{r}^{y\,T}_l \,\} &= \mathbf{0}_{n_x \times n_y}\,, & \forall\; k,\, l\,, \end{aligned} \tag{2}$$

follows, where $\delta_{kl} = 1$ for $l = k$ and, otherwise, $\delta_{kl} = 0$. Beside the state estimate $\hat{\boldsymbol{x}}$, any Kalman Filter provides an estimate $\mathbf{P}^x_k \in \mathbb{R}^{n_x \times n_x}$ of the estimation error covariance matrix $E\{\,(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)^T\,\}$.

As the focus of this contribution is on the description and embedding of hidden parameter maps, the reader is referred to (Julier et al. (2000); Van der Merwe and Wan (2001)) for a comprehensive introduction to the UKF scheme and its implementation. For the investigation below, a UKF concept proposed by Kolås et al. (2009) is applied which ensures bounded state estimates $\boldsymbol{x}_l \le \hat{\boldsymbol{x}}_k \le \boldsymbol{x}_u$ via extensions that are referred to as *sigma point clipping* and *reformulated correction step.* The whole procedure of a single Kalman Filter recursion using current measurements $\boldsymbol{y}_k$ will be abbreviated as follows:

$$\text{KF:}\quad \{\hat{\boldsymbol{x}}_{k-1},\, \mathbf{P}_{k-1},\, \boldsymbol{y}_k,\, \mathbf{R}^x_{k-1},\, \mathbf{R}^y_k\} \longmapsto \{\hat{\boldsymbol{x}}_k,\, \mathbf{P}_k\}\,. \tag{3}$$

A joint estimation of states and parameters can easily be achieved by replacing the state vector $\boldsymbol{x}$ and the update function $\boldsymbol{f}$ with their augmented counterparts $\boldsymbol{x}^a = \left[\boldsymbol{x}^T\ \boldsymbol{\theta}^T\right]^T$ and $\boldsymbol{f}^a = \left[\boldsymbol{f}^T\ \boldsymbol{f}^{\theta\,T}\right]^T$, respectively, if the augmented system is observable. Usually, the real dynamic parameter evolution over time is unknown, and therefore,

$$\boldsymbol{\theta}_k = \boldsymbol{f}^\theta\left(\boldsymbol{x}^a_{k-1},\, \boldsymbol{u}_{k-1},\, \boldsymbol{r}^a_{k-1},\, k\right) = \boldsymbol{\theta}_{k-1} + \boldsymbol{r}^\theta_{k-1} \tag{4}$$

is a common parameter update function, where $\boldsymbol{r}^a = \left[\boldsymbol{r}^{x\,T}\ \boldsymbol{r}^{\theta\,T}\right]^T$ is the augmented system noise with covariance matrix $\mathbf{R}^a_k = \begin{bmatrix} \mathbf{R}^x_k & \mathbf{0}_{n_x \times n_\theta} \\ \mathbf{0}_{n_\theta \times n_x} & \mathbf{R}^\theta_k \end{bmatrix}$. $\mathbf{R}^x_k$, $\mathbf{R}^\theta_k$, $\mathbf{R}^y_k$, as well as the initial estimate $\hat{\boldsymbol{x}}^a_0$ and estimation error covariance matrix $\mathbf{P}^a_0$ of the augmented system are tuning parameters of this approach.

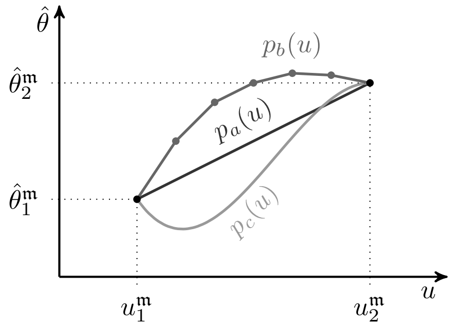

Fig. 1. Hidden map approximation

3. PARAMETER PRESERVATION VIA GRID VECTORS

By definition, a hidden dependency of the operating point on parameters is unknown. Consider, for instance, a situation where the system was running at two different operating points for a long time, adjusted via scalar inputs $u^{\mathfrak{m}}_1$ and $u^{\mathfrak{m}}_2$, respectively. The superscript $\mathfrak{m}$ refers to a value that is connected with a map presentation in what follows. Two corresponding reliable, but deviating estimates, $\hat{\theta}^{\mathfrak{m}}_1$ and $\hat{\theta}^{\mathfrak{m}}_2$, were obtained for some unknown parameter. For predicting this parameter for an operation with an input between $u^{\mathfrak{m}}_1$ and $u^{\mathfrak{m}}_2$, an interpolation between $\hat{\theta}^{\mathfrak{m}}_1$ and $\hat{\theta}^{\mathfrak{m}}_2$ would be the most intuitive approach. A simple linear interpolation with the interpolant $p_a(u)$ in Fig. 1 might serve this purpose. Whether $p_a(u)$ is well suited to approximate the hidden dependency, cannot be evaluated until new operating points have been approached via a set of inputs $U \subsetneq\,]u^{\mathfrak{m}}_1;\, u^{\mathfrak{m}}_2[$ and a new set of parameter estimates $\Theta = \{\hat{\theta}(u) \,|\, u \in U\}$ has been determined.

Vice versa, if Θ is given first, one could ask for the values $\hat{\boldsymbol{\theta}}^{\mathfrak{m}} = \left[\hat{\theta}^{\mathfrak{m}}_1\ \hat{\theta}^{\mathfrak{m}}_2\right]^T$ that yield an optimal regression function $p_a(U)$ for the set Θ. In this scenario, $\hat{\theta}^{\mathfrak{m}}_1$ and $\hat{\theta}^{\mathfrak{m}}_2$ act as adjustable grid values within an adjustable interpolant

$$p_a\left(u,\, \hat{\boldsymbol{\theta}}^{\mathfrak{m}}\right) = \frac{u^{\mathfrak{m}}_2 - u}{u^{\mathfrak{m}}_2 - u^{\mathfrak{m}}_1}\, \hat{\theta}^{\mathfrak{m}}_1 + \frac{u - u^{\mathfrak{m}}_1}{u^{\mathfrak{m}}_2 - u^{\mathfrak{m}}_1}\, \hat{\theta}^{\mathfrak{m}}_2\,, \quad u \in U\,, \tag{5}$$

the structure of which retains a linear interpolation between $\hat{\theta}^{\mathfrak{m}}_1$ and $\hat{\theta}^{\mathfrak{m}}_2$. The optimal regression function, more precisely the optimal grid vector $\hat{\boldsymbol{\theta}}^{\mathfrak{m}}$, accompanied by some additional measure for the approximation error, preserves the former operating point dependency $U \to \Theta$ to some extent. In a real-time application, where the number of elements in U and Θ would increase continuously, but computational resources and memory are limited, it appears unavoidable to take advantage of such a parameter preservation and discard previous estimates after some recursive grid vector adaption has been applied.

Accordingly, the grid vectors become time-variant parameters that need to be estimated, but the structure of the interpolant, i.e., the type of interpolation and the number of grid points, remains the same for all time steps. If the parameter map is expected to be far more complex than a linear line (univariate interpolation) or hyperplane (multivariate interpolation), then the number of grid points of a piecewise linear interpolant can be increased, as it is depicted for $p_b(u)$ in Fig. 1. Alternatively, a more flexible interpolant, like the (piecewise) cubic interpolant $p_c(u)$, can be chosen. It should be noted that selecting a very flexible interpolant might hold a risk for over-parametrization. A polynomial interpolant with many grid points, for instance, might be prone to inherent uncertainties in Θ. A countermeasure to over-parametrization is regularization which is introduced in Section 4.2.

The proposed concept considers the use of interpolants that can be transformed into the generic structure

$$p\,(i_1, i_2, \ldots, i_{n_i}, \boldsymbol{z}^{\mathtt{m}}) = \boldsymbol{c}^T\big(\boldsymbol{i}_1^{\mathtt{m}}, \boldsymbol{i}_2^{\mathtt{m}}, \ldots, \boldsymbol{i}_{n_i}^{\mathtt{m}}, i_1, i_2, \ldots, i_{n_i}\big) \cdot \boldsymbol{z}^{\mathtt{m}}\,, \tag{6}$$

where i_j denote *independent variables*, $\boldsymbol{i}_j^{\mathtt{m}} \in \mathbb{R}^{n_{\mathtt{m}}}$ are the corresponding nodes of the map along i_j, n_i is the number of independent variables, $\boldsymbol{c} \in \mathbb{R}^{n_z}$ contains a set of grid dependent coefficients, and $\boldsymbol{z}^{\mathtt{m}} \in \mathbb{R}^{n_z}$ is the adjustable grid vector. For the example above, with p_a according to (5), these variables are $n_i = 1$, $i_1 = u$, $\boldsymbol{i}_1^{\mathtt{m}} = [u_1^{\mathtt{m}}\ u_2^{\mathtt{m}}]^T$, $\boldsymbol{z}^{\mathtt{m}} = \hat{\boldsymbol{\theta}}^{\mathtt{m}}$, and $\boldsymbol{c}^T = [c_1\ c_2]$. The independent variable i_j may represent a specific state from $\hat{\boldsymbol{x}}$ or a single input from $\boldsymbol{u}$. Quite clearly, the method's hassle is the inevitable need to calculate $\boldsymbol{c}$. Every further dependency that is considered increases n_i and, consequently, the computational load and memory requirement. Therefore, it is advisable to take as few independent variables as possible into account. Within the scope and space of this paper, only univariate $(n_i = 1)$ parameter maps are considered. In the subsequent sections, the derivation of $\boldsymbol{c}$ for interpolants applied below is provided.

3.1 Univariate Linear Interpolation

A univariate linear interpolation with two nodes and two coefficients has already been presented in Eq. (5). With $n_{\mathtt{m}}$ grid points $(i_{1,j}^{\mathtt{m}}, \hat{\theta}_j^{\mathtt{m}})$, $j = \{1, 2, \ldots, n_{\mathtt{m}}\}$, where an ascending order $i_{1,j}^{\mathtt{m}} < i_{1,j+1}^{\mathtt{m}}$ is satisfied, the j-th entry of the linear coefficient vector $\boldsymbol{c}^{\text{lin}} \in \mathbb{R}^{n_z}$ is [2]

$$(\boldsymbol{c}^{\text{lin}})_j = \begin{cases} \dfrac{i_{1,j+1}^{\mathtt{m}} - i_1}{i_{1,j+1}^{\mathtt{m}} - i_{1,j}^{\mathtt{m}}}\,, & i_1 \in \big[i_{1,j}^{\mathtt{m}};\, i_{1,j+1}^{\mathtt{m}}\big[\,, \\ & \text{or } i_1 < i_{1,1}^{\mathtt{m}} \text{ and } j = 1\,, \\ & \text{or } i_1 \geq i_{1,n_{\mathtt{m}}}^{\mathtt{m}} \text{ and } j = n_{\mathtt{m}} - 1 \\ \dfrac{i_1 - i_{1,j-1}^{\mathtt{m}}}{i_{1,j}^{\mathtt{m}} - i_{1,j-1}^{\mathtt{m}}}\,, & i_1 \in \big[i_{1,j-1}^{\mathtt{m}};\, i_{1,j}^{\mathtt{m}}\big[\,, \\ & \text{or } \ i_1 < i_{1,1}^{\mathtt{m}} \text{ and } j = 2\,, \\ & \text{or } \ i_1 \geq i_{1,n_{\mathtt{m}}}^{\mathtt{m}} \text{ and } j = n_{\mathtt{m}} \\ 0\,, & \text{else}\,. \end{cases} \tag{7}$$

The optional clauses for $i_1 < i_{1,1}^{\mathtt{m}}$ and $i_1 \geq i_{1,n_{\mathtt{m}}}^{\mathtt{m}}$ provide a linear extrapolation. For this interpolant, the adjustable grid vector is constructed via $(\boldsymbol{z}_{\text{lin}}^{\mathtt{m}})_j = \hat{\theta}_j^{\mathtt{m}}$, and thus, $n_z = n_{\mathtt{m}}$.

3.2 Univariate Cubic Hermite Interpolation

Again, $n_{\mathtt{m}}$ grid points $(i_{1,j}^{\mathtt{m}}, \hat{\theta}_j^{\mathtt{m}})$, $j = \{1, 2, \ldots, n_{\mathtt{m}}\}$, with an ascending order $i_{1,j}^{\mathtt{m}} < i_{1,j+1}^{\mathtt{m}}$ are required. The interpolant p^{cub} provided below is *cubic Hermite*; i.e., the interpolation is C^1 continuous and piecewise cubic. The derivative at a node is stored as the j-th entry of $\hat{\boldsymbol{d}}^{\mathtt{m}} \in \mathbb{R}^{n_{\mathtt{m}}}$; i.e. $(\hat{\boldsymbol{d}}^{\mathtt{m}})_j = \partial\, p^{\text{cub}}(i_1 = i_{1,j}^{\mathtt{m}})/\partial\, i_1$. The interpolant can be arranged as

$$p^{\text{cub}}(\,i_1,\, \boldsymbol{z}_{\text{cub}}^{\mathtt{m}}\,) = \underbrace{\Big[\boldsymbol{c}^{\theta\,T}(\,\boldsymbol{i}_1^{\mathtt{m}}, i_1\,)\ \boldsymbol{c}^{d\,T}(\,\boldsymbol{i}_1^{\mathtt{m}}, i_1\,)\Big]}_{\boldsymbol{c}^{\text{cub}\,T}(\,\boldsymbol{i}_1^{\mathtt{m}}, i_1\,)} \cdot \underbrace{\begin{bmatrix} \hat{\boldsymbol{\theta}}^{\mathtt{m}} \\ \hat{\boldsymbol{d}}^{\mathtt{m}} \end{bmatrix}}_{\boldsymbol{z}_{\text{cub}}^{\mathtt{m}}}\,, \tag{8}$$

the structure of which is compliant to (6). Since $\boldsymbol{z}_{\text{cub}}^{\mathtt{m}}$ has been declared the adjustable grid vector, the approach proposed yields a joint estimation of grid values $(\hat{\boldsymbol{\theta}}^{\mathtt{m}})_j = \hat{\theta}_j^{\mathtt{m}}$ and derivatives at these points $(\hat{\boldsymbol{d}}^{\mathtt{m}})_j$, and thus, $n_z = 2\,n_{\mathtt{m}}$. The j-th entry of coefficient vectors $\boldsymbol{c}^{\theta} \in \mathbb{R}^{n_{\mathtt{m}}}$ and $\boldsymbol{c}^{d} \in \mathbb{R}^{n_{\mathtt{m}}}$ are determined as follows:

$$(\boldsymbol{c}^{\theta})_j = \begin{cases} 1\,, & i_1 < i_{1,1}^{\mathtt{m}} \text{ and } j = 1\,, \\ & \text{or } i_1 \geq i_{1,n_{\mathtt{m}}}^{\mathtt{m}} \text{ and } j = n_{\mathtt{m}} \\ c_{0,j}^{\theta}(\,\boldsymbol{i}_1^{\mathtt{m}}, i_1\,)\,, & i_1 \in \big[i_{1,j}^{\mathtt{m}};\, i_{1,j+1}^{\mathtt{m}}\big[\\ c_{1,j-1}^{\theta}(\,\boldsymbol{i}_1^{\mathtt{m}}, i_1\,)\,, & i_1 \in \big[i_{1,j-1}^{\mathtt{m}};\, i_{1,j}^{\mathtt{m}}\big[\\ 0\,, & \text{else}\,. \end{cases} \tag{9}$$

$$(\boldsymbol{c}^{d})_j = \begin{cases} i_1 - i_{1,j}^{\mathtt{m}}\,, & i_1 < i_{1,1}^{\mathtt{m}} \text{ and } j = 1\,, \\ & \text{or } i_1 \geq i_{1,n_{\mathtt{m}}}^{\mathtt{m}} \text{ and } j = n_{\mathtt{m}} \\ c_{0,j}^{d}(\,\boldsymbol{i}_1^{\mathtt{m}}, i_1\,)\,, & i_1 \in \big[i_{1,j}^{\mathtt{m}};\, i_{1,j+1}^{\mathtt{m}}\big[\\ c_{1,j-1}^{d}(\,\boldsymbol{i}_1^{\mathtt{m}}, i_1\,)\,, & i_1 \in \big[i_{1,j-1}^{\mathtt{m}};\, i_{1,j}^{\mathtt{m}}\big[\\ 0\,, & \text{else}\,. \end{cases} \tag{10}$$

The optional clauses for $i_1 < i_{1,1}^{\mathtt{m}}$ and $i_1 \geq i_{1,n_{\mathtt{m}}}^{\mathtt{m}}$ are derived to provide a C^1 continuous linear extrapolation. The basis functions contained are (cf. (Kahaner et al., 1989, p. 106))

$$c_{\alpha,j}^{\theta}(\,\boldsymbol{i}_1^{\mathtt{m}}, i_1\,) = \frac{2 - 4\,\alpha}{\big[i_{1,j+1}^{\mathtt{m}} - i_{1,j}^{\mathtt{m}}\big]^3}\,\big[i_1 - i_{1,j+1-\alpha}^{\mathtt{m}}\big]^2 \cdot \left[i_1 - i_{1,j+\alpha}^{\mathtt{m}} + \frac{1 - 2\,\alpha}{2}\,\big[i_{1,j+1}^{\mathtt{m}} - i_{1,j}^{\mathtt{m}}\big]\right]\,, \tag{11}$$

$$c_{\alpha,j}^{d}(\,\boldsymbol{i}_1^{\mathtt{m}}, i_1\,) = \big[i_{1,j+1}^{\mathtt{m}} - i_{1,j}^{\mathtt{m}}\big]^{-2} \cdot \big[i_1 - i_{1,j}^{\mathtt{m}}\big]^{1+\alpha}\,\big[i_1 - i_{1,j+1}^{\mathtt{m}}\big]^{2-\alpha}\,, \tag{12}$$

where $\alpha \in \{0,\, 1\}$.

4. REAL-TIME PARAMETER MAP ESTIMATION

In the subsequent sections, two methods are presented that aim to estimate a hidden parameter map in real-time. Therefore, the grid vector $\boldsymbol{z}^{\mathtt{m}}$ of a preset interpolant is adjusted in a way that an approximation error is reduced. Both methods utilize the joint estimation of states and parameters, as presented in Section 2, in a different fashion. To avoid excessive use of indexes, we assume that only one parameter θ of the original model (1) has to be estimated and this parameter is assumed to have a hidden operating point dependence. The concept can easily be extended for additional unknown parameters and unknown parameter maps.

[2] Note that $i_1 \notin \big[i_{1,\alpha}^{\mathtt{m}};\, i_{1,\beta}^{\mathtt{m}}\big[$ if $\alpha < 1$ or $\beta > n_{\mathtt{m}}$.

4.1 Joint Estimation of States and Grid Vectors

For this method, the parameter is replaced by the preset interpolant $\theta = p(i_1, i_2, \ldots, i_{n_i}, \boldsymbol{z}^{\mathtt{m}})$ within the original model $(\boldsymbol{f}, \boldsymbol{g})$. There, a subset of states and inputs is assigned as independent variables i_j, whose values determine the operating point. From Section 3 it is clear that the adjustable grid vector $\boldsymbol{z}^{\mathtt{m}}$ contains the actual parameters to be estimated. Following this idea and the procedure from Section 2, the state vector can be augmented as $\boldsymbol{x}^a := \left[\boldsymbol{x}^T \ \boldsymbol{z}^{\mathtt{m}T}\right]^T \in \mathbb{R}^{n_x+n_z}$, with the aim to perform a joint estimation of states and grid vectors within the Kalman Filter scheme; cf. (3):

$$\begin{aligned} \text{KF:} \quad & \{\hat{\boldsymbol{x}}^a_{k-1}, \mathbf{P}^a_{k-1}, \boldsymbol{y}_k, \mathbf{R}^a_{k-1}, \mathbf{R}^y_k\} \longmapsto \{\hat{\boldsymbol{x}}^a_k, \mathbf{P}^a_k\}, \\ & (\boldsymbol{z}^{\mathtt{m}}_k)_j = (\hat{\boldsymbol{x}}^a_k)_{j+n_x}, \qquad j = \{1, 2, \ldots, n_z\}. \end{aligned} \tag{13}$$

As mentioned above, this approach was proposed already by Höckerdal et al. (2011). There, a univariate linear interpolant was used to describe the parameter map and the joint estimation was applied within the *Extended Kalman Filter* (EKF) scheme.

Without appropriate countermeasures, this approach is prone to continuously forget

(i) very past grid value estimates of the current operating range and
(ii) even recent estimates of departed operating ranges.

Due to issue (i), the method is capable of tracking time-variant parameter maps that may arise from system aging or faults. Issue (ii) is a basically undesired behavior that results from a lack of observability of parameters in $\boldsymbol{z}^{\mathtt{m}}$ that are not involved into the current interpolation (that is where the corresponding coefficient vector entry is $c_j = 0$). For a detailed discussion on both issues and an exhaustive observability analysis, which is beyond the scope and space of this paper, the reader is referred to (Höckerdal et al. (2011)).

Fortunately, Höckerdal et al. (2011) provide a simple countermeasure that reduces the effect of issue (ii) and prevent the divergence of the filter, which may arise otherwise. They propose to restrict the diagonal elements of the augmented covariance matrix estimate, thus $(\mathbf{P}^a_k)_{jj} \leq (\mathbf{P}^a_0)_{jj}$ for the probably unobservable "states" $j = \{n_x+1, n_x+2, \ldots, n_x+n_z\}$, where $\mathbf{P}^a_0 \in \mathbb{R}^{(n_x+n_z)\times(n_x+n_z)}$ is the initial augmented covariance matrix of the estimation error; cf. Section 2.

Since the computational complexity of the standard EKF/UKF algorithm is $\mathcal{O}(n_a^3)$, cf. (Van der Merwe and Wan (2001)), where n_a is the dimension of the augmented state vector, the augmentation by $n_z = n_{\mathtt{m}}$ (linear interpolant) or $n_z = 2\,n_{\mathtt{m}}$ (cubic Hermite interpolant) additional states is computationally expensive, compared to a common joint estimation of states and parameter θ, where $n_a = n_x + 1$, but no parameter map is estimated.

4.2 Dual Estimation of Parameters and Grid Vectors

Here, an alternative concept is presented, where the number of states is still $n_a = n_x + 1$ for the UKF; i.e., $\boldsymbol{x}^a = \left[\boldsymbol{x}^T \ \delta\right]^T \in \mathbb{R}^{n_x+1}$. In addition, an $\mathcal{O}(n_z^2)$ algorithm that accomplishes the recursive adaption of the grid vector $\boldsymbol{z}^{\mathtt{m}} \in \mathbb{R}^{n_z}$ is utilized. Consequently, this approach is computationally far less expensive than the joint estimation of (states,) parameters and grid vectors from Section 4.1. An analogous lack of observability is eliminated via *regularization*, thus, a comparable behavior like issue (ii) from the previous section is avoided. Without modification, however, the concept is rather suited for identifying time-invariant maps than for tracking time-variant maps (aging).

For the proposed concept, the parameter is replaced by the preset interpolant $\theta = p(i_1, i_2, \ldots, i_{n_i}, \boldsymbol{z}^{\mathtt{m}}) + \delta$ within the original model $(\boldsymbol{f}, \boldsymbol{g})$; i.e. δ presents a *map offset*. Applying a Kalman Filter recursion with the augmented state vector $\boldsymbol{x}^a := \left[\boldsymbol{x}^T \ \delta\right]^T$, an estimate of the map offset $\hat{\delta}_k$ is achieved; cf. (3):

$$\begin{aligned} \text{KF:} \quad & \{\hat{\boldsymbol{x}}^a_{k-1}, \mathbf{P}^a_{k-1}, \boldsymbol{y}_k, \mathbf{R}^a_{k-1}, \mathbf{R}^y_k\} \longmapsto \{\hat{\boldsymbol{x}}^a_k, \mathbf{P}^a_k\}, \\ & \hat{\delta}_k = (\hat{\boldsymbol{x}}^a_k)_{n_a}. \end{aligned} \tag{14}$$

Then, $\hat{\delta}_k$ is used to adjust the interpolant by a recursively formulated adaption of the grid vector:

$$\boldsymbol{z}^{\mathtt{m}}_k = \boldsymbol{z}^{\mathtt{m}}_{k-1} + \mathbf{Z}_k\, \boldsymbol{c}_k w_{1,k}\, \hat{\delta}_k\,. \tag{15}$$

Here, $\boldsymbol{c}_k$ is an abbreviation for the coefficient vector (cf. Section 3) at the current operating point, $w_{1,k}$ is a time-variant weight, and $\mathbf{Z}_k \in \mathbb{R}^{n_z\times n_z}$ is the *inverted information matrix*, which can be calculated recursively, too:

$$\mathbf{Z}_k = \mathbf{Z}_{k-1} - \frac{\mathbf{Z}_{k-1}\,\boldsymbol{c}_k\,\boldsymbol{c}_k^T\,\mathbf{Z}_{k-1}^T}{w_{1,k}^{-1} + \boldsymbol{c}_k^T\,\mathbf{Z}_{k-1}\,\boldsymbol{c}_k}\,. \tag{16}$$

Eqs. (15)–(16) are known as the *Recursive Least Squares* (RLS) algorithm, cf. (Ljung, 1999, pp. 363 ff.), which is $\mathcal{O}(n_z^2)$ (Jiang and Zhang (2004)). Consequently, any known issue and modification of the RLS algorithm that can be found in the literature may apply. The distinguishing feature of the proposed *Recursive Map Adaption* (RMA) is *regularization*, which enters the RMA algorithm exclusively in the initialization step:

$$\boldsymbol{z}^{\mathtt{m}}_0 = \mathbf{Z}_0\mathbf{W}_2\,\boldsymbol{z}^{\mathtt{m}}_*\,, \quad \mathbf{Z}_0 = \left[\mathbf{W}_2 + \mathbf{L}_g + \mathbf{L}_c\right]^{-1}\,. \tag{17}$$

Consider a batch of consecutive parameter estimates $\hat{\theta}_j = \boldsymbol{c}_j^T \boldsymbol{z}^{\mathtt{m}}_{j-1} + \hat{\delta}_j$ for $j = \{1, 2, \ldots, k\}$ with $\hat{\delta}_j$ obtained from (14) and variables $\hat{\boldsymbol{\theta}}_k \in \mathbb{R}^k$, $\mathbf{C}_k \in \mathbb{R}^{k\times n_z}$, $\mathbf{W}_{1,k} \in \mathbb{R}^{k\times k}$ preserving that batch:

$$(\hat{\boldsymbol{\theta}}_k)_j = \hat{\theta}_j\,, \quad (\mathbf{C}_k)_{jl} = (\boldsymbol{c}_j)_l\,, \quad (\mathbf{W}_{1,k})_{jl} = \begin{cases} w_{1,j}\ , & l = j \\ 0\ , & \text{else}\,. \end{cases} \tag{18}$$

Applying the RMA procedure (17)→(16)↻(15), a recursive solution of the optimization task

$$\arg\min_{\boldsymbol{z}^{\mathtt{m}}_k} \ \left[\hat{\boldsymbol{\theta}}_k - \mathbf{C}_k\,\boldsymbol{z}^{\mathtt{m}}_k\right]^T \mathbf{W}_{1,k}\left[\hat{\boldsymbol{\theta}}_k - \mathbf{C}_k\,\boldsymbol{z}^{\mathtt{m}}_k\right] \tag{19}$$

$$+ \left[\boldsymbol{z}^{\mathtt{m}}_k - \boldsymbol{z}^{\mathtt{m}}_*\right]^T \mathbf{W}_2 \left[\boldsymbol{z}^{\mathtt{m}}_k - \boldsymbol{z}^{\mathtt{m}}_*\right] + \boldsymbol{z}^{\mathtt{m}T}_k \left[\mathbf{L}_g + \mathbf{L}_c\right] \boldsymbol{z}^{\mathtt{m}}_k \tag{20}$$

is obtained. The first term (19) yields a weighted least squares regression of the adjustable interpolant $p = \boldsymbol{c}^T\boldsymbol{z}^{\mathtt{m}}$ with regard to the batch of parameter estimates contained in $\hat{\boldsymbol{\theta}}_k$. The second term with time-invariant weighting matrix $\mathbf{W}_2$ is a type of Tikhonov regularization of the probably ill-posed problem (19), penalizing deviations from a nominal grid vector $\boldsymbol{z}^{\mathtt{m}}_*$, which is a tuning parameter of the RMA. Regularization terms $\mathbf{L}_g$ and $\mathbf{L}_c$ penalize approximate gradients and curvatures, respectively, in order to obtain a smooth parameter map, even for a high number of grid points. For the univariate case, they are determined as follows:

$$\mathbf{L}_g = \frac{1}{n_{\mathrm{m}} - 1} \left[\boldsymbol{\Delta}_1 \boldsymbol{\Gamma}_1\right]^T \mathbf{W}_g \left[\boldsymbol{\Delta}_1 \boldsymbol{\Gamma}_1\right] , \tag{21}$$

$$\mathbf{L}_c = \frac{4}{n_{\mathrm{m}} - 2} \left[\boldsymbol{\Delta}_2 \boldsymbol{\Gamma}_2 \boldsymbol{\Delta}_1 \boldsymbol{\Gamma}_1\right]^T \mathbf{W}_c \left[\boldsymbol{\Delta}_2 \boldsymbol{\Gamma}_2 \boldsymbol{\Delta}_1 \boldsymbol{\Gamma}_1\right] , \tag{22}$$

where matrices $\boldsymbol{\Delta}_\alpha \in \mathbb{R}^{(n_z-\alpha)\times(n_z-\alpha)}$ and $\boldsymbol{\Gamma}_\alpha \in \mathbb{R}^{(n_z-\alpha)\times(n_z-\alpha+1)}$ for $\alpha \in \{1, 2\}$ are defined as follows:

$$(\boldsymbol{\Delta}_\alpha)_{jl} = \begin{cases} \left[i^{\mathrm{m}}_{1,j+\alpha} - i^{\mathrm{m}}_{1,j}\right]^{-1}, & l = j \text{ and } j \leq n_{\mathrm{m}} - \alpha \\ 0 & , \text{ else} . \end{cases} \tag{23}$$

$$(\boldsymbol{\Gamma}_\alpha)_{jl} = \begin{cases} -1 & , \; l = j \\ 1 & , \; l = j + 1 \\ 0 & , \text{ else} . \end{cases} \tag{24}$$

Any introduced weighting matrix $\mathbf{W}$ is symmetric and of appropriate dimension. In particular, $\mathbf{W}_2$ is positive definite and the remaining weighting matrices are at least positive semi-definite. Since the adaption of the interpolant via RMA is applied next to the UKF scheme, $\hat{\boldsymbol{x}}$ has to be adjusted in order to keep the actual *posteriori* UKF "parameter estimate" $\hat{\theta}_k = \boldsymbol{c}_k^T \boldsymbol{z}^{\mathrm{m}}_{k-1} + \hat{\delta}_k$ as *a priori* estimate for the next UKF recursion (14) by a proper declaration:

$$(\hat{\boldsymbol{x}}^a_k)_{n_a} := \hat{\theta}_k - \boldsymbol{c}_k^T \boldsymbol{z}^{\mathrm{m}}_k . \tag{25}$$

5. NUMERICAL STUDY - COMPARISON AND RESULTS

Both approaches, the *Joint Estimation* (JE) from Section 4.1 and the *Dual Estimation* (DE) from Section 4.2, are applied to estimate the (voltage) gain θ of a (peristaltic) pump that supplies water into a throughput vessel. Therefore, noisy measurements of the vessel's filling level, which is the sole dynamic state of the nonlinear system

$$\begin{cases} x_k = x_{k-1} + \dfrac{1}{a_1} \displaystyle\int_{t_{k-1}}^{t_k} \left(\theta(u_{k-1})\, u_{k-1} - a_2 \sqrt{2g\, x(t)}\right) \mathrm{d}t , \\ y_k = x_k + r^y_k , \end{cases} \tag{26}$$

are considered; i.e., $n_x = n_y = 1$. Here, a_1 and a_2 denote specific areas in the vessel system and g is gravity. The true parameter dependence $\theta(u)$ can be seen in Fig. 2. Contrary, a typical model description for that type of pump would be a constant gain ($\theta_{k+1} = \theta_k \; \forall\, k$), which has to be identified from past experiments or estimated in real-time during operation. The joint estimation of state and parameter as presented in Section 2 might serve the latter purpose. Such an approach is referred to as *Standard parameter Estimation* (SE) in the following.

The numerical simulation conducted can be seen in Fig. 3. To enable a parameter adaption $\|\hat{\boldsymbol{\theta}}_{k+1} - \hat{\boldsymbol{\theta}}_k\| > 0$ within the Kalman Filter scheme,[3] a positive noise covariance $\mathbf{R}^\theta_k = q_k \mathbf{R}^\theta_0$ is required. Considering Fig. 3a and b, the effect of q_k can be studied. For $q_k \gg 10^{-2}$ (initial phase) the estimates are unacceptable noisy. For $q_k \approx 10^{-2}$, even the SE is capable of tracking the true time-variant gain,

[3] Note that $\hat{\boldsymbol{\theta}}$ represents $\hat{\theta}$, $\hat{\delta}$, or $\boldsymbol{z}^{\mathrm{m}}$ for the SE, DE, or JE, respectively, in this context. Therefore, $\mathbf{R}^\theta$ is scalar for SE and DE.

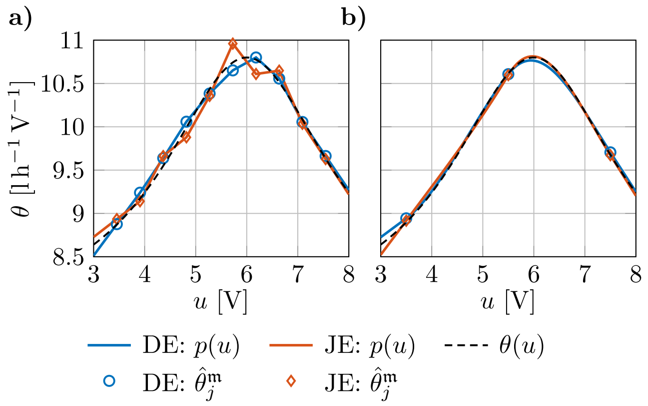

Fig. 2. True (dashed) and estimated (solid) pump gain map at $k = 2500$ applying the *Dual Estimation* (DE) and the *Joint Estimation* (JE) approach; a) with linear interpolants ($n_{\mathrm{m}} = 10$) and b) with cubic Hermite interpolants ($n_{\mathrm{m}} = 3$)

SE	linear ($n_{\mathrm{m}} = 10$)		cubic ($n_{\mathrm{m}} = 3$)	
	DE	JE	DE	JE
1	1.02	10.49	1.06	26.01

Table 1. Normalized duration measured to perform the *Standard Estimation* (SE), *Dual Estimation* (DE), and *Joint Estimation* (JE)

but the estimates are still noisy and slightly delayed. When q_k decreases further, the SE gets smoother, but the parameter dependence on u falls into oblivion, naturally, since no preservation scheme is applied. If a preservation scheme is applied, as is the case for the DE, the resulting quality of estimates can improve if q_k decreases after sufficient information at relevant operating points has been incorporated.

The final map estimates are depicted in Fig. 2 based on linear interpolants with 10 grid points (left) and based on cubic Hermite interpolants with three grid points (right). The noticeably smoother shape of the map in Fig. 2a that is achieved by applying the DE is a result of the regularization. The duration it takes the PC[4] to run the SE, DE, and JE each, which were implemented in Matlab® (R2018a) is listed in Table 1. The value for method X was determined as follows:

$$\frac{1}{2491} \sum_{k=10}^{2500} \frac{\text{time to perform } k\text{-th recursion for X}}{\text{time to perform } k\text{-th recursion for SE}} .$$

6. CONCLUSION

Two approaches for real-time estimation of parameter maps were investigated that aim to preserve the operating point dependency of parameters, which falls into oblivion for common "constant-parameter" identification or estimation schemes when operating points change. Therefore, the unknown parameter map is approximated via interpolants. The fixed-sized grid vectors of which are

[4] Intel® Core™ i7-7700K (4.2 GHz), 32 GB RAM, Windows® 10 (64 Bit)

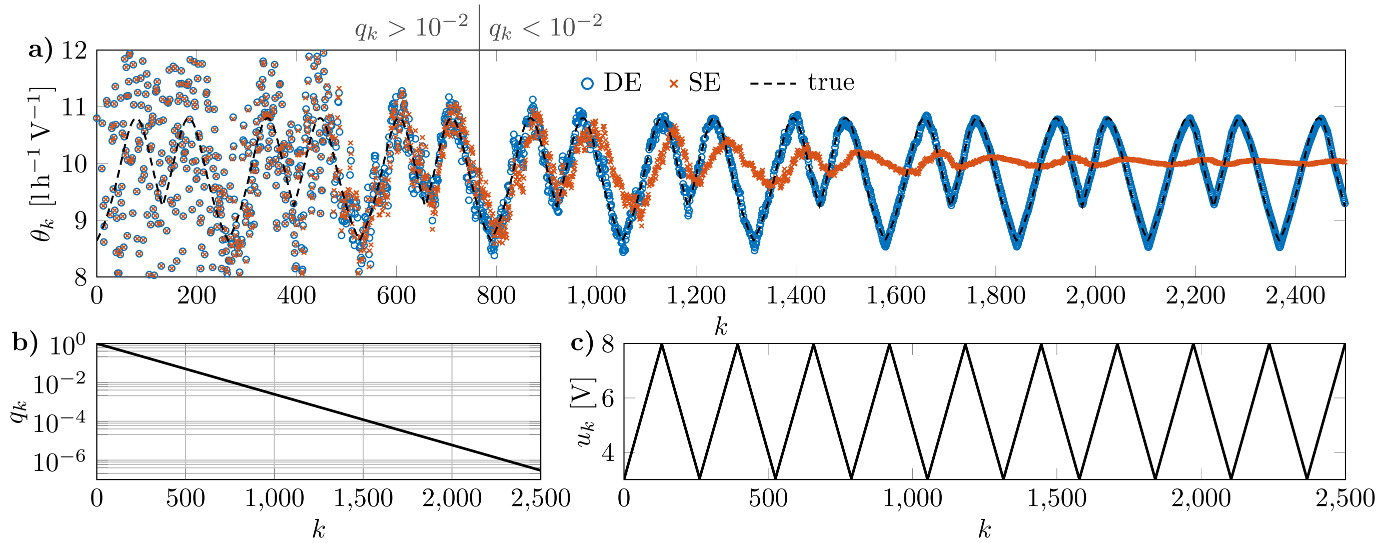

Fig. 3. *Standard real-time parameter Estimation* (SE) and real-time estimation applying the *Dual Estimation* (DE) of parameter and parameter map with a linear interpolant ($n_\mathrm{m} = 10$); a) true (dashed) and estimated (marked by symbols) parameter evolution, b) applied time-variant noise gain, c) applied system inputs

adjusted when measurements arise in order to reduce the approximation error. The method proposed requires a linear interpolant structure, which can be achieved easily even for cubic Hermite interpolants, as shown in Section 3.2. Höckerdal et al. (2011) presented the joint estimation of n_x dynamic system states and n_m grid points of a linear interpolant within the *Extended Kalman Filter*, and thus, the method's computational complexity is of order $\mathcal{O}((n_x + n_\mathrm{m})^3)$ if solely one parameter map is to be estimated.

Here, an alternative approach is introduced that performs the estimation of parameter and parameter map in sequence. The map adaption is essentially a *Recursive Least Squares* algorithm, whereby *regularization* terms are incorporated, for which ready-to-use formulae were presented for univariate parameter maps. This dual estimation scheme is of order $\mathcal{O}(n_x^3)$ if solely one parameter map is to be estimated, as is the case for the standard state and parameter estimation within a joint Kalman Filter approach if a single parameter is to be estimated.

Via numerical simulation it could be shown that both advanced approaches are capable of outperforming the standard parameter estimation. Although univariate parameter maps have been investigated in this paper exclusively, the dual estimation has already been validated with bilinear interpolants for bivariate dependencies. For the application concerned, dual estimation was verified with real data, for which the sole assessment measure is plausibility.

While the joint estimation approach is well suited to track time-variant parameter maps, dual estimation is adapted to (online) identification of time-invariant maps. Countermeasures may be investigated to keep the map flexible even in a converged state in order to enable tracking time-variant maps (system aging). While global (exponential) forgetting, the common approach for the *Recursive Least Squares* scheme, is inappropriate for the given application, a proper adapted version of local forgetting schemes, see (Hägglund (1985)) for instance, may serve this purpose.

ACKNOWLEDGEMENTS

This work was supported by MAN Energy Solutions SE and the Federal Ministry for Economic Affairs and Energy based on a decision by the German Bundestag as part of the ECOFLEX-Turbo project [grant number 03ET7091T].

REFERENCES

Höckerdal, E., Frisk, E., and Eriksson, L. (2011). EKF-based adaptation of look-up tables with an air mass-flow sensor application. *Control Engineering Practice*, 19(5), 442 – 453. doi:10.1016/j.conengprac.2011.01.006.

Hägglund, T. (1985). Recursive estimation of slowly time-varying parameters. *IFAC Proceedings Volumes*, 18(5), 1137 – 1142. doi:10.1016/S1474-6670(17)60715-8.

Jiang, J. and Zhang, Y. (2004). A revisit to block and recursive least squares for parameter estimation. *Computers & Electrical Engineering*, 30(5), 403 – 416. doi:10.1016/j.compeleceng.2004.05.002.

Julier, S.J., Uhlmann, J.K., and Durrant-Whyte, H.F. (2000). A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Transactions on Automatic Control*, 45(3), 477–482. doi:10.1109/9.847726.

Kahaner, D., Moler, C., and Nash, S. (1989). *Numerical methods and software*. Prentice Hall.

Kolås, S., Foss, B., and Schei, T. (2009). Constrained nonlinear state estimation based on the UKF approach. *Computers & Chemical Engineering*, 33(8), 1386 – 1401. doi:10.1016/j.compchemeng.2009.01.012.

Ljung, L. (1999). *System Identification: Theory for the User*, volume 2. Prentice Hall PTR.

Van der Merwe, R. and Wan, E.A. (2001). The Square-Root Unscented Kalman Filter for state and parameter-estimation. In *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 6, 3461–3464. doi:10.1109/ICASSP.2001.940586.