F3

Faculty of Electrical Engineering
Department of Control Engineering

Doctoral Thesis

State estimation and fault detection with reduced error sensitivity to parameters

Ing. Jaroslav Tabaček

Ph.D. Programme: Electrical Engineering and Information Technology
Branch of study: Control Engineering and Robotics

March 2023
Supervisor: prof. Ing. Vladimír Havlena, CSc.

Acknowledgement / Declaration

I would like to express my deepest gratitude to my supervisor *Prof. Vladimír Havlena* for his willingness to share his extensive academic and industry experience and his invaluable patience, guidance, and endless optimism.

Many thanks to *ČVUT*, *GAČR*, and *TAČR* for partially funding my research and enabling me to participate in conferences all over the world. Thanks should also go to *Honeywell* for giving me an opportunity to experience corporate research and culture. I would like to extend my sincere thanks to *UCEEB ČVUT* for facilitating the final stage of my study.

I am also grateful to *Jaroslav Pekař* and *Daniel Pachner* for suggesting and persuading me to apply for a Ph.D. and introducing me to industrial research in control theory. I would like to extend my sincere thanks to all my former colleagues at *Honeywell Prague Laboratory* for all the fun and interesting discussions. It was a great honor to work with such ingenious people. Special thanks to my peers *Pavel Otta* and *Jiří Dostál* for making this journey easier and more enjoyable.

This endeavor would not have been possible without my family and friends, who supported me during difficult times. Finally, words cannot express my gratitude to my wife, *Jarka*, for her unconditional love, wisdom, motivation, and patience.

I hereby declare that this doctoral thesis is my own personal effort and I have carefully cited all the literature sources I have used.

.....

Abstrakt / Abstract

Táto dizertačná práca rozširuje teóriu filtrovania so zníženou citlivosťou, tzv. desenzitizované filtrovanie. Desenzitizované filtrovanie je efektívny prístup k odhadovaniu stavu systémov s neurčitými parametrami. Stochastický prístup k redukcii citlivosti, ktorý je popísaný v tejto práci vedie k algoritmu presného desenzitizovaného Kalmanového filtru bez použitia plne odôvodnených predpokladov. Na základe tohto výsledku sú odvodené ďalšie varianty filtru, ktoré sú vhodné pre špecifické prípady použitia. Stochastický prístup taktiež umožňuje priamočiaru analýzu stability odhadu. V tejto práci je desenzitizovaný filter použitý aj spolu s metódou interagujúcich modelov. Touto kombináciou vzniká algoritmus pre detekciu a diagnostiku, ktorý funguje aj so zjednodušenými modelmi. Ďalšie užitočné využitie desenzitizovaného Kalmanového filtru je v algoritmoch pre distribuovaný odhad stavu. V tomto prípade pomáha zlepšiť lokálny odhad tým, že zahrňuje neurčitosť susedných odhadov bez toho aby zvýšil komunikačné zaťaženie siete. Tento prístup je tiež použitý na vytvorenie metódy na distribuovanú detekciu a diagnostiku chýb, ktorá dokáže odhadovať lokálne ale aj globálne chyby v systéme.

Kľúčové slová: odhad stavu, detekcia chýb, znecitlivené filtrovanie, neurčité parametre, distribuované systémy

This thesis extends the theory of desensitized filtering. Desensitized filtering is an efficient approach to state estimation for systems with uncertain parameters. The stochastic approach to sensitivity reduction developed in this thesis leads to the exact desensitized Kalman filter (XDKF) without using assumptions that are not fully justified. Based on this result, several variations of the XDKF are introduced for specific purposes. The stochastic approach allows providing a straightforward way to conduct stability analysis. The XDKF is also used with the interacting multiple model method, which results in fault detection and diagnosis (FDD) algorithms that work with simplified models. Another useful application of the XDKF was found in the distributed state estimation algorithms, where it helps to improve local estimation by considering neighbor estimate uncertainty without increased communication burden. This distributed approach is also used for developing a distributed FDD method, which can detect and diagnose local and global faults.

Keywords: state estimation, fault detection, desensitized filtering, uncertain parameters, distributed systems

Contents /

Chapter 1
Introduction

Advanced control methods are based on a mathematical model of a controlled system described by input, output, and state variables. The evolution of state variables in time captures the entire system behavior. Various feedback control methods require information about the system state, described by the state variables at a given time. In most cases, the state is not or cannot be measured directly. Therefore, the state needs to be estimated. Even when measurements are available, state estimation can provide a smoother signal without outliers, enabling better control performance. The state-of-the-art method for state estimation is the Kalman filter (KF) [1]. The KF is the optimal estimator if a state-space model used in the KF exactly corresponds to a real system.

Parameters in state-space models are often uncertain as a consequence of creating models that significantly simplify the structure of detailed models of real systems. Nevertheless, for control purposes, their behavior still sufficiently approximates the behavior of real systems. Models with uncertain parameters can be used in conventional state estimation algorithms without uncertainty information, but the optimality and stability of these algorithms cannot be guaranteed. Therefore, special algorithms are used for the state estimation of systems where the uncertainty of parameters is considered.

1.1 State estimation of systems with uncertain parameters

Many engineers have solved state estimation of systems with modeling errors in the past. The Kalman filter (KF) was among the first algorithms that could estimate the state of such systems efficiently. Successful applications in trajectory planning and navigation helped the KF to gain popularity. The initial achievements of the KF were nicely summed up by Kalman's colleague Schmidt in [2]. As he explained, the issues of systems with uncertain parameters were researched since the beginning, and several solutions were proposed.

Reformulating the uncertain parameters as state variables can solve the problem. If the parameters are not additive, this operation transforms a linear system into a nonlinear one. The state vector augmented by the parameters is then estimated using the extended Kalman filter (EKF) [3]. This way, the uncertainty is considered, and the parameter vector is estimated. This method is not robust when the EKF updates the parameter vector to an infeasible value because the linearization at the infeasible operating point might crash the algorithm. Another downside is that updating the parameter statistics is not desirable when the correct parameter statistics is known a priori.

The next option is to use the Schmidt-Kalman filter (SKF) [4, 3]. At its beginnings, this filter used to be referred to as the Kalman-Schmidt filter [2]. However, nowadays, the name Schmidt-Kalman filter is regularly used. Similarly to the EKF, this method

augments the state vector by the parameters. In this case, the Kalman filter is used in a reduced state form. That is, the parameter vector statistics is not updated, but used for updating the state statistics. Later, the same algorithm was derived by Woodbury [5] using the 'consider' analysis, called the consider Kalman filter (CKF). Several similar reduced-order Kalman filters were developed, and their summary can be found in [6].

The class of observers called the unknown input observers [7] can also be used for the state estimation of the systems with uncertain parameters. As the name suggests, this algorithm models the uncertain parameters as additional unknown disturbances of a system. The estimation error of these methods converges to zero regardless of the unknown input in the system. These methods are mostly used in fault detection algorithms [8].

Later, a robust filtering approach to the state estimation of uncertain systems became popular. These methods are designed for state estimation under the worst-case scenario based on norm-bounded parametric uncertainty [9–11]. More modern methods solve robust H_2 and H_∞ filtering for systems with norm-bounded or polytopic uncertainty using the LMI approach [12–14]. The robust Kalman filter for stochastic systems with correlated noises was also developed [15]. A good overview of the robust methods can be found in [16]. The performance of these methods depends on fulfilling the assumptions of parametric uncertainty. If they are not met, the filter performance is not guaranteed. Therefore, these methods are difficult to use when the parametric uncertainty cannot be determined or differs from the types assumed by the published methods.

More recently, the desensitized Kalman filter (DKF) was developed [17]. The idea for the DKF comes from the desensitized optimal control [18–19]. The method is based on augmenting the standard Kalman filter criterion by weighted state estimate sensitivity to unknown parameters. The goal is to balance the minimum-variance state estimation and the state estimation with minimal state estimate sensitivity to unknown parameters. The trade-off between objectives is controlled by weights for the particular objective. The parametric uncertainty does not need to be specified. Only the expected value of the parameters needs to be provided. Unfortunately, the DKF is not optimal due to the assumptions used for algorithm derivation. The DKF was included in Desensitized Optimal Filtering and Sensor Fusion Tool Kit [20] funded by NASA. The computational complexity of the DKF is significantly higher than the standard KF because the gain is obtained by computing an implicit algebraic equation at each step. This issue was solved by a special case of the DKF (SDKF) proposed in [21], also called the fast DKF [22], where an explicit formula expresses the gain. The limiting factor of the SDKF compared to the DKF is that weights on the state estimation error sensitivity to the parameter are the same for all states. Published papers related to the DKF are mostly application-oriented, focusing on applications to the Mars-entry navigation [23–25]. The desensitized approach was also applied to more complex filtering methods, such as the unscented DKF [26], the divided difference filter [27], the cubature DKF [28], and the ensemble DKF [29].

1.2 Fault detection and diagnosis

Faults in a process can result in increased economic cost, out-of-specification products, and even negative environmental impacts. Of course, these faults are not desirable. Therefore, companies push for developing detection and diagnosis systems to remain competitive and comply with economic and environmental demands [30].

Many industries, if not all, require the detection of faults. Research and development in this area is done by engineers with various backgrounds, which can easily create confusion when it comes to using basic terminology. The terminology was established by the IFAC SAFEPROCESS Technical Comitee [31]. First of all, the fault should not be mistaken for failure. The main difference is that the system can operate when a fault occurs but stops when a failure occurs. The *fault detection* is defined as a simple task with two possible outcomes — fault occurs or not. This task is usually followed by the *fault isolation*, which means determining the kind, place, and time of the fault. These two terms create together the acronym FDI — Fault Detection and Isolation. Furthermore, the FDI can be extended by the *fault identification*, which represents determining the size and behavior of the fault, which extends the acronym to the FDII. Also, the *fault diagnosis* term can be used as a replacement for the fault isolation and identification which changes the acronym to the FDD — Fault Detection and Diagnosis.

In general, the FDD approaches can be divided into signal-based [32], qualitative model-based [33], and quantitative model-based methods [34]. The signal-based methods have no prior knowledge of the process. Qualitative model-based or knowledge-based methods use machine learning techniques to detect faults as the unusual behavior of a system. The quantitative model-based methods require a mathematical model of a system.

The control community is mostly focused on the quantitative model-based FDD methods. Many great books and reviews of this research area are available, including [8, 30, 34–37]. These methods detect faults by analyzing the residuals generated using measurements and analytic models. The main methods for residual generation can be divided into observer-based methods, unknown input observers, parity relations approach, optimization-based approach, and Kalman filter-based approach. Huang's review of these methods was published in [37].

Generally, the fault detection and diagnosis problem can be solved by state estimation of a hybrid stochastic system. The goal of hybrid state estimation is to simultaneously estimate the mode and state of the system. In the case of linear hybrid systems such as Markov jump linear systems [38], the state and mode estimation is a multiple-model problem. The multiple-model adaptive estimation (MMAE) approach [39] is a Kalman filter-based approach that runs a bank of parallel Kalman filters where each filter represents the particular system mode (nominal mode or fault). However, this solution to a multiple-model problem requires implementing a fast-growing number of Kalman filters. On the other hand, the interacting multiple-model (IMM) filter [40] is an approximate method that solves the multiple-model problem efficiently when the number of system modes is low and the system is not very complex. The IMM is a popular estimator for systems where the models with Markovian jumps are used (e.g., navigation and tracking problems [41]). However, it is also used in fault detection and diagnostics [42–43] using the modes to describe the nominal and faulty behavior of the system. In such a configuration, the IMM computes the occurrence probability of each mode determined by the likelihood. Then it is possible to make a quantified decision on the fault occurrence and identify the probable source of the fault based on the likelihood values.

1.3 Distributed algorithms

Typical applications of distributed state estimation algorithms are large networks of intelligent edge devices which can gather information and estimate the local state independently. They need to operate only locally with fractional knowledge about the

other subsystems in the network, so their computational demand is low compared to the centralized system. If done correctly, such distributed networks can also be more secure since the failure of any individual node does not propagate to the entire system. Distributed state estimation is usually defined as a problem where distributed nodes share local information to estimate the state of an entire network. The most common solution is a consensus strategy introduced by Olfati-Saber [44]. This method aims to find a consensus on the full state of a network. Another strategy is information diffusion, introduced by Cattivelli and Sayed [45], which is motivated by practical applications where the consensus between nodes is not a priority. However, the full state of the network is still the objective of estimation. Estimation of the full state of a network is impractical in large-scale networks, and the priority is to estimate a local state. State estimation methods for large-scale networks are usually based on partitioning the network system into subsystems and estimating their local state using local Kalman filters (LKF) with additional information obtained from the closest neighbors. These methods need to compromise between the state estimation accuracy and communication burden. Among the methods with a low communication burden is the distributed Kalman filter for a network of interconnected linear systems via locally measured outputs by Marelli et al. [46], where a local system state can be estimated by sharing only the measured outputs. Roshany-Yamchi et al. [47] used the local Kalman filter (LKF) for distributed predictive control with an even lower communication burden since only the state estimates need to be shared. A method with high state estimation accuracy but a large communication burden is the partition-based Kalman filter (PKF) introduced in [48]. It includes the uncertainty information of neighbor estimates using state estimation error covariances. The PKF requires sharing the state estimates and the corresponding error covariances. Sharing the error covariances between nodes increases the size of transferred data related to the state estimation information from $\mathcal{O}(n)$ to $\mathcal{O}(n^2)$, where n is the dimension of the state vector.

The distributed model-based fault detection methods for large-scale systems are based on analyzing the local residuals and detecting a threshold breach. Boem et al. published various papers related to distributed fault detection and isolation, including [49], where the residuals are generated using local Luenberger estimators. Moreover, the detection threshold computation considers the uncertainty of neighbor estimates. In [50], stochastic systems are considered, the residuals are generated using a consensus-like method, and the uncertainty of neighbors is included in the detection threshold computation. In [51], a similar technique is applied to nonlinear systems. The distributed IMM algorithms are based on fusion [52], consensus [53], or diffusion [54], which are all impractical for application to large-scale systems.

Chapter 2
Motivation

In most real applications of Kalman filtering, simplified models are used. The Kalman filter loses its optimality with simplified models, and its performance can be improved. One of the easiest methods for improving the performance in such cases is to include the uncertainty information in the model structure and, consequently, into a filtering algorithm. For example, a first-order dynamical system is often chosen as a system model in industrial applications. The usual development process starts with fitting the model gain and time constant to the real system and then tuning the Kalman filter noise covariances to achieve satisfactory performance. The uncertainty in parameters arises necessarily during the model fitting process, but it is not considered in the filtering even if it is quantifiable. When using filtering algorithms that are robust to parameters, the uncertainty information from the fitting process is included in the filtering. Depending on the filtering algorithm, the uncertainty is expressed as a parameter interval, a probability distribution, or a weight. Quantifying the uncertainty is difficult, and is often transformed into tuning the uncertainty to achieve the desired filtering performance. In the case of the SKF and the robust H_2 and H_∞ filtering methods, this tuning can lead to counterintuitive parameter variance or parameter interval values. On the other hand, the desensitized Kalman filter (DKF) is tuned by setting weights that represent confidence in parameter accuracy, where zero means accurate and infinity means inaccurate. This intuitive tuning is the main advantage of the desensitized filtering approach and is why it is suitable for applications where the uncertainty of the parameter is difficult to quantify. The DKF is a promising, intuitive, and robust filtering algorithm. Its main drawback is that it did not attract interest among researchers. Therefore, it needs a thorough theoretical analysis. The main topics for research are the assumption of zero gain sensitivity to parameters, the stability of the algorithm, and the connection between stability and weight settings. The fault detection and diagnosis methods based on filtering methods can equally benefit from the desensitized approach. Furthermore, the benefits of the desensitized approach for considering uncertainty in the currently popular distributed systems should also be studied. This thesis aims to contribute to the desensitized filtering theory by solving the abovementioned challenges.

Chapter 3
Desensitized state estimation of linear systems

This chapter summarizes the state estimation algorithms with reduced sensitivity to parameters for linear systems. Most of the results in this chapter were originally published in [C2, J1, J2].

3.1 Algorithms for systems with noise correlation

The original desensitized Kalman filter (DKF) [17] was derived for linear stochastic systems with uncertain parameters and uncorrelated process and measurement noise. This section shows the derivation of desensitized filters suitable for linear systems with uncertain parameters and noise correlation, which were proposed in [C2]. Two filtering algorithms are derived. One uses the sensitivity definition and optimization criterion from the desensitized Kalman filter [17], The other uses the assumptions of the special case of desensitized Kalman filter with analytical gain [21] (SDKF), also called the fast desensitized Kalman filter [22] (FDKF). Both are single-step filtering algorithms compared to the DKF and SDKF, which are two-step filtering algorithms.

3.1.1 Linear system with noise correlation

The algorithms assume the linear stochastic system in the state space form

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A}(\boldsymbol{\theta})\mathbf{x}_k + \mathbf{B}(\boldsymbol{\theta})\mathbf{u}_k + \mathbf{v}_k, \\ \mathbf{y}_k &= \mathbf{C}(\boldsymbol{\theta})\mathbf{x}_k + \mathbf{D}(\boldsymbol{\theta})\mathbf{u}_k + \mathbf{e}_k, \end{aligned} \tag{3.1}$$

where $\mathbf{x}_k$ is the state vector of dimension N_x, $\mathbf{u}_k$ is the input vector of dimension N_u, $\mathbf{y}_k$ is the measurement vector of dimension N_y, $\boldsymbol{\theta}$ is the parameter vector of dimension N_θ, $\mathbf{A}(\boldsymbol{\theta})$ is $N_x \times N_x$ matrix, $\mathbf{B}(\boldsymbol{\theta})$ is $N_x \times N_u$ matrix, $\mathbf{C}(\boldsymbol{\theta})$ is $N_y \times N_x$ matrix and $\mathbf{D}(\boldsymbol{\theta})$ is $N_y \times N_y$ matrix. The process noise $\mathbf{v}_k$ and the measurement noise $\mathbf{e}_k$ are white noise sequences with samples from a normal distribution

$$\begin{bmatrix} \mathbf{v}_k \\ \mathbf{e}_k \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{Q} & \mathbf{S} \\ \mathbf{S}^T & \mathbf{R} \end{bmatrix} \right). \tag{3.2}$$

The non-zero cross-covariance matrix $\mathbf{S}$ states the correlation between the process and measurement noise. Also, both $\mathbf{v}_k$ and $\mathbf{e}_k$ are independent of the current state

$$\operatorname{cov}(\mathbf{v}_k, \mathbf{x}_k) = \operatorname{cov}(\mathbf{e}_k, \mathbf{x}_k) = 0. \tag{3.3}$$

When the observer is designed using some estimated value $\hat{\boldsymbol{\theta}}$, the true value is defined as

$$\boldsymbol{\theta} = \hat{\boldsymbol{\theta}} + \tilde{\boldsymbol{\theta}}, \tag{3.4}$$

where $\tilde{\boldsymbol{\theta}}$ denotes the difference between the true and estimated parameter vector, which is assumed to be unknown. Notations

$$\begin{aligned} \hat{\mathbf{A}} &\equiv \mathbf{A}(\hat{\boldsymbol{\theta}}),\ \hat{\mathbf{B}} \equiv \mathbf{B}(\hat{\boldsymbol{\theta}}), \\ \hat{\mathbf{C}} &\equiv \mathbf{C}(\hat{\boldsymbol{\theta}}),\ \hat{\mathbf{D}} \equiv \mathbf{D}(\hat{\boldsymbol{\theta}}), \end{aligned} \tag{3.5}$$

will be used to keep the text compact.

The correlation between the measurement and process noise is often neglected in developing models and estimation algorithms. However, the cost of adding such information is low, while it can significantly improve state estimation accuracy. For example, the noise correlation must be considered in the models created by discretization of a continuous-time system with discrete-time measurements. When a control algorithm is developed for such a system, it should use the latest information about the system state to increase the accuracy of the control. Therefore, the asynchronous sampling of the control input and the measurements is used. Such sampling leads to the discretization of the process and measurement equation, which creates noise correlation in the system.

3.1.2 Desensitized Kalman filter

The single-step Kalman filter equations for correlated measurement and process noise estimate the state vector optimally in terms of minimizing the error covariance matrix. The update equations in Joseph's form are defined as follows

$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{A}\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}\mathbf{u}_k + \mathbf{K}_k\left(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1} - \mathbf{D}\mathbf{u}_k\right), \tag{3.6}$$

$$\mathbf{P}_{k+1|k} = (\mathbf{A} - \mathbf{K}_k\mathbf{C})\mathbf{P}_{k|k-1}(\mathbf{A} - \mathbf{K}_k\mathbf{C})^T + \mathbf{Q} + \mathbf{K}_k\mathbf{R}\mathbf{K}_k^T - \mathbf{S}\mathbf{K}_k^T - \mathbf{K}_k\mathbf{S}^T, \tag{3.7}$$

where

$$\hat{\mathbf{x}}_{k+1|k} = \mathbb{E}\left[\mathbf{x}_{k+1}\,\middle|\,\mathcal{D}^k\right] \tag{3.8}$$

represents the conditional mean of the estimated state vector at time $k+1$ given all available data $\mathcal{D}^k = \{\mathbf{u}_1, \ldots \mathbf{u}_k, \mathbf{y}_1, \ldots \mathbf{y}_k\}$ up to time k. The matrix $\mathbf{P}_{k+1|k}$ denotes the state estimation error covariance, and it follows the state estimate notation

$$\mathbf{P}_{k+1|k} = \mathbb{E}\left[\left(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k}\right)\left(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k}\right)^T\right]. \tag{3.9}$$

The conditional index notation changes the definitions of state and covariance analogically with the definitions above

$$\hat{\mathbf{x}}_{k|k-1} = \mathbb{E}\left[\mathbf{x}_k\,\middle|\,\mathcal{D}^{k-1}\right], \tag{3.10}$$

$$\mathbf{P}_{k|k-1} = \mathbb{E}\left[\left(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}\right)\left(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}\right)^T\right]. \tag{3.11}$$

The matrix $\mathbf{K}_k$ represents the optimal Kalman gain at time k and it is obtained as

$$\mathbf{K}_k = \left(\mathbf{A}\mathbf{P}_{k|k-1}\mathbf{C}^T + \mathbf{S}\right)\left(\mathbf{C}\mathbf{P}_{k|k-1}\mathbf{C}^T + \mathbf{R}\right)^{-1}. \tag{3.12}$$

The covariance update (3.7) keeps the covariance positive semi-definite, which is proved as follows.

Proof. The joint probability distribution of the process and measurement noise is described by (3.2). A covariance matrix is always positive semi-definite. Therefore, the covariance of noise distribution is also positive semi-definite

$$\begin{bmatrix} \mathbf{Q} & \mathbf{S} \\ \mathbf{S}^T & \mathbf{R} \end{bmatrix} \geq 0. \tag{3.13}$$

Then the following matrix is also positive semi-definite [55]

$$\begin{bmatrix} \mathbf{I}_x & -\mathbf{K}_k \end{bmatrix} \begin{bmatrix} \mathbf{Q} & \mathbf{S} \\ \mathbf{S}^T & \mathbf{R} \end{bmatrix} \begin{bmatrix} \mathbf{I}_x \\ -\mathbf{K}_k^T \end{bmatrix} = \mathbf{Q} - \mathbf{K}_k\mathbf{S}^T - \mathbf{S}\mathbf{K}_k^T + \mathbf{K}_k\mathbf{R}\mathbf{K}_k^T \geq 0, \tag{3.14}$$

where $\mathbf{I}_x$ is the identity matrix of size N_x. The covariance update is defined in (3.7). It is defined as the sum of the positive semi-definite matrix (3.14) and the matrix $(\mathbf{A} - \mathbf{K}_k\mathbf{C})\mathbf{P}_{k|k-1}(\mathbf{A} - \mathbf{K}_k\mathbf{C})^T$ which is a positive semi-definite matrix. Then the entire update is the sum of positive semi-definite matrices, which results in a positive semi-definite matrix. ∎

The desensitized Kalman filter for system (3.1) is based on the updates (3.6) and (3.7) where the matrices with estimated parameters (3.5) are used

$$\hat{\mathbf{x}}_{k+1|k} = \hat{\mathbf{A}}\hat{\mathbf{x}}_{k|k-1} + \hat{\mathbf{B}}\mathbf{u}_k + \mathbf{K}_k\left(\mathbf{y} - \hat{\mathbf{C}}\hat{\mathbf{x}}_{k|k-1} - \hat{\mathbf{D}}\mathbf{u}_k\right), \tag{3.15}$$

$$\mathbf{P}_{k+1|k} = (\hat{\mathbf{A}} - \mathbf{K}_k\hat{\mathbf{C}})\mathbf{P}_{k|k-1}(\hat{\mathbf{A}} - \mathbf{K}_k\hat{\mathbf{C}})^T + \mathbf{Q} + \mathbf{K}_k\mathbf{R}\mathbf{K}_k^T - \mathbf{S}\mathbf{K}_k^T - \mathbf{K}_k\mathbf{S}^T, \tag{3.16}$$

and $\mathbf{K}_k$ is derived from a modified optimality criterion. The derivation starts with the sensitivity definition. The state error sensitivity to a particular uncertain parameter is defined as stated in [17]

$$\boldsymbol{\sigma}_{p,k+1|k} = \frac{d\tilde{\mathbf{x}}_{k+1|k}}{d\hat{\theta}_p} = \frac{d\left(\hat{\mathbf{x}}_{k+1|k} - \mathbf{x}_{k+1}\right)}{d\hat{\theta}_p} = \frac{d\hat{\mathbf{x}}_{k+1|k}}{d\hat{\theta}_p} = \boldsymbol{\xi}_{p,k} - \mathbf{K}_k\boldsymbol{\gamma}_{p,k}, \tag{3.17}$$

where the index p denotes the particular parameter in the parameter vector and

$$\boldsymbol{\xi}_{p,k} = \hat{\mathbf{A}}\boldsymbol{\sigma}_{p,k|k-1} + \frac{\partial\hat{\mathbf{A}}}{\partial\hat{\theta}_p}\hat{\mathbf{x}}_{k|k-1} + \frac{\partial\hat{\mathbf{B}}}{\partial\hat{\theta}_p}\mathbf{u}_k, \tag{3.18}$$

$$\boldsymbol{\gamma}_{p,k} = \hat{\mathbf{C}}\boldsymbol{\sigma}_{p,k|k-1} + \frac{\partial\hat{\mathbf{C}}}{\partial\hat{\theta}_p}\hat{\mathbf{x}}_{k|k-1} + \frac{\partial\hat{\mathbf{D}}}{\partial\hat{\theta}_p}\mathbf{u}_k. \tag{3.19}$$

Notice that the sensitivity definition assumes that the true state value is not sensitive to the expected value of the parameter set in the model description. Then the state estimation error sensitivity to the parameter is reduced to the state estimate sensitivity to the parameter. On top of that, the assumption of the zero gain sensitivity is used, formulated as follows.

Assumption 3.1.

$$\text{For all } \hat{\theta}_p,\ p = 1, \ldots, N_\theta\text{:}\quad \frac{\partial\mathbf{K}_k}{\partial\hat{\theta}_p} = 0.$$

The goal of the derivation is to find the gain which has reduced sensitivity to the parameter error. If this goal is achieved, then the gain sensitivity is close to zero, which is approximated by Assumption 3.1. Also, Assumption 3.1 significantly simplifies the algorithm derivation.

The optimality criterion for the filtering problem is defined in the sense of desensitized filtering, which means there are two optimization objectives. The first objective is minimizing the trace of the estimation error covariance matrix. The second is the minimization of the weighted state error sensitivities to the parameter. The optimization criterion is defined as

$$\min_{\mathbf{K}_k} J(\mathbf{K}_k),\ J(\mathbf{K}_k) = \operatorname{tr}\left(\mathbf{P}_{k+1|k}\right) + \sum_{p=1}^{N_\theta}\left(\boldsymbol{\sigma}_{p,k+1|k}^T\mathbf{W}_p\boldsymbol{\sigma}_{p,k+1|k}\right), \tag{3.20}$$

where J is the cost function, tr is the matrix trace operator, $\mathbf{W}_p$ are symmetric weighting matrices representing tuning parameters between objectives in the optimality criterion. If $\mathbf{W}_p$ is large, then the emphasis is put on reducing the sensitivity to the parameter θ_p. On the other hand, if the weight matrix is small, then the minimum mean square error is prioritized. If all $\mathbf{W}_p$ are set to zero, then the algorithm becomes the standard Kalman filter.

To find the gain $\mathbf{K}_k$, the following equation need to be solved

$$\frac{\partial J(\mathbf{K}_k)}{\partial \mathbf{K}_k} = 0. \tag{3.21}$$

First, the partial derivative of the cost is obtained as

$$\begin{aligned}\frac{\partial J(\mathbf{K}_k)}{\partial \mathbf{K}_k} = -2\left(\hat{\mathbf{A}}\mathbf{P}_{k|k-1}\hat{\mathbf{C}}^T + \mathbf{S} + \sum_{p=1}^{N_\theta}\left(\mathbf{W}_p\boldsymbol{\xi}_{p,k}\boldsymbol{\gamma}_{p,k}^T\right)\right) + \\ + 2\mathbf{K}_k\left(\hat{\mathbf{C}}\mathbf{P}_{k|k-1}\hat{\mathbf{C}}^T + \mathbf{R}\right) + 2\sum_{p=1}^{N_\theta}\left(\mathbf{W}_p\mathbf{K}_k\boldsymbol{\gamma}_{p,k}\boldsymbol{\gamma}_{p,k}^T\right).\end{aligned} \tag{3.22}$$

Then the optimal gain is the solution of the equation

$$\begin{aligned}\mathbf{K}_k\left(\hat{\mathbf{C}}\mathbf{P}_{k|k-1}\hat{\mathbf{C}}^T + \mathbf{R}\right) + \sum_{p=1}^{N_\theta}\left(\mathbf{W}_p\mathbf{K}_k\boldsymbol{\gamma}_{p,k}\boldsymbol{\gamma}_{p,k}^T\right) = \\ = \hat{\mathbf{A}}\mathbf{P}_{k|k-1}\hat{\mathbf{C}}^T + \mathbf{S} + \sum_{p=1}^{N_\theta}\left(\mathbf{W}_p\boldsymbol{\xi}_{p,k}\boldsymbol{\gamma}_{p,k}^T\right).\end{aligned} \tag{3.23}$$

The derived algorithm consists of the update equations (3.15), (3.16) and (3.17) where $\mathbf{K}_k$ is the solution of (3.23). The algorithm is summed up in Algorithm 3.1.

Algorithm 3.1 (DKF for linear systems with noise correlation — DKF-NC).

- Evaluate model sensitivities

$$\begin{aligned}\boldsymbol{\xi}_{p,k} &= \hat{\mathbf{A}}\boldsymbol{\sigma}_{p,k|k-1} + \frac{\partial\hat{\mathbf{A}}}{\partial\hat{\theta}_p}\hat{\mathbf{x}}_{k|k-1} + \frac{\partial\hat{\mathbf{B}}}{\partial\hat{\theta}_p}\mathbf{u}_k,\\ \boldsymbol{\gamma}_{p,k} &= \hat{\mathbf{C}}\boldsymbol{\sigma}_{p,k|k-1} + \frac{\partial\hat{\mathbf{C}}}{\partial\hat{\theta}_p}\hat{\mathbf{x}}_{k|k-1} + \frac{\partial\hat{\mathbf{D}}}{\partial\hat{\theta}_p}\mathbf{u}_k.\end{aligned}$$

- Solve for $\mathbf{K}_k$

$$\begin{aligned}\mathbf{K}_k\left(\hat{\mathbf{C}}\mathbf{P}_{k|k-1}\hat{\mathbf{C}}^T + \mathbf{R}\right) + \sum_{p=1}^{N_\theta}\left(\mathbf{W}_p\mathbf{K}_k\boldsymbol{\gamma}_{p,k}\boldsymbol{\gamma}_{p,k}^T\right) = \\ = \hat{\mathbf{A}}\mathbf{P}_{k|k-1}\hat{\mathbf{C}}^T + \mathbf{S} + \sum_{p=1}^{N_\theta}\left(\mathbf{W}_p\boldsymbol{\xi}_{p,k}\boldsymbol{\gamma}_{p,k}^T\right).\end{aligned}$$

- Update the state, sensitivity, and covariance

$$\begin{aligned}\boldsymbol{\sigma}_{p,k+1|k} &= \boldsymbol{\xi}_{p,k} - \mathbf{K}_k\boldsymbol{\gamma}_{p,k},\\ \hat{\mathbf{x}}_{k+1|k} &= \hat{\mathbf{A}}\hat{\mathbf{x}}_{k|k-1} + \hat{\mathbf{B}}\mathbf{u}_k + \mathbf{K}_k\left(\mathbf{y} - \hat{\mathbf{C}}\hat{\mathbf{x}}_{k|k-1} - \hat{\mathbf{D}}\mathbf{u}_k\right),\\ \mathbf{P}_{k+1|k} &= (\hat{\mathbf{A}} - \mathbf{K}_k\hat{\mathbf{C}})\mathbf{P}_{k|k-1}(\hat{\mathbf{A}} - \mathbf{K}_k\hat{\mathbf{C}})^T + \mathbf{Q} + \mathbf{K}_k\mathbf{R}\mathbf{K}_k^T - \mathbf{S}\mathbf{K}_k^T - \mathbf{K}_k\mathbf{S}^T.\end{aligned}$$

3.1.3 Analytical gain

The main issue of the DKF-NC is that it requires solving the implicit equation at each update step, creating a significant computational burden. The root cause of this issue is a form of sensitivity definition and cost definition. The issue with an implicit gain equation can be overcome by using a modified objective for sensitivity reduction as proposed in [21]. First, sensitivity vectors (3.17) are joined into a sensitivity matrix

$$\boldsymbol{\Sigma}_{k+1|k} = \frac{d\tilde{\mathbf{x}}_{k+1|k}}{d\hat{\boldsymbol{\theta}}} = \frac{d\left(\hat{\mathbf{x}}_{k+1|k} - \mathbf{x}_{k+1}\right)}{d\hat{\boldsymbol{\theta}}} = \frac{d\hat{\mathbf{x}}_{k+1|k}}{d\hat{\boldsymbol{\theta}}} = \boldsymbol{\Xi}_k - \mathbf{K}_k\boldsymbol{\Gamma}_k, \tag{3.24}$$

where

$$\boldsymbol{\Xi}_k = \hat{\mathbf{A}}\boldsymbol{\Sigma}_{k|k-1} + \frac{\partial\left(\hat{\mathbf{A}}\hat{\mathbf{x}}_{k|k-1} + \hat{\mathbf{B}}\mathbf{u}_k\right)}{\partial\hat{\boldsymbol{\theta}}}, \tag{3.25}$$

$$\boldsymbol{\Gamma}_k = \hat{\mathbf{C}}\boldsymbol{\Sigma}_{k|k-1} + \frac{\partial\left(\hat{\mathbf{C}}\hat{\mathbf{x}}_{k|k-1} + \hat{\mathbf{D}}\mathbf{u}_k\right)}{\partial\hat{\boldsymbol{\theta}}}. \tag{3.26}$$

The derivation is also simplified using Assumption 3.1. The objective for sensitivity reduction is modified to the minimization of the trace of the weighted sensitivity matrix

$$\min_{\mathbf{K}_k} J(\mathbf{K}_k), \; J(\mathbf{K}_k) = \operatorname{tr}\left(\mathbf{P}_{k+1|k}\right) + \operatorname{tr}\left(\boldsymbol{\Sigma}_{k+1|k}\mathbf{W}\boldsymbol{\Sigma}_{k+1|k}^T\right). \tag{3.27}$$

Remark 3.1. The weighting matrices in (3.20) are the symmetric matrices $\mathbf{W}_p \in \mathbb{R}^{N_x \times N_x}$, where N_x is the size of the state vector. On the other hand, the weighting matrix in (3.27) is the symmetric matrix $\mathbf{W} \in \mathbb{R}^{N_\theta \times N_\theta}$, where N_θ is the size of the parameter vector. The weighting in the DKF-NC can be tuned more accurately since it is possible to weight each parameter for each state separately. The weights in the SDKF-NC set the same weights of the parameter vector for all states.

The partial derivative of the criterion with respect to the gain is obtained as

$$\begin{aligned}\frac{\partial J(\mathbf{K}_k)}{\partial \mathbf{K}_k} = &-2\left(\hat{\mathbf{A}}\mathbf{P}_{k|k-1}\hat{\mathbf{C}}^T + \mathbf{S} + \boldsymbol{\Xi}_k\mathbf{W}\boldsymbol{\Gamma}_k^T\right) \\ &+ 2\mathbf{K}_k\left(\hat{\mathbf{C}}\mathbf{P}_{k|k-1}\hat{\mathbf{C}}^T + \mathbf{R} + \boldsymbol{\Gamma}_k\mathbf{W}\boldsymbol{\Gamma}_k^T\right).\end{aligned} \tag{3.28}$$

The optimal gain is the solution of the equation, which is created by putting the above derivative equal to zero. Then the optimal gain is obtained by the explicit formula

$$\mathbf{K}_k = \left(\hat{\mathbf{A}}\mathbf{P}_{k|k-1}\hat{\mathbf{C}}^T + \mathbf{S} + \boldsymbol{\Xi}_k\mathbf{W}\boldsymbol{\Gamma}_k^T\right)\left(\hat{\mathbf{C}}\mathbf{P}_{k|k-1}\hat{\mathbf{C}}^T + \mathbf{R} + \boldsymbol{\Gamma}_k\mathbf{W}\boldsymbol{\Gamma}_k^T\right)^{-1}. \tag{3.29}$$

The covariance update equation can be further modified by substituting the optimal gain defined in (3.29) into (3.7). After the substitution, the alternative form of the covariance update equation is obtained as follows

$$\mathbf{P}_{k+1|k} = \hat{\mathbf{A}}\mathbf{P}_{k|k-1}\hat{\mathbf{A}}^T + \mathbf{Q} - \mathbf{K}_k\left(\hat{\mathbf{C}}\mathbf{P}_{k|k-1}\hat{\mathbf{A}}^T + \mathbf{S}^T\right) + \boldsymbol{\Sigma}_{k+1|k}\mathbf{W}\boldsymbol{\Gamma}_k^T\mathbf{K}_k^T. \tag{3.30}$$

The formulas (3.24) and (3.29) can be substituted in (3.30) to clarify that the alternative covariance update results in a symmetric matrix. Then the obtained formulation is a symmetric matrix

$$\begin{aligned}\mathbf{P}_{k+1|k} = &\hat{\mathbf{A}}\mathbf{P}_{k|k-1}\hat{\mathbf{A}}^T + \mathbf{Q} - \left(\hat{\mathbf{A}}\mathbf{P}_{k|k-1}\hat{\mathbf{C}}^T + \mathbf{S}\right)\mathbf{P}_y^{-1}\left(\hat{\mathbf{C}}\mathbf{P}_{k|k-1}\hat{\mathbf{A}}^T + \mathbf{S}^T\right) - \\ &- \mathbf{K}_k\boldsymbol{\Gamma}_k\mathbf{W}\boldsymbol{\Gamma}_k^T\mathbf{K}_k^T + \left(\boldsymbol{\Xi}_k\mathbf{W}\boldsymbol{\Gamma}_k^T\right)\mathbf{P}_y^{-1}\left(\boldsymbol{\Xi}_k\mathbf{W}\boldsymbol{\Gamma}_k^T\right)^T,\end{aligned} \tag{3.31}$$

where

$$\mathbf{P}_y = \hat{\mathbf{C}}\mathbf{P}_{k|k-1}\hat{\mathbf{C}}^T + \mathbf{R} + \boldsymbol{\Gamma}_k\mathbf{W}\boldsymbol{\Gamma}_k^T. \tag{3.32}$$

The special case of DKF-NC (SDKF-NC) algorithm consists of the update equations (3.15), (3.16) (or (3.30)) and (3.24) where $\mathbf{K}_k$ is defined in (3.29). This algorithm updates the statistics with explicit formulas. Therefore, it is more efficient and faster than the DKF-NC. The algorithm is summed up in Algorithm 3.2.

Algorithm 3.2 (Special DKF for systems with noise correlation — SDKF-NC).

- Evaluate model sensitivities

$$\boldsymbol{\Xi}_k = \hat{\mathbf{A}}\boldsymbol{\Sigma}_{k|k-1} + \frac{\partial\left(\hat{\mathbf{A}}\hat{\mathbf{x}}_{k|k-1} + \hat{\mathbf{B}}\mathbf{u}_k\right)}{\partial\hat{\boldsymbol{\theta}}},$$
$$\boldsymbol{\Gamma}_k = \hat{\mathbf{C}}\boldsymbol{\Sigma}_{k|k-1} + \frac{\partial\left(\hat{\mathbf{C}}\hat{\mathbf{x}}_{k|k-1} + \hat{\mathbf{D}}\mathbf{u}_k\right)}{\partial\hat{\boldsymbol{\theta}}}.$$

- Compute the gain

$$\mathbf{K}_k = \left(\hat{\mathbf{A}}\mathbf{P}_{k|k-1}\hat{\mathbf{C}}^T + \mathbf{S} + \boldsymbol{\Xi}_k\mathbf{W}\boldsymbol{\Gamma}_k^T\right)\left(\hat{\mathbf{C}}\mathbf{P}_{k|k-1}\hat{\mathbf{C}}^T + \mathbf{R} + \boldsymbol{\Gamma}_k\mathbf{W}\boldsymbol{\Gamma}_k^T\right)^{-1}.$$

- Update the state, sensitivity and covariance

$$\begin{aligned}
\boldsymbol{\Sigma}_{k+1|k} &= \boldsymbol{\Xi}_k - \mathbf{K}_k\boldsymbol{\Gamma}_k,\\
\hat{\mathbf{x}}_{k+1|k} &= \hat{\mathbf{A}}\hat{\mathbf{x}}_{k|k-1} + \hat{\mathbf{B}}\mathbf{u}_k + \mathbf{K}_k\left(\mathbf{y} - \hat{\mathbf{C}}\hat{\mathbf{x}}_{k|k-1} - \hat{\mathbf{D}}\mathbf{u}_k\right),\\
\mathbf{P}_{k+1|k} &= (\hat{\mathbf{A}} - \mathbf{K}_k\hat{\mathbf{C}})\mathbf{P}_{k|k-1}(\hat{\mathbf{A}} - \mathbf{K}_k\hat{\mathbf{C}})^T + \mathbf{Q} + \mathbf{K}_k\mathbf{R}\mathbf{K}_k^T - \mathbf{S}\mathbf{K}_k^T - \mathbf{K}_k\mathbf{S}^T,\\
&\text{or}\\
\mathbf{P}_{k+1|k} &= \hat{\mathbf{A}}\mathbf{P}_{k|k-1}\hat{\mathbf{A}}^T + \mathbf{Q} - \mathbf{K}_k\left(\hat{\mathbf{C}}\mathbf{P}_{k|k-1}\hat{\mathbf{A}}^T + \mathbf{S}^T\right) + \boldsymbol{\Sigma}_{k+1|k}\mathbf{W}\boldsymbol{\Gamma}_k^T\mathbf{K}_k^T.
\end{aligned}$$

Remark 3.2. The DKF-NC and SDKF-NC are identical when applied to a first-order system with a single uncertain parameter.

3.1.4 Example

The importance of including cross-correlation information in filtering algorithms is discussed based on a simple example. The example for the performance testing is chosen to be a continuous-time system with discrete-time measurements. The reason for selecting such a system is that it is more natural to describe physical phenomena in the continuous-time domain. Also, the system output is measured by devices in the discrete-time domain. The continuous-discrete state estimators are available, but it is more practical to use the discrete state estimation, where a continuous system is discretized first.

In real applications, the same models are usually used for estimation and control at the same time. However, often the measurement and control input sampling is offset. This method is called asynchronous sampling, and it is used to pass the maximum amount of information to the control algorithm. The asynchronous sampling method is depicted in Fig. 3.1. The control input sampling period T_s is split by the parameter ε. The period $T_c = (1-\varepsilon)\,T_s$ represents the time needed for computation of the new input,

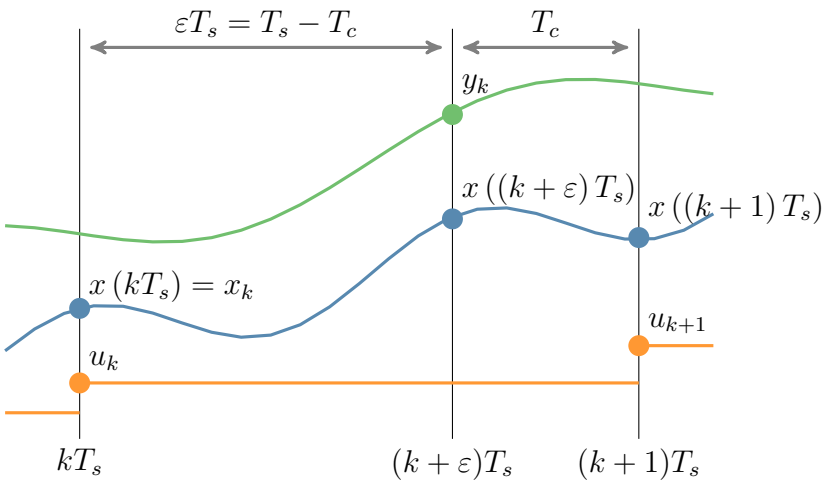

Figure 3.1. Asynchronous sampling of a system with control.

and it is usually minimized to supply the control algorithm with the latest information. Such sampling configuration causes a correlation between process and measurement noise based on a common process noise input in the period $(kT_s, (k+\varepsilon)\,T_s)$. This is a typical example of how the noise correlation is introduced to the system because of the sampling method.

The example used for performance tests is a simple first-order continuous system with an uncertain parameter. The continuous-discrete system description is

$$\begin{aligned} dx(t) &= \frac{1}{\theta}\left(u\left(t\right) - x\left(t\right)\right)dt + dw\left(t\right), \\ y(t) &= x\left(t\right) + e_{\mathrm{c}}\left(t\right), \end{aligned} \tag{3.33}$$

where θ is the uncertain parameter which represents the uncertain time constant of the first-order system. The noise statistics is defined as

$$dw\left(t\right) \sim \mathcal{N}\left(0, Q_{\mathrm{c}}\,dt\right), \quad e_{\mathrm{c}}\left(t\right) \sim \mathcal{N}\left(0, R_{\mathrm{c}}\right), \tag{3.34}$$

where dw is the Wiener process increment. The system is discretized with a sampling period $T_s = 0.5\,\mathrm{sec}$, and the measurements are asynchronous with $\varepsilon = 0.75$. The exact discretization [56] of the system results in the discrete-time system

$$\begin{aligned} x_{k+1} &= e^{\frac{-T_s}{\theta}} x_k + \left(\int_0^{T_s} e^{\frac{-t}{\theta}}\,dt\right)\frac{1}{\theta}u_k + v_k, \\ y_k &= e^{\frac{-\varepsilon T_s}{\theta}} x_k + \left(\int_0^{\varepsilon T_s} e^{\frac{-t}{\theta}}\,dt\right)\frac{1}{\theta}u_k + e_k, \end{aligned} \tag{3.35}$$

where the output y_k describes the measurement at time $(k+\varepsilon)T_s$, but the traditional notation with index k is used instead. The discretized process noise e_k and measurement noise v_k have statistics

$$\begin{bmatrix} v_k \\ e_k \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} Q_{T_s} & Q_{\varepsilon T_s} \\ Q_{\varepsilon T_s} & Q_{\varepsilon T_s} + R_c \end{bmatrix}\right), \tag{3.36}$$

where

$$Q_{T_s} = \int_0^{T_s} e^{\frac{-t}{\theta}} Q_c e^{\frac{-t}{\theta}}\,dt, \quad Q_{\varepsilon T_s} = \int_0^{\varepsilon T_s} e^{\frac{-t}{\theta}} Q_c e^{\frac{-t}{\theta}}\,dt. \tag{3.37}$$

The example is tested for various configurations of Q_c, R_c.

The SDKF-NC and DKF-NC are identical when applied to (3.35). Therefore, only the SDKF-NC is tested here, but its performance results also apply to the DKF-NC. The SDKF-NC is compared to the following state estimation algorithms in the example. The first is a perfect Kalman Filter (PKF) representing the theoretical case with information about true parameter value. The PKF is the standard KF in (3.6), (3.7) with the information about the true value of the parameter. Therefore, the PKF is the best estimator by the minimum mean square error criterion. The next algorithm is an imperfect Kalman Filter (IKF), the Kalman filter where the parameter is set to the expected value, which can differ from the true parameter value.

The algorithms are tested in the simulation, where the system responses to initial conditions and an input step. The initial input has amplitude 20 and then changes to 10. The initial system state has amplitude 10, and the initial state estimate is set to 0. The initial state estimation error variance is set to 10. The uncertain parameter θ in (3.33) is expected to be from interval $\theta \in [0.5, 9.5]$, with $\hat{\theta} = 5$ used as the parameter value in the model used in the SDKF-NC and IKF. The SDKF-NC is tested at weight settings 1, 9, and 16 to show the impact on the state estimation accuracy. Each test configuration is repeated in 100 Monte Carlo simulations, where the true value of the parameter is generated randomly from the defined interval. The performance was tested on scenarios with various process and measurement noise covariances of the simulated system. The estimation algorithms used the exactly discretized noise covariance matrices computed from the covariance matrix used in the simulation. The results of the Monte Carlo simulation for $Q_c = 0.01$, $R_c = 0.01$ are shown in Fig. 3.2. The figure nicely depicts the impact of uncertain time constant in the first-order system. The impact is high during the transient state, where the time constant plays a significant role. On the other hand, the uncertainty is low during the steady state.

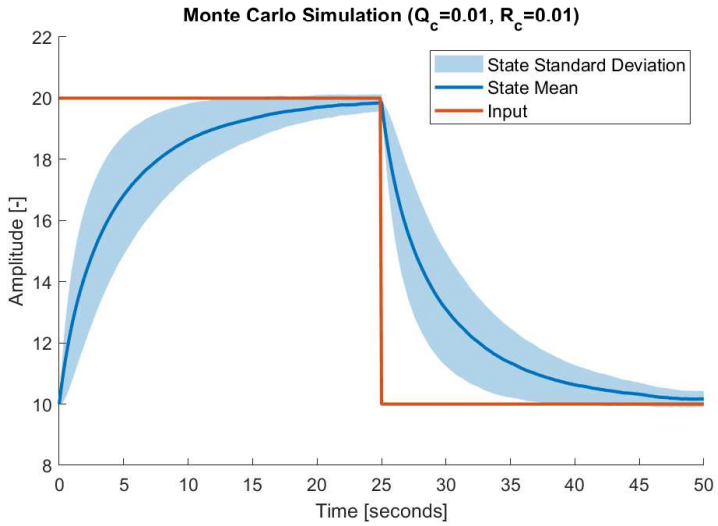

Figure 3.2. The figure shows the statistics from 100 Monte Carlo simulations for the scenario with noise statistics: $Q_c = 0.01, R_c = 0.01$. The input is the same for all simulations. The parameter value is generated from interval $[0.5, 9.5]$ for each simulation.

An ideal behavior of a robust Kalman filter for this example would be to increase the covariance matrix during the transient state due to higher uncertainty and decrease the covariance during the steady state. Put simply, the Kalman filter would rely more on the measurement information during the transient state and follow the model dy-

namics during the steady state. Fig. 3.3 depicts that the SDKF-NC behaves according to these notes. The variance of the SDKF-NC is increased during the transient state, and it decreases with decreasing impact of the uncertain parameter. In this example, the variance changes are larger with higher weight settings which confirms the desensitized approach. However, this is true only if the weight results in stable estimates. For very high weight, the minimum mean square error objective is overweighted with desensitizing objective, and the state estimation diverges.

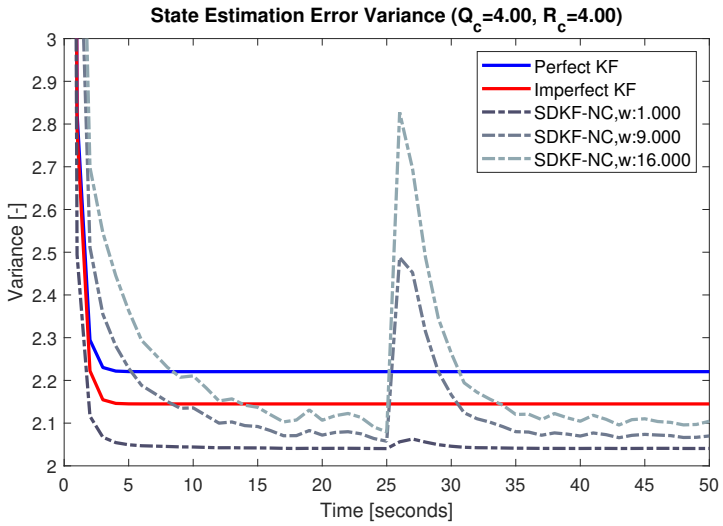

Figure 3.3. The figure shows the mean value of variance from 100 Monte Carlo simulations for the scenario with noise statistics: $Q_c = 4, R_c = 4$.

The Kalman filtering goal is to minimize variance, and the PKF, being the best estimator, should have the lowest possible variance. Therefore, someone might need clarification on why the PKF has a larger variance than the IKF in Fig. 3.3. The explanation is that variance updates are computed based on the model and they are independent of data. Then the low variance is optimal for some model but the higher variance might be optimal for the real system. Hence the lowest variance does not infer the optimality. In this example, the variance for the model with $\hat{\theta}$ is lower than the average variance for models with the true parameter values.

The accuracy test results are shown using the total root-mean-square error (RMSE) metric in Fig. 3.4. The RMSE is obtained with the formula

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(x_k - \hat{x}_{k|k-1}\right)^2}. \tag{3.38}$$

When the measurement noise covariance is small, the IKF performs comparably to the PKF because the algorithm follows the measurement information and does not rely on model dynamics too much. Therefore, there is no reason to use the SDKF-NC or other method robust to parameter uncertainty in such a situation. On the other hand, the SDKF-NC is valuable when the measurement noise is high, and the KF relies more on the model dynamics. The weight of SDKF-NC needs to be tuned to achieve the required performance. For example, if the goal is to achieve the lowest total RMSE, then $W = 9$ accomplishes it in this case.

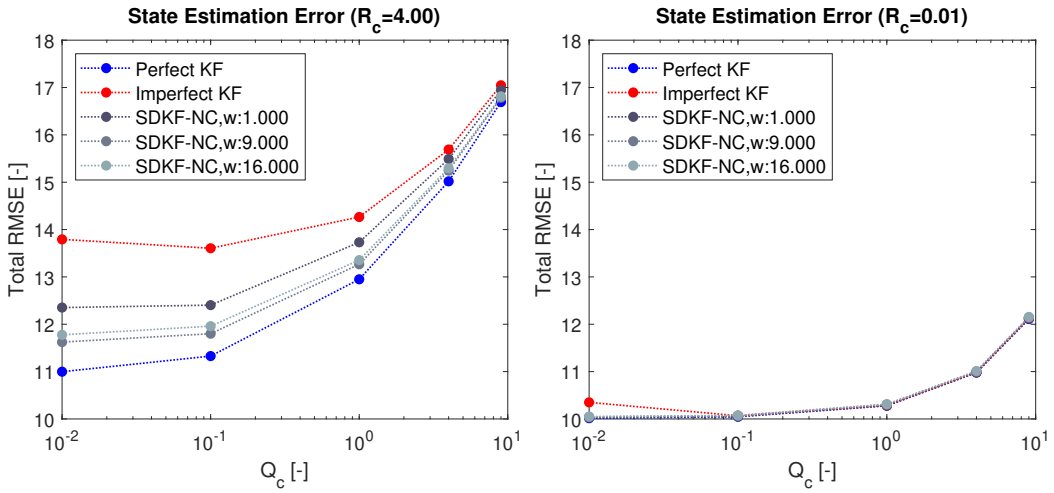

Figure 3.4. The figure shows the mean value of total RMSE from 100 Monte Carlo simulations for systems with various noise statistics.

3.2 Exact desensitized Kalman filter

This section uses a stochastic approach to reducing the sensitivity to derive the exact desensitized Kalman filter (XDKF) published in [J1]. The derivation of the XDKF does not require the assumption of zero gain sensitivity. On the other hand, by applying this assumption, the SDKF equivalent is obtained in a form corresponding to the Kalman filter with the time-varying correlated process and measurement noise. This form is used to propose a parametrized steady-state XDKF and, consequently, to define the stability conditions. The adaptive objective normalization technique achieves improved performance and more intuitive weight tuning.

3.2.1 Linear system

The system in this section is not as general as (3.1). Using the general system (3.1) to derive the XDKF leads to complicated equations. Instead, a system with uncertain parameters only in the state propagation equation and no noise correlation is used. This system allows a clear presentation of the impact of the stochastic sensitivity definition on the algorithm derivation.

Assume a discrete-time linear stochastic system which depends on a parameter vector $\boldsymbol{\theta}$ in the state propagation equation

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A}(\boldsymbol{\theta})\mathbf{x}_k + \mathbf{B}(\boldsymbol{\theta})\mathbf{u}_k + \mathbf{v}_k, \\ \mathbf{y}_k &= \mathbf{C}\mathbf{x}_k + \mathbf{e}_k, \end{aligned} \tag{3.39}$$

where $\mathbf{x}_k$ is the state vector of dimension N_x, $\mathbf{u}_k$ is the input vector of dimension N_u, $\mathbf{y}_k$ is the measurement vector of dimension N_y, $\boldsymbol{\theta}$ is the parameter vector of dimension N_θ, and $\mathbf{v}_k$ and $\mathbf{e}_k$ are white noise sequences with statistics

$$\begin{bmatrix} \mathbf{v}_k \\ \mathbf{e}_k \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{bmatrix} \right). \tag{3.40}$$

$\mathbf{A}(\boldsymbol{\theta})$ is $N_x \times N_x$ matrix, $\mathbf{B}(\boldsymbol{\theta})$ is $N_x \times N_u$ matrix and $\mathbf{C}$ is $N_y \times N_x$ matrix. It is possible to formulate the system with the output matrix $\mathbf{C}$ dependent on the parameter vector, but it results in an extensive derivation process. Assuming the parametric dependency

only in the state propagation equation keeps the algorithm derivation straightforward and compact. The matrices $\mathbf{A}(\boldsymbol{\theta})$, $\mathbf{B}(\boldsymbol{\theta})$ are assumed to be linear matrix functions of parameters

$$\mathbf{A}(\boldsymbol{\theta}) = \mathbf{A}_0 + \theta_1\mathbf{A}_1 + \theta_2\mathbf{A}_2 + \ldots + \theta_{N_\theta}\mathbf{A}_{N_\theta}, \tag{3.41}$$
$$\mathbf{B}(\boldsymbol{\theta}) = \mathbf{B}_0 + \theta_1\mathbf{B}_1 + \theta_2\mathbf{B}_2 + \ldots + \theta_{N_\theta}\mathbf{B}_{N_\theta}. \tag{3.42}$$

The linear matrix function descriptions in (3.41) and (3.42) are inspired by Taylor series expansion of a nonlinear function in the nominal point $\boldsymbol{\theta}_{\text{nom}}$

$$f(\boldsymbol{\theta}) = f(\boldsymbol{\theta}_{\text{nom}}) + (\partial f(\boldsymbol{\theta})/\partial\theta_1)\,|_{\boldsymbol{\theta}_{\text{nom}}}\Delta\theta_1 + (\partial f(\boldsymbol{\theta})/\partial\theta_2)\,|_{\boldsymbol{\theta}_{\text{nom}}}\Delta\theta_2 + \ldots \tag{3.43}$$

When the parameter value is known, a linear observer for the system (3.39) estimates the state as

$$\hat{\mathbf{x}}^*_{k+1|k} = \mathbf{A}(\boldsymbol{\theta})\hat{\mathbf{x}}^*_{k|k-1} + \mathbf{B}(\boldsymbol{\theta})\mathbf{u}_k + \mathbf{K}_k\left(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}^*_{k|k-1}\right), \tag{3.44}$$

and generates the state prediction error

$$\tilde{\mathbf{x}}^*_{k+1|k} = \mathbf{x}_{k+1} - \hat{\mathbf{x}}^*_{k+1|k} = (\mathbf{A}(\boldsymbol{\theta}) - \mathbf{K}_k\mathbf{C})\,\tilde{\mathbf{x}}^*_{k|k-1} + \mathbf{v}_k - \mathbf{K}_k\mathbf{e}_k, \tag{3.45}$$

where $\mathbf{K}_k$ is a time-variant observer gain. The exact definition of $\mathbf{K}_k$ is not important at this stage. The mean value of the observer error (3.45) is zero. Therefore, minimizing its covariance will result in the optimal observer in the minimum mean square error (MMSE) sense. The optimal observer gain in the MMSE sense is called the Kalman gain. Time index notations for observer gain will be omitted to keep the text compact.

When the observer is designed using some estimated value $\hat{\boldsymbol{\theta}}$, the true value is defined as

$$\boldsymbol{\theta} = \hat{\boldsymbol{\theta}} + \tilde{\boldsymbol{\theta}}, \tag{3.46}$$

where $\tilde{\boldsymbol{\theta}}$ denotes the parameter uncertainty, which is assumed to be unknown. The observer state prediction mean becomes

$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{A}(\hat{\boldsymbol{\theta}})\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}(\hat{\boldsymbol{\theta}})\mathbf{u}_k + \mathbf{K}_k\left(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1}\right), \tag{3.47}$$

which generates the prediction error

$$\begin{aligned}\tilde{\mathbf{x}}_{k+1|k} &= \mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k}, \\ &= \left(\mathbf{A}(\hat{\boldsymbol{\theta}}) - \mathbf{K}_k\mathbf{C}\right)\tilde{\mathbf{x}}_{k|k-1} + \mathbf{v}_k - \mathbf{K}_k\mathbf{e}_k + \tilde{\mathbf{A}}\mathbf{x}_k + \tilde{\mathbf{B}}\mathbf{u}_k,\end{aligned} \tag{3.48}$$

where the matrices $\tilde{\mathbf{A}}, \tilde{\mathbf{B}}$ are defined as

$$\tilde{\mathbf{A}} \equiv \mathbf{A}(\boldsymbol{\theta}) - \mathbf{A}(\hat{\boldsymbol{\theta}}),\ \tilde{\mathbf{B}} \equiv \mathbf{B}(\boldsymbol{\theta}) - \mathbf{B}(\hat{\boldsymbol{\theta}}). \tag{3.49}$$

Notations $\hat{\mathbf{A}} \equiv \mathbf{A}(\hat{\boldsymbol{\theta}})$ and $\hat{\mathbf{B}} \equiv \mathbf{B}(\hat{\boldsymbol{\theta}})$ will be used to keep the text compact.

Keeping the low sensitivity of prediction error to parameters guarantees that the parameter error $\tilde{\boldsymbol{\theta}}$ will have limited impact on $\tilde{\mathbf{x}}_{k+1|k}$. The sensitivity of (3.48) to a

parameter $\hat{\theta}_p$, $p = 1, \ldots, N_\theta$, is obtained as

$$\mathbf{s}_{p,k+1|k} \equiv \frac{d\tilde{\mathbf{x}}_{k+1|k}}{d\hat{\theta}_p}, \tag{3.50}$$

$$= \left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)\mathbf{s}_{p,k|k-1} + \left(\frac{\partial\hat{\mathbf{A}}}{\partial\hat{\theta}_p} - \frac{d\mathbf{K}}{d\hat{\theta}_p}\mathbf{C}\right)\tilde{\mathbf{x}}_{k|k-1} - \frac{d\mathbf{K}}{d\hat{\theta}_p}\mathbf{e}_k$$
$$- \frac{\partial\hat{\mathbf{A}}}{\partial\hat{\theta}_p}\mathbf{x}_k - \frac{\partial\hat{\mathbf{B}}}{\partial\hat{\theta}_p}\mathbf{u}_k, \tag{3.51}$$

$$= \left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)\mathbf{s}_{p,k|k-1} + \left(\mathbf{A}_p - \mathbf{K}_{\theta_p}\mathbf{C}\right)\tilde{\mathbf{x}}_{k|k-1} - \mathbf{K}_{\theta_p}\mathbf{e}_k$$
$$- \mathbf{A}_p\mathbf{x}_k - \mathbf{B}_p\mathbf{u}_k, \tag{3.52}$$

$$= \left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)\mathbf{s}_{p,k|k-1} - \mathbf{K}_{\theta_p}\mathbf{C}\tilde{\mathbf{x}}_{k|k-1} - \mathbf{K}_{\theta_p}\mathbf{e}_k$$
$$- \mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} - \mathbf{B}_p\mathbf{u}_k, \tag{3.53}$$

where $\mathbf{A}_p, \mathbf{B}_p$ are the matrix coefficients of linear matrix functions defined in (3.41) and (3.42), and

$$\mathbf{K}_{\theta_p} \equiv \frac{d\mathbf{K}}{d\hat{\theta}_p}. \tag{3.54}$$

The sensitivity (3.50) is intentionally computed as a derivative with respect to parameter value $\hat{\theta}_p$ rather than a derivative to a true value θ_p with approximation in the parameter $\hat{\theta}_p$

$$\left.\frac{d\tilde{\mathbf{x}}_{k+1|k}}{d\theta_p}\right|_{\theta_p=\hat{\theta}_p}. \tag{3.55}$$

For example, take a state estimation where $\hat{\theta}_p = \theta_p$. In this case, the optimal state estimation solution is the standard KF, and reducing the sensitivity to the true parameter or any other parameter will only worsen the performance. Therefore, high sensitivity to a parameter does not generally cause high state estimation error. Using the sensitivity definition (3.55) would make sense if the error $\tilde{\mathbf{x}}_{k+1|k}$ would be defined using only the true parameter. Then it would approximate the sensitivity of the error to the real parameter in the selected parameter value. However, the state estimation error (3.48) explicitly defines the impact of the selected parameter $\hat{\theta}_p$. Then the reduction can target the sensitivity of the state estimation to the selected parameter while not affecting irrelevant high sensitivity to the true parameter.

Note that the real state $\mathbf{x}_k$ is independent of $\hat{\theta}_p$. This property is not used directly in the definition of the sensitivity (3.50) as it is done in the DKF. It is used only after evaluation of the prediction error derivative to eliminate the element $\tilde{\mathbf{A}}\left(\partial\mathbf{x}_k/\partial\hat{\theta}_p\right)$, which would otherwise appear in (3.53). This approach allows exploiting the stochastic properties of the filter, e.g., the nonzero mean of the prediction error sequence caused by a model bias. Consequently, the sensitivity is defined as the sensitivity of the true prediction error rather than the sensitivity of the Kalman filter mean estimate.

3.2.2 Optimal filter

The optimal Kalman gain can be derived by minimization of the mean square prediction error

$$\min_{\mathbf{K}_k} J(\mathbf{K}_k), \quad J(\mathbf{K}_k) = \operatorname{tr}\mathbb{E}\left[\tilde{\mathbf{x}}_{k+1|k}\tilde{\mathbf{x}}_{k+1|k}^T\right]. \tag{3.56}$$

However, using such criterion results in the estimator, which does not use information about the accuracy of model parameters. Model parameters are often uncertain or biased, which can be caused by model simplification, linearization, or identification from data. Incorporating information about the possible inaccuracy of parameters in state estimation algorithm design can be done in several ways. In this thesis, a state estimation method with dual objectives is proposed. First, the state estimation error is minimized as it is in the standard Kalman filter. Secondly, the state prediction error sensitivity to parameters is minimized. The sensitivity is a random variable with the second non-central moment

$$\mathbb{E}\left[\mathbf{s}_{p,k+1|k}\,\mathbf{s}_{p,k+1|k}^{T}\right] = \hat{\mathbf{s}}_{p,k+1|k}\,\hat{\mathbf{s}}_{p,k+1|k}^{T} + \operatorname{cov}\left(\mathbf{s}_{p,k+1|k}\right). \tag{3.57}$$

Minimizing only a trace of the sensitivity covariance (second central moment) would minimize the variations around the sensitivity mean. Therefore, the trace of the second non-central moment needs to be minimized to minimize the sensitivity. These objectives are achieved by minimizing the optimization criterion defined as a convex combination of second moments

$$\begin{aligned} \min_{\mathbf{K}_k} J(\mathbf{K}_k),\ J(\mathbf{K}_k) &= \alpha \operatorname{tr} \mathbf{P}_{k+1|k}^{xx} + \sum_{p} \gamma_p \operatorname{tr} \mathbf{P}_{k+1|k}^{p,ss}, \\ \alpha \equiv 1 - \sum_{p} \gamma_p,\ 0 \leq \sum_{p} \gamma_p &< 1,\ \text{and}\ \gamma_p \geq 0, \end{aligned} \tag{3.58}$$

where the second central moment of the state is denoted as

$$\mathbf{P}_{k+1|k}^{xx} \equiv \mathbb{E}\left[\tilde{\mathbf{x}}_{k+1|k}\,\tilde{\mathbf{x}}_{k+1|k}^{T}\right], \tag{3.59}$$

and the second non-central moment of the sensitivity is denoted as

$$\mathbf{P}_{k+1|k}^{p,ss} \equiv \mathbb{E}\left[\mathbf{s}_{p,k+1|k}\,\mathbf{s}_{p,k+1|k}^{T}\right]. \tag{3.60}$$

Notation in (3.60) is usually reserved for the second central moment. However, here it is used to denote both central and non-central moments, which helps to increase text readability. In the following text, the subscript k at gain $\mathbf{K}_k$ is omitted to keep the text lucid. The second moment of sensitivity can be evaluated as

$$\begin{aligned} \mathbf{P}_{k+1|k}^{p,ss} = &\left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)\mathbf{P}_{k|k-1}^{p,ss}\left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)^{T} + \left(-\mathbf{K}_{\theta_p}\mathbf{C}\right)\mathbf{P}_{k|k-1}^{xx}\left(-\mathbf{K}_{\theta_p}\mathbf{C}\right)^{T} \\ &+ \left(-\mathbf{K}_{\theta_p}\mathbf{C}\right)\mathbf{P}_{k|k-1}^{p,xs}\left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)^{T} + \left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)\mathbf{P}_{k|k-1}^{p,sx}\left(-\mathbf{K}_{\theta_p}\mathbf{C}\right)^{T} \\ &+ \mathbf{K}_{\theta_p}\mathbf{R}\mathbf{K}_{\theta_p}^{T} + \left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)\left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)^{T} \\ &- \left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)\hat{\mathbf{s}}_{p,k|k-1}^{T}\left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)^{T} \\ &- \left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)\hat{\mathbf{s}}_{p,k|k-1}\left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)^{T}, \end{aligned} \tag{3.61}$$

where the notation $\mathbf{P}_{k|k-1}^{p,sx} \equiv \mathbb{E}\left[\mathbf{s}_{p,k|k-1}\,\tilde{\mathbf{x}}_{k|k-1}^{T},\right]$ $\mathbf{P}_{k|k-1}^{p,xs} \equiv \mathbb{E}\left[\tilde{\mathbf{x}}_{k|k-1}\,\mathbf{s}_{p,k|k-1}^{T}\right]$ is used. The rest of the second moments are evaluated as

$$\mathbf{P}_{k+1|k}^{xx} = \left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)\mathbf{P}_{k|k-1}^{xx}\left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)^{T} + \mathbf{K}\mathbf{R}\mathbf{K}^{T} + \mathbf{Q}, \tag{3.62}$$

$$\mathbf{P}_{k+1|k}^{p,xs} = \left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)\mathbf{P}_{k|k-1}^{p,xs}\left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)^{T} - \left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)\mathbf{P}_{k|k-1}^{xx}\mathbf{C}^{T}\mathbf{K}_{\theta_p}^{T} + \mathbf{K}\mathbf{R}\mathbf{K}_{\theta_p}^{T}, \tag{3.63}$$

$$\mathbf{P}_{k+1|k}^{p,sx} = \left(\mathbf{P}_{k+1|k}^{p,xs}\right)^{T}. \tag{3.64}$$

The optimal gain $\mathbf{K}$ can be found by solving the equation

$$\frac{\partial J}{\partial \mathbf{K}} = 0. \tag{3.65}$$

The derivatives of the criterion objectives required to evaluate (3.65) are

$$\frac{\partial \operatorname{tr} \mathbf{P}^{xx}_{k+1|k}}{\partial \mathbf{K}} = -2\hat{\mathbf{A}}\mathbf{P}^{xx}_{k|k-1}\mathbf{C}^T + 2\mathbf{K}\left(\mathbf{C}\mathbf{P}^{xx}_{k|k-1}\mathbf{C}^T + \mathbf{R}\right), \tag{3.66}$$

$$\begin{aligned}\frac{\partial \operatorname{tr} \mathbf{P}^{p,ss}_{k+1|k}}{\partial \mathbf{K}} &= -2\hat{\mathbf{A}}\mathbf{P}^{p,ss}_{k|k-1}\mathbf{C}^T + 2\mathbf{K}_{\theta_p}\mathbf{C}\mathbf{P}^{p,xs}_{k|k-1}\mathbf{C}^T + 2\mathbf{K}\mathbf{C}\mathbf{P}^{p,ss}_{k|k-1}\mathbf{C}^T \\ &\quad + 2\left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)\hat{\mathbf{s}}^T_{p,k|k-1}\mathbf{C}^T.\end{aligned} \tag{3.67}$$

The resulting equation for optimal gain is obtained by substituting (3.66) and (3.67) to (3.65)

$$\begin{aligned}&\mathbf{K}\left[\alpha\left(\mathbf{C}\mathbf{P}^{xx}_{k|k-1}\mathbf{C}^T + \mathbf{R}\right) + \sum_p \gamma_p \mathbf{C}\mathbf{P}^{p,ss}_{k|k-1}\mathbf{C}^T\right] + \sum_p \gamma_p \mathbf{K}_{\theta_p}\mathbf{C}\mathbf{P}^{p,xs}_{k|k-1}\mathbf{C}^T = \\ &= \alpha\left(\hat{\mathbf{A}}\mathbf{P}^{xx}_{k|k-1}\mathbf{C}^T\right) + \sum_p \gamma_p\left[\hat{\mathbf{A}}\mathbf{P}^{p,ss}_{k|k-1}\mathbf{C}^T - \left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)\hat{\mathbf{s}}^T_{p,k|k-1}\mathbf{C}^T\right].\end{aligned} \tag{3.68}$$

Since the multiplier of the gain $\mathbf{K}$ in (3.68) is always a positive definite matrix, the optimal gain can be expressed as

$$\begin{aligned}\mathbf{K} = &\left[\hat{\mathbf{A}}\left(\alpha\mathbf{P}^{xx}_{k|k-1} + \sum_p \gamma_p \mathbf{P}^{p,ss}_{k|k-1}\right)\mathbf{C}^T + \mathbf{S}_k\right] \\ &\times\left[\mathbf{C}\left(\alpha\mathbf{P}^{xx}_{k|k-1} + \sum_p \gamma_p \mathbf{P}^{p,ss}_{k|k-1}\right)\mathbf{C}^T + \alpha\mathbf{R}\right]^{-1},\end{aligned} \tag{3.69}$$

where

$$\mathbf{S}_k = -\sum_p \gamma_p\left[\mathbf{K}_{\theta_p}\mathbf{C}\mathbf{P}^{p,xs}_{k|k-1}\mathbf{C}^T + \left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)\hat{\mathbf{s}}^T_{p,k|k-1}\mathbf{C}^T\right]. \tag{3.70}$$

The optimal gain definition (3.69) is an implicit equation

$$\mathbf{K} = \mathbf{f}_K\left(\frac{d\mathbf{K}}{d\hat{\theta}_p}\right). \tag{3.71}$$

The solution of (3.71) can be found by a fixed-point iteration method. First, an implicit matrix function is obtained by calculating the derivative of $\mathbf{K}$ [1]

$$\frac{d\mathbf{K}}{d\hat{\theta}_p} = \frac{\partial \mathbf{K}}{\partial \hat{\theta}_p} + \sum_{i=1}^{N_x}\sum_{j=1}^{N_x}\frac{\partial \mathbf{K}}{\partial \hat{A}_{ij}}\frac{d\hat{A}_{ij}}{d\hat{\theta}_p}, \tag{3.72}$$

[1] Note that if there were uncertain parameters in the output equation, additional elements $\partial\hat{C}_{ij}/\partial\hat{\theta}_p$ would appear in (3.72). Such generalization also leads to a result, but the derivation process is extensive.

where $\hat{A}_{ij}$ is an element of $\hat{\mathbf{A}}$. The partial derivatives of prior second moments with respect to the parameter are considered zero. The derivative in (3.72) can be evaluated as

$$\begin{aligned}\frac{d\mathbf{K}}{d\hat{\theta}_p} = &-\gamma_p \frac{\partial}{\partial\hat{\theta}_p}\frac{d\mathbf{K}}{d\hat{\theta}_p}\mathbf{C}\mathbf{P}^{p,xs}_{k|k-1}\mathbf{C}^T\left[\mathbf{C}\left(\alpha\mathbf{P}^{xx}_{k|k-1}+\sum_p \gamma_p\mathbf{P}^{p,ss}_{k|k-1}\right)\mathbf{C}^T+\alpha\mathbf{R}\right]^{-1}\\ &+\sum_{i=1}^{N_x}\sum_{j=1}^{N_x}\left[\mathbf{T}^A_{ij}\left(\alpha\mathbf{P}^{xx}_{k|k-1}+\sum_p \gamma_p\mathbf{P}^{p,ss}_{k|k-1}\right)\mathbf{C}^T\right]\\ &\times\left[\mathbf{C}\left(\alpha\mathbf{P}^{xx}_{k|k-1}+\sum_p \gamma_p\mathbf{P}^{p,ss}_{k|k-1}\right)\mathbf{C}^T+\alpha\mathbf{R}\right]^{-1}A_{p,ij},\end{aligned} \tag{3.73}$$

and it can be modified to

$$\begin{aligned}\frac{d\mathbf{K}}{d\hat{\theta}_p} = &\left[\mathbf{A}_p\left(\alpha\mathbf{P}^{xx}_{k|k-1}+\sum_p \gamma_p\mathbf{P}^{p,ss}_{k|k-1}\right)\mathbf{C}^T-\gamma_p\frac{\partial}{\partial\hat{\theta}_p}\frac{d\mathbf{K}}{d\hat{\theta}_p}\mathbf{C}\mathbf{P}^{p,xs}_{k|k-1}\mathbf{C}^T\right]\\ &\times\left[\mathbf{C}\left(\alpha\mathbf{P}^{xx}_{k|k-1}+\sum_p \gamma_p\mathbf{P}^{p,ss}_{k|k-1}\right)\mathbf{C}^T+\alpha\mathbf{R}\right]^{-1},\end{aligned} \tag{3.74}$$

where $A_{p,ij}$ is the element of $\mathbf{A}_p$ from the definition of $\mathbf{A}$ in (3.41) and $\mathbf{T}^A_{ij} \equiv \partial\hat{\mathbf{A}}/\partial\hat{A}_{ij}$ is the structure matrix of $\hat{\mathbf{A}}$ [57]. Here, a general structure of $\hat{\mathbf{A}}$ is assumed, so $\mathbf{T}^A_{ij}$ is a single-entry matrix. The implicit equation (3.73) can be shortened to

$$\mathbf{K}_{\theta_p} = \mathbf{f}_{K_\theta}\left(\mathbf{K}_{\theta_p}\right), \tag{3.75}$$

and modified into an iterated matrix function form

$$\mathbf{K}^{(n+1)}_{\theta_p} = \mathbf{f}_{K_\theta}\left(\mathbf{K}^{(n)}_{\theta_p}\right), \tag{3.76}$$

where n is the iteration number. The fixed-point solution $\mathbf{K}_{\theta_p}$ can be found if the sequence $\mathbf{K}^{(0)}_{\theta_p}, \mathbf{K}^{(1)}_{\theta_p}, \mathbf{K}^{(2)}_{\theta_p}, \ldots$ converges for an arbitrary selected $\mathbf{K}^{(0)}_{\theta_p}$. Let's start with selecting $\mathbf{K}^{(0)}_{\theta_p}$ such that $\mathbf{K}^{(0)}_{\theta_p} \neq \mathbf{0}$ and the partial derivative $\partial\mathbf{K}^{(0)}_{\theta_p}/\partial\hat{\theta}_p = 0$, i.e., $\mathbf{K}^{(0)}_{\theta_p}$ is nonzero and independent of $\hat{\theta}_p$. Then the first iteration is evaluated as

$$\begin{aligned}\mathbf{K}^{(1)}_{\theta_p} = &\left[\mathbf{A}_p\left(\alpha\mathbf{P}^{xx}_{k|k-1}+\sum_p \gamma_p\mathbf{P}^{p,ss}_{k|k-1}\right)\mathbf{C}^T\right]\\ &\times\left[\mathbf{C}\left(\alpha\mathbf{P}^{xx}_{k|k-1}+\sum_p \gamma_p\mathbf{P}^{p,ss}_{k|k-1}\right)\mathbf{C}^T+\alpha\mathbf{R}\right]^{-1}.\end{aligned} \tag{3.77}$$

Notice that $\mathbf{K}^{(1)}_{\theta_p}$ does not depend on $\mathbf{K}^{(0)}_{\theta_p}$ nor $\hat{\theta}_p$. Then, the second iteration $\mathbf{K}^{(2)}_{\theta_p}$ and all that follows will be identical to $\mathbf{K}^{(1)}_{\theta_p}$. Therefore, the fixed point solution is found in a single iteration. The fixed point solution of (3.75), i.e., the optimal gain sensitivity, is given by (3.77).

Since the formula in (3.77) does not depend on $\mathbf{K}^{(n)}_{\theta_p}$ from the previous iterations, the resulting state estimation algorithm does not need to repeat the iteration process, and the gain sensitivities $\mathbf{K}_{\theta_p}$ can be calculated explicitly. The derived algorithm is denoted exact desensitized Kalman filter (XDKF), which is summed up in Algorithm 3.3.

Algorithm 3.3 (eXact Desensitized Kalman Filter — XDKF).

- Given γ_p and prior estimates $\hat{\mathbf{x}}_{k|k-1}, \mathbf{P}^{xx}_{k|k-1}, \hat{\mathbf{s}}_{p,k|k-1}, \mathbf{P}^{p,ss}_{k|k-1}, \mathbf{P}^{p,sx}_{k|k-1}$.
- Compute the gain sensitivity $\mathbf{K}_{\theta_p}$ for all $p = 1, \ldots, N_\theta$ using (3.77).
- Compute the optimal gain $\mathbf{K}$ using (3.69).
- Update the state, sensitivity, and second moments

$$
\begin{aligned}
\hat{\mathbf{x}}_{k+1|k} &= \hat{\mathbf{A}}\hat{\mathbf{x}}_{k|k-1} + \hat{\mathbf{B}}\mathbf{u}_k + \mathbf{K}\left(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1}\right), \\
\hat{\mathbf{s}}_{p,k+1|k} &= \left(\hat{\mathbf{A}} - \mathbf{KC}\right)\hat{\mathbf{s}}_{p,k|k-1} - \mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} - \mathbf{B}_p\mathbf{u}_k - \mathbf{K}_{\theta_p}\left(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1}\right), \\
\mathbf{P}^{xx}_{k+1|k} &= \left(\hat{\mathbf{A}} - \mathbf{KC}\right)\mathbf{P}^{xx}_{k|k-1}\left(\hat{\mathbf{A}} - \mathbf{KC}\right)^T + \mathbf{KRK}^T + \mathbf{Q}, \\
\mathbf{P}^{p,ss}_{k+1|k} &= \left(\hat{\mathbf{A}} - \mathbf{KC}\right)\mathbf{P}^{p,ss}_{k|k-1}\left(\hat{\mathbf{A}} - \mathbf{KC}\right)^T + \left(-\mathbf{K}_{\theta_p}\mathbf{C}\right)\mathbf{P}^{xx}_{k|k-1}\left(-\mathbf{K}_{\theta_p}\mathbf{C}\right)^T \\
&\quad + \left(-\mathbf{K}_{\theta_p}\mathbf{C}\right)\mathbf{P}^{p,xs}_{k|k-1}\left(\hat{\mathbf{A}} - \mathbf{KC}\right)^T + \left(\hat{\mathbf{A}} - \mathbf{KC}\right)\mathbf{P}^{p,sx}_{k|k-1}\left(-\mathbf{K}_{\theta_p}\mathbf{C}\right)^T \\
&\quad + \mathbf{K}_{\theta_p}\mathbf{R}\mathbf{K}^T_{\theta_p} + \left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)\left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)^T \\
&\quad - \left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)\hat{\mathbf{s}}^T_{p,k|k-1}\left(\hat{\mathbf{A}} - \mathbf{KC}\right)^T \\
&\quad - \left(\hat{\mathbf{A}} - \mathbf{KC}\right)\hat{\mathbf{s}}_{p,k|k-1}\left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)^T, \\
\mathbf{P}^{p,xs}_{k+1|k} &= \left(\hat{\mathbf{A}} - \mathbf{KC}\right)\mathbf{P}^{p,xs}_{k|k-1}\left(\hat{\mathbf{A}} - \mathbf{KC}\right)^T - \left(\hat{\mathbf{A}} - \mathbf{KC}\right)\mathbf{P}^{xx}_{k|k-1}\mathbf{C}^T\mathbf{K}^T_{\theta_p} + \mathbf{KRK}^T_{\theta_p}, \\
\mathbf{P}^{p,sx}_{k+1|k} &= \left(\hat{\mathbf{A}} - \mathbf{KC}\right)\mathbf{P}^{p,sx}_{k|k-1}\left(\hat{\mathbf{A}} - \mathbf{KC}\right)^T - \mathbf{K}_{\theta_p}\mathbf{C}\mathbf{P}^{xx}_{k|k-1}\left(\hat{\mathbf{A}} - \mathbf{KC}\right)^T + \mathbf{K}_{\theta_p}\mathbf{R}\mathbf{K}^T.
\end{aligned}
$$

3.2.3 Suboptimal filters

This section introduces suboptimal solutions focused on implementation complexity in real applications. Presented methods are easier to implement, and some have lower computational burden than the optimal XDKF. Furthermore, the performance of the suboptimal algorithms in terms of state estimation accuracy is not degraded. They can also serve as a starting point for deriving numerically robust versions of the XDKF, which is crucial for industrial applications.

Zero gain sensitivity. First, the solution of the minimization problem (3.58) is simplified by adding the following assumption

Assumption 3.2. For all $\hat{\theta}_p, p = 1, \ldots, N_\theta$: $\mathbf{K}_{\theta_p} = \mathbf{0}$.

The meaning of Assumption 3.2 is the same as in Assumption 3.1. It is repeated here to emphasize its impact in the XDKF context. The rationale behind Assumption 3.2 is that reducing the state estimation sensitivity to parameters results in reducing the amplitude of $\mathbf{K}_{\theta_p}$. Then the impact of $\mathbf{K}_{\theta_p}$ is small and can be eliminated. Minimizing (3.58) using Assumption 3.2 results in the gain (3.69) where the term $\mathbf{S}_k$ (3.70) is simplified to

$$
\mathbf{S}_k = -\sum_p \gamma_p \left[\left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)\hat{\mathbf{s}}^T_{p,k|k-1}\mathbf{C}^T\right]. \tag{3.78}
$$

There are two important consequences of applying Assumption 3.2. Firstly, the cross terms $\mathbf{P}^{p,sx}, \mathbf{P}^{p,xs}$ need not be updated because they are no longer used in (3.61). Secondly, the sensitivity update (3.53) is simplified to

$$
\mathbf{s}_{p,k+1|k} = \left(\hat{\mathbf{A}} - \mathbf{KC}\right)\mathbf{s}_{p,k|k-1} - \mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} - \mathbf{B}_p\mathbf{u}_k, \tag{3.79}
$$

i.e., Assumption 3.2 eliminates the stochastic terms $\mathbf{e}_k, \tilde{\mathbf{x}}_{k|k-1}$ from the sensitivity update. Therefore, $\mathbf{s}_{p,k+1|k}$ is no longer a stochastic variable but a deterministic variable. Consequently, $\mathbf{s}_{p,k+1|k} = \hat{\mathbf{s}}_{p,k+1|k}$ and the second moment update (3.61) is reduced to

$$\mathbf{P}^{p,ss}_{k+1|k} = \mathbb{E}\left[\mathbf{s}_{p,k+1|k}\,\mathbf{s}^T_{p,k+1|k}\right] = \hat{\mathbf{s}}_{p,k+1|k}\,\hat{\mathbf{s}}^T_{p,k+1|k}\,. \tag{3.80}$$

Equation (3.80) implies that updating the $\mathbf{P}^{p,ss}_{k+1|k}$ is unnecessary since it can be directly computed from the sensitivity anytime.

The XDKF algorithm with the zero gain sensitivity assumption (XDKF-Z) is summed up in Algorithm 3.4.

Algorithm 3.4 (XDKF with the zero gain sensitivity assumption — XDKF-Z).

- Given γ_p and prior estimates $\hat{\mathbf{x}}_{k|k-1}\,, \mathbf{P}_{k|k-1}\,, \hat{\mathbf{s}}_{p,k|k-1}\,.$
- Compute the auxiliary variable

$$\mathbf{S}_k = -\sum_p \gamma_p \left[\left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)\hat{\mathbf{s}}^T_{p,k|k-1}\,\mathbf{C}^T\right].$$

- Compute the gain

$$\begin{aligned}\mathbf{K} = &\left[\hat{\mathbf{A}}\left(\alpha\mathbf{P}^{xx}_{k|k-1} + \sum_p \gamma_p\hat{\mathbf{s}}_{p,k|k-1}\,\hat{\mathbf{s}}^T_{p,k|k-1}\right)\mathbf{C}^T + \mathbf{S}_k\right]\\ &\times\left[\mathbf{C}\left(\alpha\mathbf{P}^{xx}_{k|k-1} + \sum_p \gamma_p\hat{\mathbf{s}}_{p,k|k-1}\,\hat{\mathbf{s}}^T_{p,k|k-1}\right)\mathbf{C}^T + \alpha\mathbf{R}\right]^{-1}.\end{aligned}$$

- Update the state prediction and the second moments

$$\begin{aligned}\hat{\mathbf{x}}_{k+1|k} &= \hat{\mathbf{A}}\hat{\mathbf{x}}_{k|k-1} + \hat{\mathbf{B}}\mathbf{u}_k + \mathbf{K}\left(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1}\right),\\ \hat{\mathbf{s}}_{p,k+1|k} &= \left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)\hat{\mathbf{s}}_{p,k|k-1} - \left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right),\\ \mathbf{P}^{xx}_{k+1|k} &= \left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)\mathbf{P}^{xx}_{k|k-1}\left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)^T + \mathbf{K}\mathbf{R}\mathbf{K}^T + \mathbf{Q}.\end{aligned}$$

It is important to note that the XDKF-Z defined in Algorithm 3.4 is an equivalent of the DKF special case in Algorithm 3.2. In other words, the XDKF is reduced to the SDKF by applying Assumption 3.2.

In order to analyze the algorithm in more detail, the XDKF-Z form needs to be further modified. First, the covariance updates can be aggregated to the cumulative update of weighted second moments

$$\begin{aligned}\mathbf{P}^{\Sigma}_{k+1|k} = &\left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)\mathbf{P}^{\Sigma}_{k|k-1}\left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)^T + \mathbf{K}\left(\alpha\mathbf{R}\right)\mathbf{K}^T + \mathbf{Q}^{\Sigma}_{k|k-1}\\ &- \mathbf{S}_{k|k-1}\,\mathbf{K}^T - \mathbf{K}\mathbf{S}^T_{k|k-1}\,,\end{aligned} \tag{3.81}$$

where

$$\mathbf{P}^{\Sigma}_{k|k-1} = \alpha\mathbf{P}^{xx}_{k|k-1} + \sum_p \gamma_p\hat{\mathbf{s}}_{p,k|k-1}\,\hat{\mathbf{s}}^T_{p,k|k-1}\,, \tag{3.82}$$

$$\begin{aligned}\mathbf{Q}^{\Sigma}_{k|k-1} = &\,\alpha\mathbf{Q} + \sum_p \gamma_p\left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)\left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)^T\\ &- \left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)\hat{\mathbf{s}}^T_{p,k|k-1}\,\hat{\mathbf{A}}^T\\ &- \hat{\mathbf{A}}\hat{\mathbf{s}}_{p,k|k-1}\left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)^T,\end{aligned} \tag{3.83}$$

$$\mathbf{S}_{k|k-1} = -\sum_p \gamma_p\left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)\hat{\mathbf{s}}^T_{p,k|k-1}\,\mathbf{C}^T. \tag{3.84}$$

When the XDKF-Z gain (3.69) with (3.78) is substituted into (3.81), the weighted cumulative update can be rewritten into the form of difference Riccati equation (DRE)

$$\begin{aligned}\mathbf{P}^{\Sigma}_{k+1|k} &= \hat{\mathbf{A}}\mathbf{P}^{\Sigma}_{k|k-1}\hat{\mathbf{A}}^{T} + \mathbf{Q}^{\Sigma}_{k|k-1} - \\ &- \left(\hat{\mathbf{A}}\mathbf{P}^{\Sigma}_{k|k-1}\mathbf{C}^{T} + \mathbf{S}_{k|k-1}\right)\left(\mathbf{C}\mathbf{P}^{\Sigma}_{k|k-1}\mathbf{C}^{T} + \alpha\mathbf{R}\right)^{-1} \times \\ &\times \left(\mathbf{C}\mathbf{P}^{\Sigma}_{k|k-1}\hat{\mathbf{A}}^{T} + \mathbf{S}^{T}_{k|k-1}\right).\end{aligned} \tag{3.85}$$

The weighted cumulative update in (3.85) can be used instead of individual updates if the knowledge of the individual second moments is not required. Then the gain can be formulated using the cumulative second moment (3.82) as

$$\mathbf{K} = \left(\hat{\mathbf{A}}\mathbf{P}^{\Sigma}_{k|k-1}\mathbf{C}^{T} + \mathbf{S}_{k|k-1}\right)\left(\mathbf{C}\mathbf{P}^{\Sigma}_{k|k-1}\mathbf{C}^{T} + \alpha\mathbf{R}\right)^{-1}. \tag{3.86}$$

Using the weighted cumulative update to update the state estimate is sufficient. Notice the similarity between the cumulative update (3.85) and the standard Kalman filter with the correlated process and measurement noise in (3.6), (3.7). Equations (3.85) and (3.86) correspond to the covariance update and gain of the standard Kalman filter where the process and measurement noise is described by time-variable statistics

$$\begin{bmatrix}\mathbf{v}_k \\ \mathbf{e}_k\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{0} \\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{Q}^{\Sigma}_{k|k-1} & \mathbf{S}_{k|k-1} \\ \mathbf{S}^{T}_{k|k-1} & \alpha\mathbf{R}\end{bmatrix}\right). \tag{3.87}$$

The cumulative form of the XDKF-Z is summarized in Algorithm 3.5.

Algorithm 3.5 (XDKF-Z with cumulative update).

- Given γ_p and prior estimates $\hat{\mathbf{x}}_{k|k-1}, \mathbf{P}^{\Sigma}_{k|k-1}, \hat{\mathbf{s}}_{p,k|k-1}$
- Compute the auxiliary variables

$$\begin{aligned}\mathbf{S}_{k|k-1} &= -\sum_p \gamma_p \left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)\hat{\mathbf{s}}^{T}_{p,k|k-1}\mathbf{C}^{T}, \\ \mathbf{Q}^{\Sigma}_{k|k-1} &= \alpha\mathbf{Q} + \sum_p \gamma_p \left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)\left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)^{T} \\ &\quad - \left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)\hat{\mathbf{s}}^{T}_{p,k|k-1}\hat{\mathbf{A}}^{T} \\ &\quad - \hat{\mathbf{A}}\hat{\mathbf{s}}_{p,k|k-1}\left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right)^{T}.\end{aligned}$$

- Compute the gain using (3.86)

$$\mathbf{K} = \left(\hat{\mathbf{A}}\mathbf{P}^{\Sigma}_{k|k-1}\mathbf{C}^{T} + \mathbf{S}_{k|k-1}\right)\left(\mathbf{C}\mathbf{P}^{\Sigma}_{k|k-1}\mathbf{C}^{T} + \alpha\mathbf{R}\right)^{-1}.$$

- Update the state, sensitivity, and cumulative second moment

$$\begin{aligned}\hat{\mathbf{x}}_{k+1|k} &= \hat{\mathbf{A}}\hat{\mathbf{x}}_{k|k-1} + \hat{\mathbf{B}}\mathbf{u}_k + \mathbf{K}\left(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1}\right), \\ \hat{\mathbf{s}}_{p,k+1|k} &= \left(\hat{\mathbf{A}} - \mathbf{K}\mathbf{C}\right)\hat{\mathbf{s}}_{p,k|k-1} - \left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right), \\ \mathbf{P}^{\Sigma}_{k+1|k} &= \hat{\mathbf{A}}\mathbf{P}^{\Sigma}_{k|k-1}\hat{\mathbf{A}}^{T} + \mathbf{Q}^{\Sigma}_{k|k-1} - \mathbf{K}\left(\mathbf{C}\mathbf{P}^{\Sigma}_{k|k-1}\hat{\mathbf{A}}^{T} + \mathbf{S}^{T}_{k|k-1}\right).\end{aligned}$$

Remark 3.3. The XDKF-Z applied to a stochastic system with uncertain parameters gives the result equivalent to the standard KF applied to a stochastic system where correlated process and measurement noise covariances are interpreted with uncertainty in the parameters.

Steady-state solution. The steady-state solution of the optimal gain is particularly useful for applications in embedded devices with limited computing power. The steady-state gain of the standard Kalman filter can be found by solving the algebraic Riccati equation (ARE). Then it is used as a fixed gain, so the embedded device needs to evaluate only the state estimation update equation (3.47). Unfortunately, the steady-state gain cannot be found for desensitized filters due to the dependency of the gain (3.86) on the state estimate, sensitivity, and input. Therefore, the gain-scheduled steady-state solution is proposed.

When the time indices are omitted from (3.85), the weighted cumulative update can be rewritten in the ARE form

$$\begin{aligned}\mathbf{P}^{\Sigma}(\mathbf{z}) = \hat{\mathbf{A}}\mathbf{P}^{\Sigma}(\mathbf{z})\hat{\mathbf{A}}^{T} + \mathbf{Q}^{\Sigma}(\mathbf{z}) - \\ - \left(\hat{\mathbf{A}}\mathbf{P}^{\Sigma}(\mathbf{z})\mathbf{C}^{T} + \mathbf{S}(\mathbf{z})\right)\left(\mathbf{C}\mathbf{P}^{\Sigma}(\mathbf{z})\mathbf{C}^{T} + \alpha\mathbf{R}\right)^{-1} \times \\ \times \left(\mathbf{C}\mathbf{P}^{\Sigma}(\mathbf{z})\hat{\mathbf{A}}^{T} + \mathbf{S}(\mathbf{z})\right),\end{aligned} \tag{3.88}$$

which is parametrized by

$$\mathbf{z} = [\,\hat{\mathbf{x}}_{k|k-1}^{T} \quad \mathbf{u}_{k}^{T} \quad \hat{\mathbf{s}}_{1,k|k-1}^{T} \quad \dots \quad \hat{\mathbf{s}}_{N_\theta,k|k-1}^{T}\,]^{T}. \tag{3.89}$$

The steady-state solution $\mathbf{P}^{\Sigma}(\mathbf{z})$ of the weighted cumulative update for the given $\mathbf{z}$ can be computed using various numerical methods for solving the ARE. Then the parameterized solution is used to obtain the steady-state gain parametrized by $\mathbf{z}$

$$\mathbf{K}_{\infty}(\mathbf{z}) = \left(\hat{\mathbf{A}}\mathbf{P}^{\Sigma}(\mathbf{z})\mathbf{C}^{T} + \mathbf{S}(\mathbf{z})\right)\left(\mathbf{C}\mathbf{P}^{\Sigma}(\mathbf{z})\mathbf{C}^{T} + \alpha\mathbf{R}\right)^{-1}. \tag{3.90}$$

The value of $\mathbf{z}$ can change during estimation, so the real steady-state gain does not exist. The steady-state gain in (3.90) defines the steady-state gain, which would lead to steady-state $\mathbf{P}^{\Sigma}(\mathbf{z})$ if $\mathbf{z}$ is constant.

If the dependency of $\mathbf{P}^{\Sigma}$ and $\mathbf{K}_{\infty}$ on $\mathbf{z}$ is smooth, then in case of operation in a point $\mathbf{z}_0$ it may be acceptable to approximate the gains of the XDKF-Z by the steady-state gain (3.90). For example, the steady-state gains of system (3.120) are depicted in Figure 3.5. It can be seen that the gain is smooth, and the relative change of the gain for different weight γ is below 10 % in the neighborhood of the selected state $\hat{\mathbf{x}}_0 = \mathbf{0}$, hence it may be acceptable to use the steady-state approach.

The algorithm XDKF-S, which uses the steady-state gains, is summarized in Algorithm 3.6.

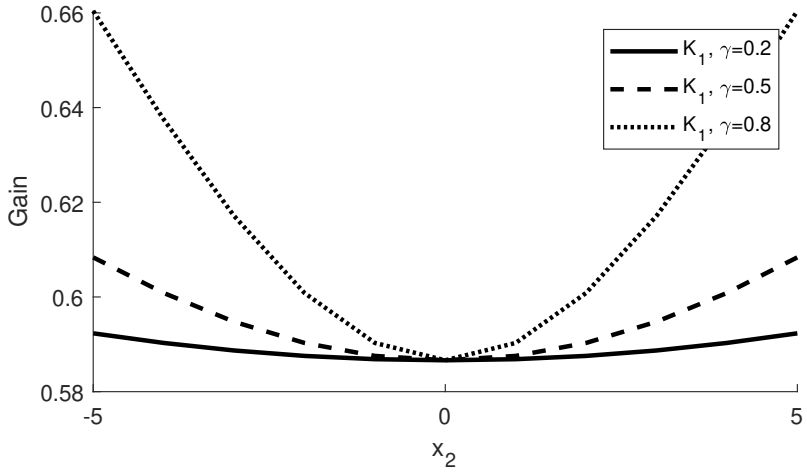

Figure 3.5. The figure shows steady-state gain K_1 for the system in (3.120) as a function of state x_2. The function is evaluated with zero sensitivity value.

Algorithm 3.6 (XDKF with steady-state gains — XDKF-S).

- Given γ_p and prior estimates $\hat{\mathbf{x}}_{k|k-1}, \hat{\mathbf{s}}_{p,k|k-1}$.
- Solve the ARE (3.88) and obtain steady-state gain $\mathbf{K}_\infty$ using (3.90).
- Update the state and sensitivity

$$\begin{aligned}\hat{\mathbf{x}}_{k+1|k} &= \hat{\mathbf{A}}\hat{\mathbf{x}}_{k|k-1} + \hat{\mathbf{B}}\mathbf{u}_k + \mathbf{K}_\infty\left(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1}\right),\\ \hat{\mathbf{s}}_{p,k+1|k} &= \left(\hat{\mathbf{A}} - \mathbf{K}_\infty\mathbf{C}\right)\hat{\mathbf{s}}_{p,k|k-1} - \left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right).\end{aligned}$$

An efficient way to apply the steady-state gain (3.90) is to precompute the steady-state gains of the XDKF-S offline. The gains should be precalculated for application-specific operating points. Then a gain scheduling method needs to be used at each iteration to select the gain corresponding to $\hat{\mathbf{x}}_{k|k-1}, \hat{\mathbf{s}}_{p,k|k-1}$ and $\mathbf{u}_k$. The accuracy of this method depends on the selection of operating points for precomputing the gains. Such an approach is commonly used for control algorithms in practice. The algorithm is summarized in Algorithm 3.7.

Algorithm 3.7 (XDKF-S with gain scheduling).

- Given γ_p and prior estimates $\hat{\mathbf{x}}_{k|k-1}$, $\hat{\mathbf{s}}_{p,k|k-1}$
- Use a look-up table, interpolation, or other gain scheduling method to determine the precomputed steady-state gain at the operating point $\mathbf{z}$

$$\mathbf{K}_\infty = \text{gainSchedulingMethod}\left(\mathbf{u}_k, \hat{\mathbf{x}}_{k|k-1}, \hat{\mathbf{s}}_{1,k|k-1}, \ldots, \hat{\mathbf{s}}_{N_\theta,k|k-1}\right).$$

- Update the state prediction

$$\begin{aligned}\hat{\mathbf{x}}_{k+1|k} &= \hat{\mathbf{A}}\hat{\mathbf{x}}_{k|k-1} + \hat{\mathbf{B}}\mathbf{u}_k + \mathbf{K}_\infty\left(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1}\right),\\ \hat{\mathbf{s}}_{p,k+1|k} &= \left(\hat{\mathbf{A}} - \mathbf{K}_\infty\mathbf{C}\right)\hat{\mathbf{s}}_{p,k|k-1} - \left(\mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k\right).\end{aligned}$$

3.2.4 Separated steps

All XDKF algorithms are derived in a single step form, i.e., combining the data update and time update steps. Sometimes, this form is not practical, and the data update (filtering) step and time update (prediction) step need to be separated. Thanks to the similarity with the standard Kalman filter, the single-step XDKF-Z in Algorithm 3.5 can be separated by decorrelating the process and measurement noise correlated by (3.84). The method for steps separation shown in [58] and [C3] updates the process noise statistics based on the output measurement

$$\begin{aligned}\hat{\mathbf{v}}_{k|k} &= \mathbb{E}\left[\mathbf{v}_k\right] - \mathbf{S}_k\left(\alpha\mathbf{R}\right)^{-1}\left(\mathbf{e}_k - \mathbb{E}\left[\mathbf{e}_k\right]\right), \\ &= -\mathbf{S}_k\left(\alpha\mathbf{R}\right)^{-1}\left(\mathbf{y}_k - \mathbf{C}\mathbf{x}_k\right), \qquad (3.91)\\ \mathbf{Q}^{\Sigma}_{k|k} &= \mathbf{Q}^{\Sigma}_{k|k-1} - \mathbf{S}_k\left(\alpha\mathbf{R}\right)^{-1}\left(\mathbf{S}_k\right)^T. \qquad (3.92)\end{aligned}$$

Note that the process noise estimate $\hat{\mathbf{v}}_{k|k}$ is not correlated with the measurement noise. Then $\hat{\mathbf{v}}_{k|k}$ can be used to transform the state propagation equation (3.39) into

$$\mathbf{x}_{k+1} = \left(\mathbf{A}(\boldsymbol{\theta}) - \mathbf{S}_k\left(\alpha\mathbf{R}\right)^{-1}\mathbf{C}\right)\mathbf{x}_k + \mathbf{B}(\boldsymbol{\theta})\mathbf{u}_k - \mathbf{S}_k\left(\alpha\mathbf{R}\right)^{-1}\mathbf{y}_k + \mathbf{v}^{\mathrm{dc}}_k, \qquad (3.93)$$

with the decorrelated process and measurement noise

$$\begin{bmatrix}\mathbf{v}^{\mathrm{dc}}_k \\ \mathbf{e}_k\end{bmatrix} = \mathcal{N}\left(\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{Q}^{\Sigma}_{k|k} & \mathbf{0}\\ \mathbf{0} & \alpha\mathbf{R}\end{bmatrix}\right). \qquad (3.94)$$

Similarly to the standard Kalman filter with the non-correlated process and measurement noise, the algorithm can be split into a data-update step and a time-update step. The final two-step algorithm is described in Algorithm 3.8.

Algorithm 3.8 (XDKF-Z with separated steps).

- Given γ_p and prior estimates $\hat{\mathbf{x}}_{k|k-1}, \mathbf{P}^{\Sigma}_{k|k-1}, \hat{\mathbf{s}}_{p,k|k-1}$.
- Compute the auxiliary variables

$$\begin{aligned}\mathbf{b}_{p,k} &= \mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k,\\ \mathbf{S}_{k|k-1} &= -\sum_p \gamma_p \mathbf{b}_{p,k}\hat{\mathbf{s}}^T_{p,k|k-1}\mathbf{C}^T,\\ \mathbf{Q}^{\Sigma}_{k|k-1} &= \alpha\mathbf{Q} + \sum_p \gamma_p\left(\mathbf{b}_{p,k}\mathbf{b}^T_{p,k} - \mathbf{b}_{p,k}\hat{\mathbf{s}}^T_{p,k|k-1}\hat{\mathbf{A}}^T - \hat{\mathbf{A}}\hat{\mathbf{s}}_{p,k|k-1}\mathbf{b}^T_{p,k}\right).\end{aligned}$$

- Data-update step

$$\begin{aligned}\mathbf{K}_k &= \mathbf{P}^{\Sigma}_{k|k-1}\mathbf{C}^T\left(\mathbf{C}\mathbf{P}^{\Sigma}_{k|k-1}\mathbf{C}^T + \alpha\mathbf{R}\right)^{-1},\\ \hat{\mathbf{x}}_{k|k} &= \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\left(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1}\right),\\ \hat{\mathbf{s}}_{p,k|k} &= \hat{\mathbf{s}}_{p,k|k-1} - \mathbf{K}_k\mathbf{C}\hat{\mathbf{s}}_{p,k|k-1},\\ \mathbf{P}^{\Sigma}_{k|k} &= \mathbf{P}^{\Sigma}_{k|k-1} - \mathbf{K}_k\mathbf{C}\mathbf{P}^{\Sigma}_{k|k-1}.\end{aligned}$$

- Time-update step

$$\begin{aligned}\hat{\mathbf{A}}_{\mathrm{dc}} &= \hat{\mathbf{A}} - \mathbf{S}_k\left(\alpha\mathbf{R}\right)^{-1}\mathbf{C},\\ \mathbf{Q}^{\Sigma}_{k|k} &= \mathbf{Q}^{\Sigma}_{k|k-1} - \mathbf{S}_k\left(\alpha\mathbf{R}\right)^{-1}\left(\mathbf{S}_k\right)^T,\\ \hat{\mathbf{x}}_{k+1|k} &= \hat{\mathbf{A}}_{\mathrm{dc}}\hat{\mathbf{x}}_{k|k} + \hat{\mathbf{B}}\mathbf{u}_k + \mathbf{S}_k\left(\alpha\mathbf{R}\right)^{-1}\mathbf{y}_k,\\ \hat{\mathbf{s}}_{p,k+1|k} &= \hat{\mathbf{A}}_{\mathrm{dc}}\hat{\mathbf{s}}_{p,k|k} - \mathbf{b}_{p,k},\\ \mathbf{P}^{\Sigma}_{k+1|k} &= \hat{\mathbf{A}}_{\mathrm{dc}}\mathbf{P}^{\Sigma}_{k|k}\left(\hat{\mathbf{A}}_{\mathrm{dc}}\right)^T + \mathbf{Q}^{\Sigma}_{k|k}.\end{aligned}$$

A numerically robust formulation of Algorithm 3.8 is proposed in Appendix A.

3.2.5 Stability

Formulation of the second moment updates in the form of the Riccati equation is an important result since this matrix equation is well studied with many published results regarding its solution and stability. Anderson and Moore [59] proved that the exponential stability of the Kalman filter is obtained when the stochastic system is uniformly detectable and stabilizable. The proof of this theorem and all other published proofs of Kalman filter asymptotic stability exploit its minimum-variance property. According to the definition of detectability, if the pair $\left(\hat{\mathbf{A}}, \mathbf{C}\right)$ is detectable, there exists some $\mathbf{L}$ such that $\left(\hat{\mathbf{A}} - \mathbf{LC}\right)$ is asymptotically stable. Then an asymptotically stable filter is obtained as

$$\begin{aligned} \hat{\mathbf{x}}_{k+1|k} &= \hat{\mathbf{A}}\hat{\mathbf{x}}_{k|k-1} + \hat{\mathbf{B}}\mathbf{u}_k + \mathbf{L}\left(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1}\right), \\ \mathbf{P}^{xx}_{k+1|k} &= \left(\hat{\mathbf{A}} - \mathbf{LC}\right)\mathbf{P}^{xx}_{k|k-1}\left(\hat{\mathbf{A}} - \mathbf{LC}\right)^T + \mathbf{Q} + \mathbf{LRL}^T. \end{aligned} \tag{3.95}$$

The covariance is updated using the Lyapunov equation with a stable matrix $\left(\hat{\mathbf{A}} - \mathbf{LC}\right)$, which implies that the covariance is bounded. The trace of the covariance matrix will be minimized when the optimal Kalman filter gain is used since it is the minimum-variance filter. Because the XDKF in Algorithm 3.3 with zero γ_p weights is identical to the standard Kalman filter, the stability conditions for the XDKF can be formulated by Theorem 3.1.

Theorem 3.1 (Stable XDKF existence). Assume the system (3.39) where the pair $\left(\hat{\mathbf{A}}, \mathbf{C}\right)$ is detectable, the pair $\left(\hat{\mathbf{A}}, \mathbf{Q}^{1/2}\right)$ is stabilizable, and $\mathbf{P}_{0|-1} \succeq 0$. Then there exist weights $\gamma_p, 0 \leq \sum_p \gamma_p < 1$, and $\gamma_p \geq 0$, such that the XDKF applied to the system (3.39) is stable.

Proof. The XDKF is reduced to the standard Kalman filter by setting all weights to zero. Anderson and Moore [59] proved that the detectability of the pair $\left(\hat{\mathbf{A}}, \mathbf{C}\right)$ and the stabilizability of the pair $\left(\hat{\mathbf{A}}, \mathbf{Q}^{1/2}\right)$ are sufficient conditions to obtain the exponential stability of the standard Kalman filter. ■

The nonzero γ_p weights weaken the minimum-variance property but decrease the sensitivity to parameter error. Therefore, in general, the matrix $\left(\hat{\mathbf{A}} - \mathbf{KC}\right)$ with gain $\mathbf{K}$ obtained by the XDKF may not be stable for all weights γ_p. Consequently, the XDKF is not generally stable for all weights γ_p. The definition of stability constraints on weights γ_p for a given system is an open question.

The stability of XDKF-S in Algorithm 3.6 can be analyzed easier because the gain is calculated from ARE (3.88) which can be rewritten in an equivalent form with the decorrelated process and measurement noise using the method in Section 3.2.4

$$\mathbf{P}^{\Sigma} = \mathbf{A}_{\mathrm{dc}}\mathbf{P}^{\Sigma}\mathbf{A}_{\mathrm{dc}}^T + \mathbf{Q}_{\mathrm{dc}} - \mathbf{A}_{\mathrm{dc}}\mathbf{P}^{\Sigma}\mathbf{C}^T\left(\mathbf{CP}^{\Sigma}\mathbf{C}^T + \alpha\mathbf{R}\right)^{-1}\mathbf{CP}^{\Sigma}\mathbf{A}_{\mathrm{dc}}^T, \tag{3.96}$$

where

$$\mathbf{A}_{\mathrm{dc}} = \hat{\mathbf{A}} - \mathbf{S}\left(\mathbf{z}\right)\left(\alpha\mathbf{R}\right)^{-1}\mathbf{C}, \tag{3.97}$$

$$\mathbf{Q}_{\mathrm{dc}} = \mathbf{Q}^{\Sigma}\left(\mathbf{z}\right) - \mathbf{S}\left(\mathbf{z}\right)\left(\alpha\mathbf{R}\right)^{-1}\mathbf{S}^T\left(\mathbf{z}\right). \tag{3.98}$$

The essential requirement is that matrix $\mathbf{Q}_{\mathrm{dc}}$ needs to be a positive semidefinite matrix. By evaluating the elements in (3.98), the requirement can be written into following

condition

$$0 \preceq \mathbf{Q} + \sum_p \frac{\gamma_p}{\alpha} \left(\mathbf{b}_{p,k}\mathbf{b}_{p,k}^T - \mathbf{b}_{p,k}\hat{\mathbf{s}}_{p,k|k-1}^T \hat{\mathbf{A}}^T - \hat{\mathbf{A}}\hat{\mathbf{s}}_{p,k|k-1}\mathbf{b}_{p,k}^T \right) - \\ - \sum_p \left(\frac{\gamma_p}{\alpha} \right)^2 \mathbf{b}_{p,k}\hat{\mathbf{s}}_{p,k|k-1}^T \mathbf{C}^T\mathbf{R}^{-1}\mathbf{C}\hat{\mathbf{s}}_{p,k|k-1}\mathbf{b}_{p,k}^T, \tag{3.99}$$

where

$$\mathbf{b}_{p,k} = \mathbf{A}_p\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p\mathbf{u}_k. \tag{3.100}$$

From the definition, $\mathbf{R}$ is a positive definite matrix, and $\mathbf{Q}$ is a positive semidefinite matrix. Then there exists a decomposition

$$\mathbf{Q} = \mathbf{\Gamma}\mathbf{\Gamma}^T. \tag{3.101}$$

If the condition (3.99) is satisfied, the following decomposition exists

$$\mathbf{Q}_{\mathrm{dc}} = \mathbf{\Gamma}_{\mathrm{dc}}\mathbf{\Gamma}_{\mathrm{dc}}^T. \tag{3.102}$$

Goodwin and Sin [60] proved that the ARE (3.96) has the unique limiting positive semidefinite solution, if the pair $(\mathbf{A}_{\mathrm{dc}}, \mathbf{C})$ is observable and the pair $(\mathbf{A}_{\mathrm{dc}}, \mathbf{\Gamma}_{\mathrm{dc}})$ is stabilizable. Following theorem shows that checking the pairs $\left(\hat{\mathbf{A}}, \mathbf{C}\right)$, $\left(\hat{\mathbf{A}}, \mathbf{\Gamma}_{\mathrm{dc}}\right)$ for observability and stabilizability is sufficient.

Theorem 3.2 (Equivalent pairs). Let (3.39) be a system with substituted parameter vector $\hat{\boldsymbol{\theta}}$ such that the pair $\left(\hat{\mathbf{A}}, \mathbf{C}\right)$ is observable and the pair $\left(\hat{\mathbf{A}}, \mathbf{\Gamma}_{\mathrm{dc}}\right)$ is stabilizable. Assume that $\mathbf{Q}_{\mathrm{dc}} \succ 0$ is true for $\mathbf{Q}_{\mathrm{dc}}$ (3.98) applied to the system. Then also the pair $(\mathbf{A}_{\mathrm{dc}}, \mathbf{C})$ is observable and the pair $(\mathbf{A}_{\mathrm{dc}}, \mathbf{\Gamma}_{\mathrm{dc}})$ is stabilizable for $\mathbf{A}_{\mathrm{dc}}$ (3.97) applied to the system.

Proof. First, the pair $\left(\hat{\mathbf{A}}, \mathbf{\Gamma}_{\mathrm{dc}}\right)$ is stabilizable; therefore, the corresponding observability matrix $\mathcal{O}$ has full rank. There exist a full rank matrix $\mathbf{T}_{\mathrm{o}}$ such that

$$\mathbf{T}_{\mathrm{o}}\mathcal{O} = \mathcal{O}_{\mathrm{dc}}, \tag{3.103}$$

where

$$\mathcal{O} = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\hat{\mathbf{A}} \\ \vdots \\ \mathbf{C}\hat{\mathbf{A}}^{N_x-1} \end{bmatrix}, \quad \mathcal{O}_{\mathrm{dc}} = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}(\hat{\mathbf{A}} - \mathbf{S}\mathbf{R}^{-1}\mathbf{C}) \\ \vdots \\ \mathbf{C}(\hat{\mathbf{A}} - \mathbf{S}\mathbf{R}^{-1}\mathbf{C})^{N_x-1} \end{bmatrix}, \\ \mathbf{T}_{\mathrm{o}} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \dots & \\ -\mathbf{C}\mathbf{S}\mathbf{R}^{-1} & \mathbf{I} & \mathbf{0} & \dots \\ -\mathbf{C}\mathbf{A}\mathbf{S}\mathbf{R}^{-1} + \left(\mathbf{C}\mathbf{S}\mathbf{R}^{-1}\right)^2 & -\mathbf{C}\mathbf{S}\mathbf{R}^{-1} & \mathbf{I} & \mathbf{0} \quad \dots \\ \vdots & & & \ddots \end{bmatrix}. \tag{3.104}$$

All elements are not displayed to keep the text compact. The matrix $\mathbf{T}_{\mathrm{o}}$ is a lower unitriangular matrix. Hence it has full rank. Then $\mathrm{rank}(\mathcal{O}_{\mathrm{dc}}) = \mathrm{rank}(\mathcal{O})$.

Similarly, the pair $\left(\hat{\mathbf{A}}, \mathbf{C}\right)$ is observable; therefore, the corresponding controllability matrix $\mathcal{C}$ has full rank. There exist a full rank matrix $\mathbf{T}_{\mathrm{c}}$ such that

$$\mathcal{C}\mathbf{T}_{\mathrm{c}} = \mathcal{C}_{\mathrm{dc}}, \tag{3.105}$$

where

$$\begin{aligned}
\mathcal{C} &= \begin{bmatrix} \mathbf{\Gamma} & \hat{\mathbf{A}}\mathbf{\Gamma} & \dots & \hat{\mathbf{A}}^{N_x-1}\mathbf{\Gamma} \end{bmatrix}, \\
\mathcal{C}_{\mathrm{dc}} &= \begin{bmatrix} \mathbf{\Gamma} & (\hat{\mathbf{A}} - \mathbf{S}\mathbf{R}^{-1}\mathbf{C})\mathbf{\Gamma} & \dots & (\hat{\mathbf{A}} - \mathbf{S}\mathbf{R}^{-1}\mathbf{C})^{N_x-1}\mathbf{\Gamma} \end{bmatrix}, \\
\mathbf{T}_{\mathrm{c}} &= \begin{bmatrix} \mathbf{I} & -\mathbf{\Gamma}^{-1}\mathbf{S}\mathbf{R}^{-1}\mathbf{C}\mathbf{\Gamma} & \dots & \mathbf{\Gamma}^{-1}\left((\hat{\mathbf{A}} - \mathbf{S}\mathbf{R}^{-1}\mathbf{C})^{N_x-1} - \hat{\mathbf{A}}^{N_x-1}\right)\mathbf{\Gamma} \\ \mathbf{0} & \mathbf{I} & \dots & \mathbf{0} \\ \vdots & & \ddots & \vdots \\ \mathbf{0} & & & \mathbf{I} \end{bmatrix}.
\end{aligned} \tag{3.106}$$

The matrix $\mathbf{T}_{\mathrm{c}}$ is an upper unitriangular matrix. Hence it has full rank. Then $\mathrm{rank}(\mathcal{C}) = \mathrm{rank}(\mathcal{C}_{\mathrm{dc}})$. ■

This result can be used to formulate the stability of the XDKF-S as follows.

Theorem 3.3 (XDKF-S stability). Let (3.39) be a system with substituted parameter vector $\hat{\boldsymbol{\theta}}$ such that the pair $\left(\hat{\mathbf{A}}, \mathbf{C}\right)$ is observable. Assume that the vector $\mathbf{z}$ (3.89) is bounded and the condition (3.99) is satisfied when they are applied to the system. Then the XDKF-S in Algorithm 3.6 with weights γ_p is stable if the pair $\left(\hat{\mathbf{A}}, \mathbf{\Gamma}_{\mathrm{dc}}\right)$ is stabilizable for each $\mathbf{z}$.

Proof. The XDKF-S with fixed $\mathbf{K}_\infty$ calculated using fixed $\mathbf{z}$ will converge to a unique limiting positive semidefinite solution $\mathbf{P}^\Sigma(\mathbf{z})$. This is true for each fixed value of $\mathbf{z}$. In the XDKF-S, the state and input values can change between iteration steps, and consequently, each $\mathbf{K}_\infty$ pushes the cumulative second moment towards different limiting $\mathbf{P}^\Sigma(\mathbf{z})$. Therefore, the algorithm will converge to a time-varying $\mathbf{P}^\Sigma(\mathbf{z})$. Although it might never converge to a single value, the algorithm cannot diverge because, according to [60] and Theorem 3.2, all possible limiting $\mathbf{P}^\Sigma(\mathbf{z})$ are positive semidefinite and finite. ■

The stability of the XDKF-S can be analyzed using the sufficient condition in the following theorem.

Theorem 3.4 (Stability bound for XDKF-S). Let $\alpha_{\mathrm{b}} \in (0, 1]$ be a weight for which the matrix

$$\mathbf{Q}_{\mathrm{L}} = \alpha_{\mathrm{b}}\mathbf{Q} - \sum_p \left(\gamma_{p,\mathrm{b}}\hat{\mathbf{A}}\hat{\mathbf{s}}_p\hat{\mathbf{s}}_p^T\hat{\mathbf{A}}^T + \gamma_{p,\mathrm{b}}^2\mathbf{b}_p(\hat{\mathbf{x}}, \mathbf{u})\hat{\mathbf{s}}_p^T\mathbf{C}^T\mathbf{R}^{-1}\mathbf{C}\hat{\mathbf{s}}_p\mathbf{b}_p^T(\hat{\mathbf{x}}, \mathbf{u})\right) \tag{3.107}$$

is positive definite for all feasible values of $\mathbf{z}$, and it can be decomposed as $\mathbf{Q}_{\mathrm{L}} = \mathbf{\Gamma}_{\mathrm{L}}\mathbf{\Gamma}_{\mathrm{L}}^T$. Let (3.39) be a system with substituted parameter vector $\hat{\boldsymbol{\theta}}$ such that the pair $\left(\hat{\mathbf{A}}, \mathbf{C}\right)$ is observable, and the pair $\left(\hat{\mathbf{A}}, \mathbf{\Gamma}_{\mathrm{L}}\right)$ is stabilizable. Then the XDKF-S with weights $\alpha = 1 - \sum \gamma_p$ is stable for all $\alpha \geq \alpha_{\mathrm{b}}$.

Proof. Assume the ARE equation (3.96) where the process noise covariance is replaced with $\mathbf{Q}_0 = \mathbf{\Gamma}_0\mathbf{\Gamma}_0^T$

$$\mathbf{P}^\Sigma = \mathbf{A}_{\mathrm{dc}}\mathbf{P}^\Sigma\mathbf{A}_{\mathrm{dc}}^T - \mathbf{A}_{\mathrm{dc}}\mathbf{P}^\Sigma\mathbf{C}^T\left(\mathbf{C}\mathbf{P}^\Sigma\mathbf{C}^T + \alpha\mathbf{R}\right)^{-1}\mathbf{C}\mathbf{P}^\Sigma\mathbf{A}_{\mathrm{dc}}^T + \mathbf{Q}_0. \tag{3.108}$$

The ARE (3.108) has the unique positive semidefinite solution if the pair $\left(\hat{\mathbf{A}}_{\mathrm{dc}}, \mathbf{C}\right)$ is observable, and the pair $\left(\hat{\mathbf{A}}_{\mathrm{dc}}, \mathbf{\Gamma}_0\right)$ is stabilizable [60]. Theorem 3.2 says that it is sufficient to analyze the observability of the pair $\left(\hat{\mathbf{A}}, \mathbf{C}\right)$, and the stabilizability of the pair $\left(\hat{\mathbf{A}}, \mathbf{\Gamma}_0\right)$.

Lemma 4 in [61] states that if $\left(\hat{\mathbf{A}}, \mathbf{\Gamma}_L\right)$ is stabilizable pair and $\mathbf{\Gamma}_0\mathbf{\Gamma}_0^T \succeq \mathbf{\Gamma}_\mathrm{L}\mathbf{\Gamma}_\mathrm{L}^T$ then the pair $\left(\hat{\mathbf{A}}, \mathbf{\Gamma}_0\right)$ is also stabilizable. Given that the pair $\left(\hat{\mathbf{A}}, \mathbf{\Gamma}_\mathrm{L}\right)$ is stabilizable, the ARE (3.108) has a unique positive semidefinite solution if $\mathbf{Q}_0 \succeq \mathbf{Q}_\mathrm{L}$.

Since

$$\begin{aligned}\left(\hat{\mathbf{A}}\hat{\mathbf{s}}_p - \mathbf{b}_p\right)\left(\hat{\mathbf{A}}\hat{\mathbf{s}}_p - \mathbf{b}_p\right)^T &\succeq 0,\\ \mathbf{b}_p\mathbf{b}_p^T - \hat{\mathbf{A}}\hat{\mathbf{s}}_p\mathbf{b}_p^T - \mathbf{b}_p\hat{\mathbf{s}}_p^T\hat{\mathbf{A}}^T &\succeq -\hat{\mathbf{A}}\hat{\mathbf{s}}_p\hat{\mathbf{s}}_p^T\hat{\mathbf{A}}^T,\end{aligned} \tag{3.109}$$

then $\mathbf{Q}_\mathrm{dc}(\alpha_\mathrm{b}) \succeq \mathbf{Q}_\mathrm{L}(\alpha_\mathrm{b})$ is always true and the ARE in (3.96) has a unique positive semidefinite solution for α_b. Furthermore, the lower bound increases with increasing α

$$\mathbf{Q}_\mathrm{L}(\alpha) - \mathbf{Q}_\mathrm{L}(\alpha_\mathrm{b}) \succeq 0, \tag{3.110}$$

$$\begin{aligned}(\alpha - \alpha_\mathrm{b})\mathbf{Q} - \sum_p (\gamma_p - \gamma_{p,\mathrm{b}})\hat{\mathbf{A}}\hat{\mathbf{s}}_p\hat{\mathbf{s}}_p^T\hat{\mathbf{A}}^T -\\ - \sum_p (\gamma_p^2 - \gamma_{p,\mathrm{b}}^2)\mathbf{b}_p(\hat{\mathbf{x}}, \mathbf{u})\hat{\mathbf{s}}_p^T\mathbf{C}^T\mathbf{R}^{-1}\mathbf{C}\hat{\mathbf{s}}_p\mathbf{b}_p^T(\hat{\mathbf{x}}, \mathbf{u}) \succeq 0.\end{aligned} \tag{3.111}$$

Then $\mathbf{Q}_\mathrm{dc}(\alpha) \succeq \mathbf{Q}_\mathrm{L}(\alpha_\mathrm{b})$ which means that the ARE in (3.96) has a unique positive semidefinite solution for all $\alpha \geq \alpha_\mathrm{b}$. ■

Remark 3.4. Theorem 3.4 states two important things. First, to check the stability of the XDKF-S, it is sufficient to check the positive definiteness of (3.107). Secondly, the stability of the XDKF-S for some weights γ_p implies the stability of the XDKF-S for all weights with the sum of weights lower than $\sum_p \gamma_p$.

3.2.6 Normalized objectives

Setting up weights remains the main challenge for desensitized Kalman filters. In the prior work, a weight on sensitivity reduction objective could be set up between zero and infinity. Then the more inaccurate the parameter is expected to be, the higher weight is set. The quantitative meaning of high weight cannot be defined since it depends on the particular system. Therefore, one has to tune the weight using simulations.

In this thesis, weights in the range between zero and one are used. This range is supposed to simplify tuning the weights by introducing a finite upper bound, making the tuning more intuitive. However, the criterion (3.58) has multiple objectives which can be symbolically rewritten as

$$J(\mathbf{K}) = \alpha J_\mathrm{KF}(\mathbf{K}) + \sum_p \gamma_p J_{\mathrm{S},p}(\mathbf{K}), \tag{3.112}$$

where J_KF is the minimum-variance objective and $J_{\mathrm{S},p}$ are the sensitivity reduction objectives. The issue with (3.112) is that the objectives are not scaled, in general. Therefore, the weight value needs to include the fixed scaling of objectives, which could result in tuning the weights in low orders of magnitude. If the normalized objectives are used, the weight can be tuned intuitively.

Section 3.2.5 discusses that not all weight settings are stable, and once some set of stable weights $\bar{\gamma}_p$ is found and condition (3.107) is satisfied, then all sets where $\sum_p \gamma_p \leq \sum_p \bar{\gamma}_p$ are also stable. Nevertheless, a set of stable weights that does not satisfy $\sum_p \gamma_p \leq \sum_p \bar{\gamma}_p$ can also exist. Using these results, the criterion J can be rewritten in a form where each objective leads to a stable solution

$$J(\mathbf{K}) = \alpha J_\mathrm{KF}(\mathbf{K}) + \sum_p \gamma_p \bar{J}_{\mathrm{S},p}(\mathbf{K}), \tag{3.113}$$

where

$$\bar{J}_{\mathrm{S},p}(\mathbf{K}) = (1-\bar{\gamma}_p)J_{\mathrm{KF}}(\mathbf{K}) + \bar{\gamma}_p J_{\mathrm{S},p}(\mathbf{K}). \tag{3.114}$$

are criteria for stable weights $\bar{\gamma}_p$. Note that if an arbitrary number of weights in the stable set of weights $\{\bar{\gamma}_1, \ldots, \bar{\gamma}_{N_\theta}\}$ are replaced by zero, the stability of the set is maintained.

The objectives can be normalized using a part of the multi-objective adaptive weighted sum method introduced in [62]. Let $\mathbf{K}^\star_{\mathrm{KF}}, \mathbf{K}^\star_{\mathrm{S},p}$ be the optimal solution for individual optimization of $J_{\mathrm{KF}}, \bar{J}_{\mathrm{S},p}$ respectively. Then the utopia points are defined as

$$J^{\mathrm{Utopia}}_{\mathrm{KF}} = J_{\mathrm{KF}}(\mathbf{K}^\star_{\mathrm{KF}}), \quad \bar{J}^{\mathrm{Utopia}}_{\mathrm{S},p} = \bar{J}_{\mathrm{S},p}(\mathbf{K}^\star_{\mathrm{S},p}). \tag{3.115}$$

In multi-objective optimization, utopia points (also called ideal points) represent a lower bound vector of the Pareto optimal solutions. Nadir points represent an upper bound of the Pareto optimal solutions

$$\begin{aligned} J^{\mathrm{Nadir}}_{\mathrm{KF}} &= \max\left[J_{\mathrm{KF}}(\mathbf{K}^\star_{\mathrm{KF}}), J_{\mathrm{KF}}(\mathbf{K}^\star_{\mathrm{S},1}), \ldots, J_{\mathrm{KF}}(\mathbf{K}^\star_{\mathrm{S},N_\theta})\right], \\ \bar{J}^{\mathrm{Nadir}}_{\mathrm{S},p} &= \max\left[\bar{J}_{\mathrm{S},p}(\mathbf{K}^\star_{\mathrm{KF}}), \bar{J}_{\mathrm{S},p}(\mathbf{K}^\star_{\mathrm{S},1}), \ldots, \bar{J}_{\mathrm{S},p}(\mathbf{K}^\star_{\mathrm{S},N_\theta})\right]. \end{aligned} \tag{3.116}$$

The normalization of the objective function is done using the utopia and nadir points as follows

$$J_{\mathrm{N}}(\mathbf{K}) = \alpha \frac{J_{\mathrm{KF}}(\mathbf{K})}{J^{\mathrm{Nadir}}_{\mathrm{KF}} - J^{\mathrm{Utopia}}_{\mathrm{KF}}} + \sum_p \gamma_p \frac{\bar{J}_{\mathrm{S},p}(\mathbf{K})}{\bar{J}^{\mathrm{Nadir}}_{\mathrm{S},p} - \bar{J}^{\mathrm{Utopia}}_{\mathrm{S},p}}, \tag{3.117}$$

$$J_{\mathrm{N}}(\mathbf{K}) = \nu_\alpha J_{\mathrm{KF}}(\mathbf{K}) + \sum_p \nu_{\gamma,p} J_{\mathrm{S},p}(\mathbf{K}), \tag{3.118}$$

where normalized weights $\nu_\alpha > 0, \nu_{\gamma,p} \geq 0$ are defined[2] as

$$\begin{aligned} \nu_\alpha &= \frac{\alpha}{J^{\mathrm{Nadir}}_{\mathrm{KF}} - J^{\mathrm{Utopia}}_{\mathrm{KF}}} + \sum_p \frac{\gamma_p(1-\bar{\gamma}_p)}{\bar{J}^{\mathrm{Nadir}}_{\mathrm{S},p} - \bar{J}^{\mathrm{Utopia}}_{\mathrm{S},p}}, \\ \nu_{\gamma,p} &= \frac{\gamma_p \bar{\gamma}_p}{\bar{J}^{\mathrm{Nadir}}_{\mathrm{S},p} - \bar{J}^{\mathrm{Utopia}}_{\mathrm{S},p}}. \end{aligned} \tag{3.119}$$

The weights $\nu_\alpha, \nu_{\gamma,p}$ do not add up to one. Therefore, the combination of objectives in (3.118) is not a convex combination in general. It could be modified into the convex combination, but it would not impact the performance since the weight ratio would not change. Nevertheless, the normalized objective (3.118) can be minimized the same way as the original objective, which results in the XDKF where the weights α, γ_p are replaced with the normalized weights $\nu_\alpha, \nu_{\gamma,p}$ respectively. The objective normalization can be applied to all proposed algorithms – XDKF, XDKF-Z, and XDKF-S. The algorithm modification is summed up in Algorithm 3.9.

[2] Paper [J1] includes a typo in the definition of ν_α. Here the correct definition is provided.

Algorithm 3.9 (XDKF/XDKF-Z/XDKF-S with normalized objectives).

- Given weights γ_p, stable weights $\bar{\gamma}_p$ and prior estimates $\hat{\mathbf{x}}_{k|k-1}$, $\mathbf{P}_{k|k-1}$, $\hat{\mathbf{s}}_{p,k|k-1}$.
- Obtain $\mathbf{K}^{\star}_{\mathrm{KF}}$ by temporary setting α to one and running XDKF/XDKF-Z/XDKF-S update step with the prior estimates.
- Obtain $\mathbf{K}^{\star}_{\mathrm{S},p}$ by temporary setting γ_p to $\bar{\gamma}_p$ and running XDKF/XDKF-Z/XDKF-S update step with the prior estimates.
- Evaluate utopia (3.115) and nadir (3.116) points.
- Obtain posterior estimates by temporary setting α, γ_p to $\nu_\alpha, \nu_{\gamma,p}$ (3.119) respectively, and running XDKF/XDKF-Z/XDKF-S update step with the prior estimates.

The range of tuning the Algorithm 3.9 using γ_p is increased with larger $\bar{\gamma}_p$. The stable weights need to be found numerically by checking the stability of candidate weights using Theorem 3.1.

The normalized weights $\nu_\alpha, \nu_{\gamma,p}$ need to be computed at each iteration. For some practical applications, it might be better to approximate the weights, e.g., by their mean value, and use them as fixed weights.

3.2.7 Numerical example

In this section, the performance of the algorithms is compared. The example used for comparison was previously used in [63–65] to compare state estimation algorithms for systems with uncertain parameters. The system is defined by the matrices

$$\mathbf{A}(\theta) = \begin{bmatrix} 0.9802 & 0.0196 + 0.099\,\theta \\ 0 & 0.9802 \end{bmatrix}, \mathbf{C} = \begin{bmatrix} 1 & -1 \end{bmatrix}, \tag{3.120}$$

and the noise covariances

$$\mathbf{Q} = \begin{bmatrix} 1.9608 & 0.0195 \\ 0.0195 & 1.9605 \end{bmatrix}, \mathbf{R} = \begin{bmatrix} 1 \end{bmatrix}. \tag{3.121}$$

The value of the uncertain parameter θ is expected to be within the interval $[-1, 1]$. The real parameter value is fixed during each simulation. The parameter value 0 is used for the models in all algorithms except the optimal Kalman filter, where the real parameter value is used. The algorithms use the initial conditions in simulations with the following values

$$\mathbf{x}_0 = \mathbf{0}, \hat{\mathbf{x}}_{0|-1} = \begin{bmatrix} 1 & 1 \end{bmatrix}^T, \mathbf{P}^{xx}_{0|-1} = \mathbf{P}^{xs}_{0|-1} = \mathbf{P}^{sx}_{0|-1} = \mathbf{P}^{ss}_{0|-1} = \mathbf{I}. \tag{3.122}$$

The simulations consisted of the response to initial conditions and lasted 1000 samples. Seven different real parameter values were used in simulations. The set of 100 Monte Carlo simulations was done for each real parameter value.

All simulation results with algorithm comparison can be repeated using the MATLAB-based application SESUP [O2] created to demonstrate the performance of the algorithms derived in this thesis. The SESUP application allows users to modify the algorithm configuration and see comparison figures.

Optimal vs. suboptimal. The optimal version XDKF differs from the suboptimal XDKF-Z only by assuming nonzero gain sensitivity to parameters. The optimal gain of the XDKF can be compared to the suboptimal gain of the XDKF-Z at each iteration using simulations. Figure 3.6 shows the first 200 samples of response to initial conditions averaged over 100 Monte Carlo simulations. The figure shows the comparison of the XDKF, the XDKF-Z, their versions with normalized objectives (XDKF-N, XDKF-ZN), and the XDKF-S. It can be seen that the difference between the gains caused by

Assumption 3.2 is very small. The small difference can be explained by the small gain sensitivity of the XDKF, which justifies Assumption 3.2. On the other hand, normalizing the objectives has a major impact on the gain. The XDKF-S is the lightweight version of the XDKF-Z, which gains can be precomputed offline. The results depicted in Figure 3.6 show that the gains differ mainly during the initial response. The XDKF-S uses steady-state gains, which explains the difference in the initial response. After the initial response, the XDKF-S gain is comparable to the XDKF-Z and XDKF. Similar behavior can be observed when comparing the standard Kalman filter to its steady-state version.

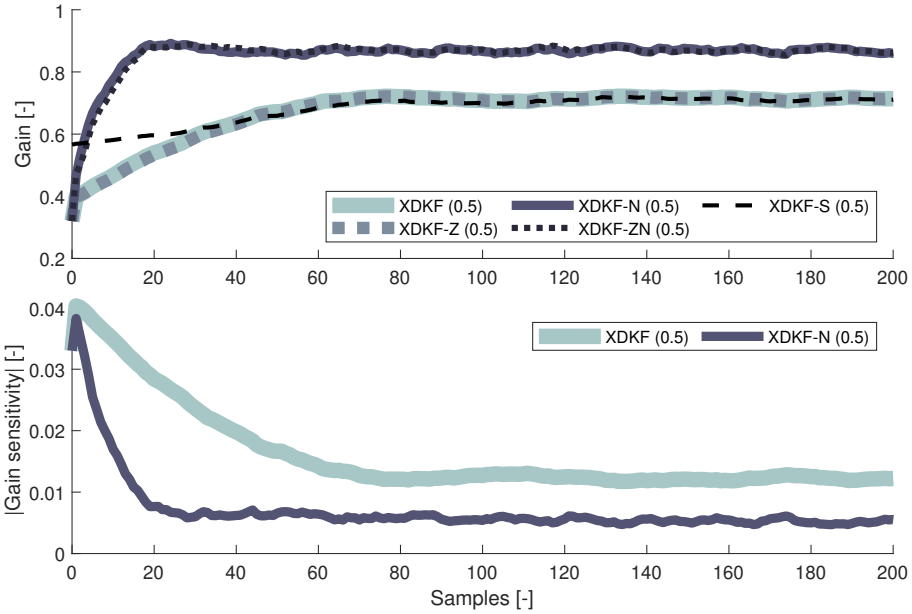

Figure 3.6. The optimal desensitized gain for $\gamma = 0.5$, absolute value of its sensitivity, and the suboptimal gains. The figure shows the result of the simulation for real parameter value 1. The parameters in brackets denote weight setting.

Normalized vs. raw objectives. Figure 3.7 compares the KF, the XDKF, and the XDKF with normalized objectives (XDKF-N). The KF was used with the model with zero parameter value. The XDKF-N is normalized with $\bar{\gamma} = 0.99$. The root-mean-square error (RMSE) is more consistent with normalized objectives, meaning that the XDKF-N reduces the state-estimate error dependency on the parameter better. The XDKF-N achieves lower sensitivity to the parameter than XDKF while using the same weight setting, which can be seen in Figure 3.6, i.e., the objectives normalization improves robustness to the parameter error. Also, because of the higher error covariance $\mathbf{P}^{xx}$, the XDKF-N reaches a higher gain and consequently responds faster than the XDKF. The XDKF weight could be modified to normalize the objectives and achieve a performance similar to the XDKF-N. In this example, it must be set much closer to 1. However, the XDKF with fixed normalization cannot achieve the XDKF-N performance because the XDKF-N normalizes the objectives adaptively at each step.

Tuning. The XDKF and its suboptimal versions are tuned using the weight γ. With $\gamma = 0$, the XDKF is reduced to the standard KF. On the other hand, the sensitivity to the parameter is reduced by increasing the weight γ. The flat curve of the average RMSE means low dependency on the parameter value, hence the robustness to the parameter. As expected, minimizing the variance is not sufficient for achieving minimal

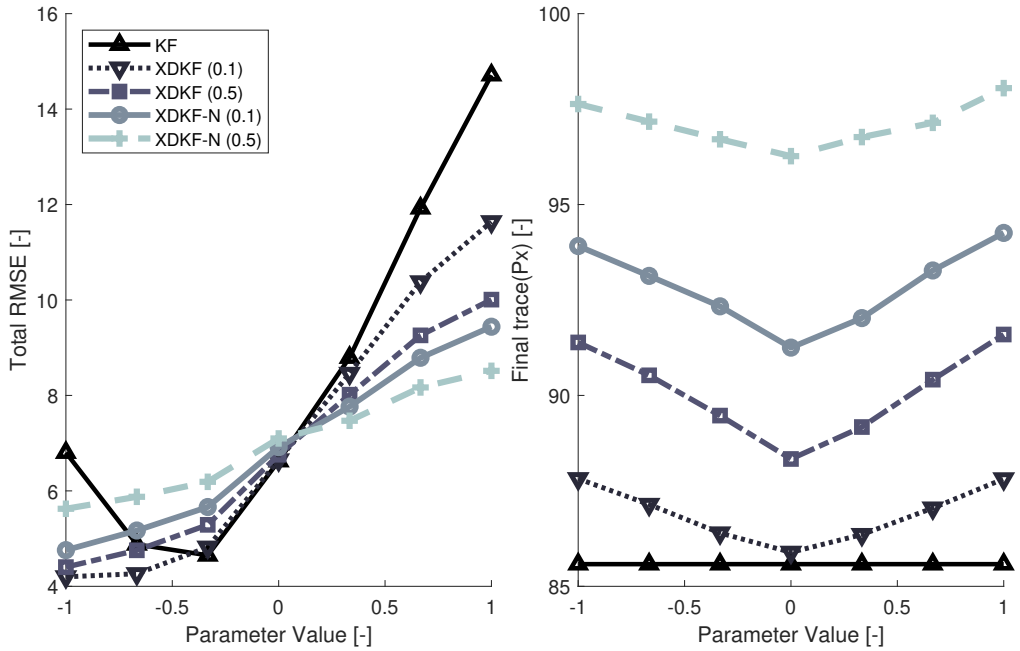

Figure 3.7. Total RMSE and final state estimation error covariance trace. Comparison of the standard KF, XDKF, and its normalized form. The γ weight value is shown in brackets. The values are averaged over 100 Monte Carlo simulations.

state estimation error if the model differs from the real system, which can be observed in Figure 3.7 where the XDKF achieves higher variance than the standard KF. The weight setting choice is tailored to the particular application and the filtering goal.

XDKF vs. state-of-the-art. Finally, the XDKF is compared to the state-of-the-art state estimation algorithms. The H_∞ filter [11] is used for comparison as the algorithm for systems with uncertain parameters. The H_∞ was used with the configuration used by Sayed [64]. The optimal KF with the accurate model parameter and the KF with the mean model parameter were tested as best and worst-case scenarios benchmarks. The optimal Kalman filter was used with the model identical to the system used for generating data. The Kalman filter was used with the model with zero parameter value. The DKF and novel XDKF were tested in a single-weight configuration, chosen subjectively as a trade-off between reducing the sensitivity and keeping low volatility of the estimates. The performance of the XDKF and the DKF can be directly compared using the corresponding weight configuration. The corresponding DKF weights are obtained by the diagonal matrix $W = \mathrm{diag}([w, w])$, with values $w = \frac{\gamma}{\alpha}$ where γ, α are the XDKF weights. For example, $\gamma = 0.5$ corresponds to $w = 1$.

The comparison between algorithms is made by comparing their average root-mean-square error (RMSE) from 100 Monte Carlo simulations. Figure 3.8 shows the test results. Overall the DKF, the XDKF, and the XDKF-N have lower RMSE than the KF or the H_∞. Furthermore, their error is the most consistent, meaning they are the best in reducing the dependency of the state-estimate error on the parameter. Another important property of desensitized algorithms is that their performance can be intuitively tuned. It means that in this example, choosing weights from zero to maximum weight would move their curve in Figure 3.8 from the curve that represents the results of the KF to an almost flat curve.

The XDKF-N outperforms other desensitized Kalman filters, showing the importance of normalized objectives in multi-objective optimization. The results of the suboptimal

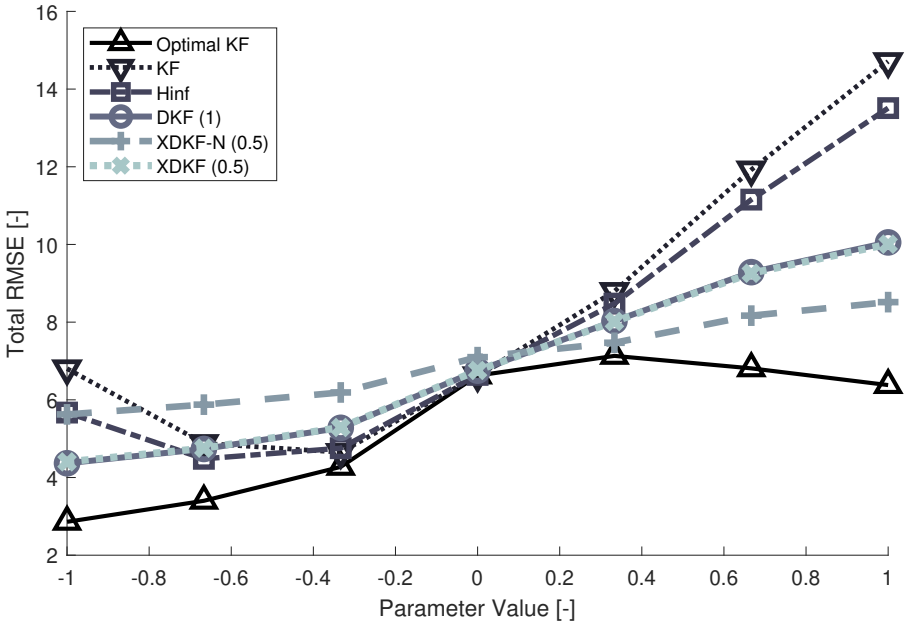

Figure 3.8. Total RMSE comparison of the XDKF and state-of-the-art algorithms. The value in brackets represents the weight value. The mean total RMSE from 100 Monte Carlo simulations for each real parameter value is shown.

versions of the XDKF are not shown since their performance is very similar to the XDKF, which would significantly deteriorate the clarity of the figure. The fact that the performance of the suboptimal algorithms is almost indistinguishable from the optimal is a remarkable result, which implies that the performance of simple gain-scheduled XDKF-S is comparable to or better than robust state-of-the-art algorithms with a significant computational burden. The results for the suboptimal algorithms can be validated using the SESUP application [O2].

3.3 Discussion

The desensitized Kalman filter (DKF) was initially derived by Karlgaard [17]. The DKF sensitivity is defined as the sensitivity of the state estimation error $\tilde{\mathbf{x}}$ to the expected parameter value $\hat{\theta}$. However, this definition is instantly reduced to the sensitivity of the state estimation mean $\hat{\mathbf{x}}$ to $\hat{\theta}$ since the real state value is assumed to be independent of parameters.

For the XDKF, the initial definition of sensitivity is the same, but the real state value is not omitted in the next step. The rationale is that the prediction error is a random variable that represents most of the valuable stochastic properties of the Kalman filter, e.g., optimality, and bias. By reducing the prediction error to a state mean prediction, the opportunity to take these properties into account is lost, and eventually, the problem is changed to desensitizing the algorithm rather than the state estimation error. These arguments are reflected in the derived sensitivity forms. The sensitivity is a deterministic variable in the DKF and a random variable in the XDKF. The sensitivity definition closely relates to the augmentation of the optimization criterion. In the DKF, the augmentation is done using a weighted sensitivity norm due to the deterministic sensitivity formulation. The XDKF sensitivity is formulated as a random variable, so the augmentation is defined by the trace of sensitivity second moments.

Regarding the complexity, the original DKF algorithm requires solving an implicit algebraic equation at each iteration step to obtain the gain, which creates a significant computational burden compared to the explicit gain evaluation. The DKF in the special case where the gain is expressed explicitly was published [C2]. The gain in the XDKF is also expressed explicitly.

The DKF assumes zero gain sensitivity, formulated in Assumption 3.1, which is equivalent to Assumption 3.2. Shen and Kaarlgard [26] admit in their remark that the assumption is not always correct, but it is necessary to derive the DKF algorithm. When this assumption is used on the XDKF, the XDKF-Z is obtained. Interestingly, the XDKF-Z can be reinterpreted as a special case of the DKF (SDKF) in Algorithm 3.2 applied to the system (3.39). In order to derive the SDKF, the original DKF weighting must be limited – the DKF allows to set the weight on the sensitivity of each state estimation error to parameter individually, whereas the SDKF weights are the same for all state estimation errors. Despite similarities between the XDKF-Z and the SDKF, the XDKF-Z is denoted by a separate name to emphasize the inheritance from the XDKF, which enables interpreting the XDKF-Z as the Kalman filter with the time-variable correlated process and measurement noise.

A thorough stability analysis of DKF is yet to be published. However, the DKF and the XDKF have a common problem with weakening the minimum-variance criterion, resulting in a potentially unstable estimator. The stability conditions of the XDKF-S are formulated in Theorem 3.4, and in the future, they can be used to define the stability conditions of the XDKF-Z.

The overview of the differences between the desensitized algorithms is shown in Table 3.1.

	Assumption 3.1 or 3.2	Gain	Sensitivity	Normalized
XDKF	no	explicit	stochastic	optional
XDKF-Z	yes	explicit	deterministic	optional
XDKF-S	yes	precomputed	deterministic	optional
DKF-NC	yes	implicit	deterministic	no
SDKF-NC	yes	explicit	deterministic	no

Table 3.1. Overview of the properties of desensitized filters.

Chapter 4
Desensitized state estimation of nonlinear systems

There are many methods for nonlinear filtering based on the Kalman filter. The simplest one uses the first order Taylor approximation of a nonlinear function. This approximation is not suitable for highly nonlinear systems but is satisfactory for many nonlinear systems. The main benefit over more advanced nonlinear filters is its similarity to the linear Kalman filter and low computational burden. In this chapter, the desensitized algorithms for linear systems are rederived for nonlinear systems using the first order Taylor approximation. First, DKF-NC and SDKF-NC, which use the deterministic approach to sensitivity reduction, are extended. Then, algorithms with the stochastic approach to sensitivity reduction are derived.

4.1 Algorithms for systems with noise correlation

In this section, the DKF-NC and the SDKF-NC, which follow the original desensitized approach, are rederived for nonlinear systems. The extensions follow the extended Kalman filter algorithm, which uses first order Taylor series approximation in a nominal point. The derivations are similar to the derivation for linear systems. The results in this section were published in [C2].

4.1.1 Nonlinear system with noise correlation

Let's define a discrete-time nonlinear stochastic system

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\theta}\right) + \mathbf{v}_k, \\ \mathbf{y}_k &= \mathbf{g}\left(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\theta}\right) + \mathbf{e}_k, \end{aligned} \tag{4.1}$$

where $\mathbf{x}_k$ is the system state vector of dimension N_x at time k, $\mathbf{y}_k$ is the system output vector of dimension N_y at time k, $\mathbf{u}_k$ is the deterministic system input vector of dimension N_u at time k, $\boldsymbol{\theta}$ is the parameter vector of dimension N_θ, $\mathbf{f}$ is the vector-valued function of state propagation, $\mathbf{g}$ is the vector-valued function of system output, and $\mathbf{v}_k, \mathbf{e}_k$ are process and measurement noise sequences with statistics

$$\begin{bmatrix} \mathbf{v}_k \\ \mathbf{e}_k \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{Q} & \mathbf{S} \\ \mathbf{S}^T & \mathbf{R} \end{bmatrix} \right). \tag{4.2}$$

The update equations of the single-step extended Kalman filter in Joseph's form are as follows

$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{f}\left(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}\right) + \mathbf{K}_k \left(\mathbf{y} - \mathbf{g}\left(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}\right)\right), \tag{4.3}$$

$$\begin{aligned} \mathbf{P}_{k+1|k} = {} & \left(\mathbf{F}_x - \mathbf{K}_k \mathbf{G}_x\right) \mathbf{P}_{k|k-1} \left(\mathbf{F}_x - \mathbf{K}_k \mathbf{G}_x\right)^T + \\ & + \mathbf{Q} + \mathbf{K}_k \mathbf{R} \mathbf{K}_k^T - \mathbf{S}\mathbf{K}_k^T - \mathbf{K}_k \mathbf{S}^T, \end{aligned} \tag{4.4}$$

where Jacobians are defined as

$$\mathbf{F}_x = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}_k}\right|_{\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}}, \quad \mathbf{G}_x = \left.\frac{\partial \mathbf{g}}{\partial \mathbf{x}_k}\right|_{\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}}, \tag{4.5}$$

and $\mathbf{K}_k$ is the optimal Kalman gain that minimizes the trace of the state estimation error covariance $\mathbf{P}_{k+1|k}$

$$\mathbf{K}_k = \left(\mathbf{F}_x \mathbf{P}_{k|k-1} \mathbf{G}_x^T + \mathbf{S}\right)\left(\mathbf{G}_x \mathbf{P}_{k|k-1} \mathbf{G}_x^T + \mathbf{R}\right)^{-1}. \tag{4.6}$$

4.1.2 Extended DKF-NC

The state error sensitivity to the particular parameter is computed similarly to the sensitivity in the linear DKF (3.17), but now it is computed using the state estimate (4.3)

$$\boldsymbol{\sigma}_{p,k+1|k} = \frac{d\hat{\mathbf{x}}_{k+1|k}}{d\hat{\theta}_p} = \boldsymbol{\xi}_{p,k} - \mathbf{K}_k \boldsymbol{\gamma}_{p,k}, \tag{4.7}$$

where the index p denotes the particular parameter in the parameter vector and

$$\boldsymbol{\xi}_{p,k} = \mathbf{F}_{x,k}\boldsymbol{\sigma}_{p,k|k-1} + \mathbf{f}_{\theta,p}, \tag{4.8}$$

$$\boldsymbol{\gamma}_{p,k} = \mathbf{G}_{x,k}\boldsymbol{\sigma}_{p,k|k-1} + \mathbf{g}_{\theta,p}. \tag{4.9}$$

and Jacobians

$$\mathbf{f}_{\theta,p} = \left.\frac{\partial \mathbf{f}}{\partial \theta_p}\right|_{\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}}, \quad \mathbf{g}_{\theta,p} = \left.\frac{\partial \mathbf{g}}{\partial \theta_p}\right|_{\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}}. \tag{4.10}$$

The optimality criterion is the same as in the linear DKF-NC

$$\min_{\mathbf{K}_k} J(\mathbf{K}_k), \; J(\mathbf{K}_k) = \operatorname{tr}\left(\mathbf{P}_{k+1|k}\right) + \sum_{p=1}^{N_\theta}\left(\boldsymbol{\sigma}_{p,k+1|k}^T \mathbf{W}_p \boldsymbol{\sigma}_{p,k+1|k}\right), \tag{4.11}$$

Since the nonlinear version differs from the linear only in the model sensitivity formulas (4.8), the solution of the optimality criterion leads to the same gain equation

$$\begin{aligned}\mathbf{K}_k\left(\mathbf{G}_{x,k}\mathbf{P}_{k|k-1}\mathbf{G}_{x,k}^T + \mathbf{R}\right) + \sum_{p=1}^{N_\theta}\left(\mathbf{W}_p \mathbf{K}_k \boldsymbol{\gamma}_{p,k}\boldsymbol{\gamma}_{p,k}^T\right) = \\ = \mathbf{F}_{x,k}\mathbf{P}_{k|k-1}\mathbf{G}_{x,k}^T + \mathbf{S} + \sum_{p=1}^{N_\theta}\left(\mathbf{W}_p \boldsymbol{\xi}_{p,k}\boldsymbol{\gamma}_{p,k}^T\right).\end{aligned} \tag{4.12}$$

The final algorithm is summed up in Algorithm 4.1.

Algorithm 4.1 (Extended DKF for systems with noise correlation — EDKF-NC).

- Evaluate Jacobians

$$\mathbf{F}_x = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}_k}\right|_{\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}}, \mathbf{f}_{\theta,p} = \left.\frac{\partial \mathbf{f}}{\partial \theta_p}\right|_{\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}},$$

$$\mathbf{G}_x = \left.\frac{\partial \mathbf{g}}{\partial \mathbf{x}_k}\right|_{\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}}, \mathbf{g}_{\theta,p} = \left.\frac{\partial \mathbf{g}}{\partial \theta_p}\right|_{\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}}.$$

- Evaluate model sensitivities

$$\begin{aligned}\boldsymbol{\xi}_{p,k} &= \mathbf{F}_{x,k}\boldsymbol{\sigma}_{p,k|k-1} + \mathbf{f}_{\theta,p},\\ \boldsymbol{\gamma}_{p,k} &= \mathbf{G}_{x,k}\boldsymbol{\sigma}_{p,k|k-1} + \mathbf{g}_{\theta,p}.\end{aligned}$$

- Solve for $\mathbf{K}_k$

$$\begin{aligned}\mathbf{K}_k\left(\mathbf{G}_{x,k}\mathbf{P}_{k|k-1}\mathbf{G}_{x,k}^T + \mathbf{R}\right) + \sum_{p=1}^{N_\theta}\left(\mathbf{W}_p\mathbf{K}_k\boldsymbol{\gamma}_{p,k}\boldsymbol{\gamma}_{p,k}^T\right) =\\ = \mathbf{F}_{x,k}\mathbf{P}_{k|k-1}\mathbf{G}_{x,k}^T + \mathbf{S} + \sum_{p=1}^{N_\theta}\left(\mathbf{W}_p\boldsymbol{\xi}_{p,k}\boldsymbol{\gamma}_{p,k}^T\right).\end{aligned}$$

- Update the state, sensitivities, and covariance

$$\begin{aligned}\boldsymbol{\sigma}_{p,k+1|k} &= \boldsymbol{\xi}_{p,k} - \mathbf{K}_k\boldsymbol{\gamma}_{p,k},\\ \hat{\mathbf{x}}_{k+1|k} &= \mathbf{f}\left(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}\right) + \mathbf{K}_k\left(\mathbf{y} - \mathbf{g}\left(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}\right)\right),\\ \mathbf{P}_{k+1|k} &= \left(\mathbf{F}_x - \mathbf{K}_k\mathbf{G}_x\right)\mathbf{P}_{k|k-1}\left(\mathbf{F}_x - \mathbf{K}_k\mathbf{G}_x\right)^T +\\ &\quad + \mathbf{Q} + \mathbf{K}_k\mathbf{R}\mathbf{K}_k^T - \mathbf{S}\mathbf{K}_k^T - \mathbf{K}_k\mathbf{S}^T.\end{aligned}$$

4.1.3 Extended SDKF-NC

The state error vector sensitivity to the parameter vector is defined similarly to (3.24). It is defined in the form of the sensitivity matrix

$$\boldsymbol{\Sigma}_{k+1|k} = \frac{d\hat{\mathbf{x}}_{k+1|k}}{d\hat{\boldsymbol{\theta}}} = \boldsymbol{\Xi}_k - \mathbf{K}_k\boldsymbol{\Gamma}_k, \tag{4.13}$$

where

$$\boldsymbol{\Xi}_k = \mathbf{F}_x\boldsymbol{\Sigma}_{k|k-1} + \mathbf{F}_{\theta,k}, \tag{4.14}$$

$$\boldsymbol{\Gamma}_k = \mathbf{G}_x\boldsymbol{\Sigma}_{k|k-1} + \mathbf{G}_\theta, \tag{4.15}$$

and Jacobians are defined as

$$\mathbf{F}_\theta = \left.\frac{\partial\mathbf{f}}{\partial\boldsymbol{\theta}}\right|_{\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}}, \mathbf{G}_\theta = \left.\frac{\partial\mathbf{g}}{\partial\boldsymbol{\theta}}\right|_{\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}}. \tag{4.16}$$

The optimality criterion is the same as the criterion in the SDKF-NC, which is

$$J = \operatorname{tr}\left(\mathbf{P}_{k+1|k}\right) + \operatorname{tr}\left(\boldsymbol{\Sigma}_{k+1|k}\mathbf{W}\boldsymbol{\Sigma}_{k+1|k}^T\right). \tag{4.17}$$

The derivation of the optimal gain is also identical to the SDKF-NC. The optimal gain which results from the minimization of (4.17) is

$$\mathbf{K}_k = \left(\mathbf{F}_x\mathbf{P}_{k|k-1}\mathbf{G}_x^T + \mathbf{S} + \boldsymbol{\Xi}_k\mathbf{W}\boldsymbol{\Gamma}_k^T\right)\left(\mathbf{G}_x\mathbf{P}_{k|k-1}\mathbf{G}_x^T + \mathbf{R} + \boldsymbol{\Gamma}_k\mathbf{W}\boldsymbol{\Gamma}_k^T\right)^{-1}. \tag{4.18}$$

The optimal gain (4.18) can be substituted into the covariance update equation (4.4), which will again give the alternative covariance update

$$\begin{aligned}\mathbf{P}_{k+1|k} &= \mathbf{F}_x\mathbf{P}_{k|k-1}\mathbf{F}_x^T + \mathbf{Q} - \mathbf{K}_k\left(\mathbf{G}_x\mathbf{P}_{k|k-1}\mathbf{G}_x^T + \mathbf{S}^T\right) +\\ &\quad + \boldsymbol{\Sigma}_{k+1|k}\mathbf{W}\boldsymbol{\Gamma}_k^T\mathbf{K}_k^T.\end{aligned} \tag{4.19}$$

The Extended SDKF-NC (ESDKF-NC) algorithm consists of the update equations (4.3), (4.4), and (4.13) where $\mathbf{K}_k$ is defined in (4.18). The algorithm is summed up in Algorithm 4.2.

Algorithm 4.2 (Extended SDKF for systems with noise correlation — ESDKF-NC).

- Evaluate Jacobians

$$\mathbf{F}_x = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}_k}\right|_{\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}}, \mathbf{F}_\theta = \left.\frac{\partial \mathbf{f}}{\partial \boldsymbol{\theta}}\right|_{\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}},$$
$$\mathbf{G}_x = \left.\frac{\partial \mathbf{g}}{\partial \mathbf{x}_k}\right|_{\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}}, \mathbf{G}_\theta = \left.\frac{\partial \mathbf{g}}{\partial \boldsymbol{\theta}}\right|_{\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}}.$$

- Evaluate model sensitivities

$$\begin{aligned}\boldsymbol{\Xi}_k &= \mathbf{F}_x\boldsymbol{\Sigma}_{k|k-1} + \mathbf{F}_\theta,\\ \boldsymbol{\Gamma}_k &= \mathbf{G}_x\boldsymbol{\Sigma}_{k|k-1} + \mathbf{G}_\theta.\end{aligned}$$

- Compute gain

$$\mathbf{K}_k = \left(\mathbf{F}_x\mathbf{P}_{k|k-1}\mathbf{G}_x^T + \mathbf{S} + \boldsymbol{\Xi}_k\mathbf{W}\boldsymbol{\Gamma}_k^T\right)\left(\mathbf{G}_x\mathbf{P}_{k|k-1}\mathbf{G}_x^T + \mathbf{R} + \boldsymbol{\Gamma}_k\mathbf{W}\boldsymbol{\Gamma}_k^T\right)^{-1}.$$

- Update the state, sensitivity, and covariance

$$\begin{aligned}\boldsymbol{\Sigma}_{k+1|k} &= \boldsymbol{\Xi}_k - \mathbf{K}_k\boldsymbol{\Gamma}_k,\\ \hat{\mathbf{x}}_{k+1|k} &= \mathbf{f}\left(\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}\right) + \mathbf{K}_k\left(\mathbf{y} - \mathbf{g}\left(\hat{\mathbf{x}}_{k|k-1},\mathbf{u}_k,\hat{\boldsymbol{\theta}}\right)\right),\\ \mathbf{P}_{k+1|k} &= \left(\mathbf{F}_x - \mathbf{K}_k\mathbf{G}_x\right)\mathbf{P}_{k|k-1}\left(\mathbf{F}_x - \mathbf{K}_k\mathbf{G}_x\right)^T +\\ &\quad + \mathbf{Q} + \mathbf{K}_k\mathbf{R}\mathbf{K}_k^T - \mathbf{S}\mathbf{K}_k^T - \mathbf{K}_k\mathbf{S}^T,\end{aligned}$$

or

$$\mathbf{P}_{k+1|k} = \mathbf{F}_x\mathbf{P}_{k|k-1}\mathbf{F}_x^T + \mathbf{Q} - \mathbf{K}_k\left(\mathbf{G}_x\mathbf{P}_{k|k-1}\mathbf{G}_x^T + \mathbf{S}^T\right) + \boldsymbol{\Sigma}_{k+1|k}\mathbf{W}\boldsymbol{\Gamma}_k^T\mathbf{K}_k^T.$$

4.2 Extended exact DKF

This section summarizes the results published in [C4]. The ideas from exact desensitized Kalman filter are extended to nonlinear systems using Taylor series approximation. The result is the extended exact desensitized Kalman filter (EXDKF). Furthermore, the adaptive weights that improve the performance by normalizing the EXDKF objectives are introduced. Interestingly, the EXDKF differs from the XDKF due to the difference in the state estimation error definition caused by the nonlinear function approximation method. Consequently, the EXDKF applied to the linear system differs from the XDKF.

4.2.1 Nonlinear system

Assume a discrete-time nonlinear stochastic system defined in (4.1) but without the correlation between the process and measurement noise

$$\begin{aligned}\mathbf{x}_{k+1} &= \mathbf{f}\left(\mathbf{x}_k,\mathbf{u}_k,\boldsymbol{\theta}\right) + \mathbf{v}_k,\\ \mathbf{y}_k &= \mathbf{g}\left(\mathbf{x}_k,\mathbf{u}_k,\boldsymbol{\theta}\right) + \mathbf{e}_k,\end{aligned} \tag{4.20}$$

where $\mathbf{x}_k$ is the system state vector of dimension N_x at time k, $\mathbf{y}_k$ is the system output vector of dimension N_y at time k, $\mathbf{u}_k$ is the deterministic system input vector of dimension N_u at time k, $\boldsymbol{\theta}$ is the parameter vector of dimension N_θ, $\mathbf{f}$ is the vector-valued

function of state propagation, $\mathbf{g}$ is the vector-valued function of system output, and $\mathbf{v}_k, \mathbf{e}_k$ are white noise sequences with statistics

$$\begin{bmatrix} \mathbf{v}_k \\ \mathbf{e}_k \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{bmatrix} \right). \tag{4.21}$$

The nonlinear system (4.20) can be linearized using first order Taylor series approximation in a nominal point $\mathbf{x}_\mathrm{n}, \mathbf{u}_\mathrm{n}, \boldsymbol{\theta}_\mathrm{n}$

$$\begin{aligned} \mathbf{x}_{k+1} \approx \mathbf{f}(\mathbf{x}_\mathrm{n}, \mathbf{u}_\mathrm{n}, \boldsymbol{\theta}_\mathrm{n}) + \mathbf{v}_k + \left.\frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\theta})}{\partial \mathbf{x}_k}\right|_{\mathbf{x}_\mathrm{n}, \mathbf{u}_\mathrm{n}, \boldsymbol{\theta}_\mathrm{n}} (\mathbf{x}_k - \mathbf{x}_\mathrm{n}) + \\ + \left.\frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\theta})}{\partial \mathbf{u}_k}\right|_{\mathbf{x}_\mathrm{n}, \mathbf{u}_\mathrm{n}, \boldsymbol{\theta}_\mathrm{n}} (\mathbf{u}_k - \mathbf{u}_\mathrm{n}) + \left.\frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right|_{\mathbf{x}_\mathrm{n}, \mathbf{u}_\mathrm{n}, \boldsymbol{\theta}_\mathrm{n}} (\boldsymbol{\theta} - \boldsymbol{\theta}_\mathrm{n}), \end{aligned} \tag{4.22}$$

$$\begin{aligned} \mathbf{y}_k \approx \mathbf{g}(\mathbf{x}_\mathrm{n}, \mathbf{u}_\mathrm{n}, \boldsymbol{\theta}_\mathrm{n}) + \mathbf{e}_k + \left.\frac{\partial \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\theta})}{\partial \mathbf{x}_k}\right|_{\mathbf{x}_\mathrm{n}, \mathbf{u}_\mathrm{n}, \boldsymbol{\theta}_\mathrm{n}} (\mathbf{x}_k - \mathbf{x}_\mathrm{n}) + \\ + \left.\frac{\partial \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\theta})}{\partial \mathbf{u}_k}\right|_{\mathbf{x}_\mathrm{n}, \mathbf{u}_\mathrm{n}, \boldsymbol{\theta}_\mathrm{n}} (\mathbf{u}_k - \mathbf{u}_\mathrm{n}) + \left.\frac{\partial \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right|_{\mathbf{x}_\mathrm{n}, \mathbf{u}_\mathrm{n}, \boldsymbol{\theta}_\mathrm{n}} (\boldsymbol{\theta} - \boldsymbol{\theta}_\mathrm{n}). \end{aligned} \tag{4.23}$$

To design an observer for system (4.22), the nominal point for linearization needs to be selected. The linearization accuracy increase with the nominal point closer to the real system state. Therefore, the standard practice is to linearize in the best available estimate. In this case it is $\hat{\mathbf{x}}_{k|k-1}$ which is the state estimation mean, $\mathbf{u}_k$ which is the known input, and $\hat{\boldsymbol{\theta}}$ which is the estimated value of parameter vector. Usually, the best available state vector point for the time propagation equation is the posterior estimate $\hat{\mathbf{x}}_{k|k}$. However, here the single step algorithm is derived, where the posterior estimate is not explicitly evaluated for clarity. Then the linearized system is obtained as

$$\begin{aligned} \mathbf{x}_{k+1} &\approx \mathbf{f}\left(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}\right) + \mathbf{v}_k + \mathbf{F}_x \tilde{\mathbf{x}}_{k|k-1} + \mathbf{F}_\theta \tilde{\boldsymbol{\theta}}, \\ \mathbf{y}_k &\approx \mathbf{g}\left(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}\right) + \mathbf{e}_k + \mathbf{G}_x \tilde{\mathbf{x}}_{k|k-1} + \mathbf{G}_\theta \tilde{\boldsymbol{\theta}}, \end{aligned} \tag{4.24}$$

where the Jacobians are defined as

$$\begin{aligned} \mathbf{F}_x &\equiv \left.\frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\theta})}{\partial \mathbf{x}_k}\right|_{\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}}, \quad \mathbf{F}_\theta \equiv \left.\frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right|_{\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}}, \\ \mathbf{G}_x &\equiv \left.\frac{\partial \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\theta})}{\partial \mathbf{x}_k}\right|_{\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}}, \quad \mathbf{G}_\theta \equiv \left.\frac{\partial \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right|_{\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}}, \end{aligned} \tag{4.25}$$

the state estimation error is obtained as

$$\tilde{\mathbf{x}}_{k|k-1} = \mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}, \tag{4.26}$$

and the true value of the parameter vector $\boldsymbol{\theta}$ is defined as

$$\boldsymbol{\theta} = \hat{\boldsymbol{\theta}} + \tilde{\boldsymbol{\theta}}, \tag{4.27}$$

where $\tilde{\boldsymbol{\theta}}$ denotes the parameter deviation, which is assumed to be unknown.

The expected values of system (4.24) are obtained by evaluating the system in the best available values of state and parameter vectors.

$$\begin{aligned} \hat{\mathbf{x}}_{k+1|k-1} &= \mathbf{f}\left(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}\right), \\ \hat{\mathbf{y}}_{k|k-1} &= \mathbf{g}\left(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}\right). \end{aligned} \tag{4.28}$$

Note that first order approximation is used. Then the observer designed for system (4.24) is designed as

$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{f}\left(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}\right) + \mathbf{K}_k\left(\mathbf{y}_k - \mathbf{g}\left(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}\right)\right), \tag{4.29}$$

which creates the state estimation error

$$\begin{aligned}\tilde{\mathbf{x}}_{k+1|k} &= \mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k} && (4.30)\\ &= \mathbf{F}_x\tilde{\mathbf{x}}_{k|k-1} + \mathbf{F}_\theta\tilde{\boldsymbol{\theta}} + \mathbf{v}_k - \mathbf{K}_k\left(\mathbf{G}_x\tilde{\mathbf{x}}_{k|k-1} + \mathbf{G}_\theta\tilde{\boldsymbol{\theta}} + \mathbf{e}_k\right), && (4.31)\end{aligned}$$

with the covariance matrix of state estimation error obtained as

$$\begin{aligned}\mathbf{P}^{xx}_{k+1|k} &= \mathbb{E}\left[\tilde{\mathbf{x}}_{k+1|k}\tilde{\mathbf{x}}^T_{k+1|k}\right], && (4.32)\\ &= \left(\mathbf{F}_x - \mathbf{K}_k\mathbf{G}_x\right)\mathbf{P}^{xx}_{k|k-1}\left(\mathbf{F}_x - \mathbf{K}_k\mathbf{G}_x\right)^T + \mathbf{Q} + \mathbf{K}_k\mathbf{R}\mathbf{K}^T_k. && (4.33)\end{aligned}$$

The sensitivity of the state estimation error to the parameter $\hat{\theta}_p$ is obtained as

$$\begin{aligned}\mathbf{s}_{p,k+1|k} &\equiv \frac{d\tilde{\mathbf{x}}_{k+1|k}}{d\hat{\theta}_p}, && (4.34)\\ &= \mathbf{F}_{x,\theta,p}\tilde{\mathbf{x}}_{k|k-1} + \mathbf{F}_x\mathbf{s}_{p,k|k-1} - \mathbf{f}_{\theta,p}\\ &\quad - \mathbf{K}_k\left(\mathbf{G}_{x,\theta,p}\tilde{\mathbf{x}}_{k|k-1} + \mathbf{G}_x\mathbf{s}_{p,k|k-1} - \mathbf{g}_{\theta,p}\right)\\ &\quad - \frac{\partial\mathbf{K}_k}{\partial\hat{\theta}_p}\left(\mathbf{G}_x\tilde{\mathbf{x}}_{k|k-1} + \mathbf{G}_{\theta,p}\tilde{\boldsymbol{\theta}} + \mathbf{e}_k\right), && (4.35)\end{aligned}$$

where

$$\begin{aligned}\mathbf{F}_{x,\theta,p} &= \frac{\partial\mathbf{F}_x}{\partial\hat{\theta}_p},\ \mathbf{f}_{\theta,p} \equiv \left.\frac{\partial\mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\theta}\right)}{\partial\theta_p}\right|_{\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}},\\ \mathbf{G}_{x,\theta,p} &= \frac{\partial\mathbf{G}_x}{\partial\hat{\theta}_p},\ \mathbf{g}_{\theta,p} \equiv \left.\frac{\partial\mathbf{g}\left(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\theta}\right)}{\partial\theta_p}\right|_{\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k, \hat{\boldsymbol{\theta}}}.\end{aligned} \tag{4.36}$$

It is important to note that the sensitivity is defined with the assumption of linear dependency of the system on a parameter, i.e.,

$$\frac{\partial\mathbf{F}_\theta}{\partial\hat{\theta}_p} = \mathbf{0}. \tag{4.37}$$

This condition is typically not satisfied when $\mathbf{f}$ depends on the parameter exponentially. Omitting this assumption is possible, but further approximation would be required. Therefore, the results in this section can be applied only to nonlinear systems which satisfy this assumption.

The gain sensitivity $(\partial\mathbf{K}_k)/(\partial\hat{\theta}_p)$ is generally not zero. However, for most systems, the assumption of zero gain sensitivity is valid. Therefore, Assumption 3.2 stating the zero gain sensitivity is used in the following derivation. For further reference on the impact of the gain sensitivity, see Section 3.2. The sensitivity definition with zero gain assumption is reduced to

$$\begin{aligned}\mathbf{s}_{p,k+1|k} &= \left(\mathbf{F}_{x,\theta,p} - \mathbf{K}_k\mathbf{G}_{x,\theta,p}\right)\tilde{\mathbf{x}}_{k|k-1} + \left(\mathbf{F}_x - \mathbf{K}_k\mathbf{G}_x\right)\mathbf{s}_{p,k|k-1} -\\ &\quad - \left(\mathbf{f}_{\theta,p} - \mathbf{K}_k\mathbf{g}_{\theta,p}\right). && (4.38)\end{aligned}$$

Since the sensitivity is a random variable, it can be described using the mean value

$$\hat{\mathbf{s}}_{p,k+1|k} = (\mathbf{F}_x - \mathbf{K}_k \mathbf{G}_x)\, \hat{\mathbf{s}}_{p,k|k-1} - \left(\mathbf{f}_{\theta,p} - \mathbf{K}_k \mathbf{g}_{\theta,p}\right), \tag{4.39}$$

and the second moment

$$\begin{aligned}
\mathbf{P}^{p,ss}_{k+1|k} &= \mathbb{E}\left[\mathbf{s}_{p,k+1|k}\, \mathbf{s}^T_{p,k+1|k}\right], && (4.40)\\
&= \left(\mathbf{F}_{x,\theta,p} - \mathbf{K}_k \mathbf{G}_{x,\theta,p}\right) \mathbf{P}^{xx}_{k|k-1} \left(\mathbf{F}_{x,\theta,p} - \mathbf{K}_k \mathbf{G}_{x,\theta,p}\right)^T \\
&\quad + (\mathbf{F}_x - \mathbf{K}_k \mathbf{G}_x)\, \mathbf{P}^{p,ss}_{k|k-1}\, (\mathbf{F}_x - \mathbf{K}_k \mathbf{G}_x)^T \\
&\quad + \left(\mathbf{F}_{x,\theta,p} - \mathbf{K}_k \mathbf{G}_{x,\theta,p}\right) \mathbf{P}^{p,xs}_{k|k-1} \left(\mathbf{F}_x - \mathbf{K}_k \mathbf{G}_x\right)^T \\
&\quad + (\mathbf{F}_x - \mathbf{K}_k \mathbf{G}_x)\, \mathbf{P}^{p,sx}_{k|k-1} \left(\mathbf{F}_{x,\theta,p} - \mathbf{K}_k \mathbf{G}_{x,\theta,p}\right)^T \\
&\quad - (\mathbf{F}_x - \mathbf{K}_k \mathbf{G}_x)\, \hat{\mathbf{s}}_{p,k|k-1} \left(\mathbf{f}_{\theta,p} - \mathbf{K}_k \mathbf{g}_{\theta,p}\right)^T \\
&\quad - \left(\mathbf{f}_{\theta,p} - \mathbf{K}_k \mathbf{g}_{\theta,p}\right) \hat{\mathbf{s}}^T_{p,k|k-1} \left(\mathbf{F}_x - \mathbf{K}_k \mathbf{G}_x\right)^T \\
&\quad + \left(\mathbf{f}_{\theta,p} - \mathbf{K}_k \mathbf{g}_{\theta,p}\right) \left(\mathbf{f}_{\theta,p} - \mathbf{K}_k \mathbf{g}_{\theta,p}\right)^T, && (4.41)
\end{aligned}$$

where the cross second moments between the state estimation error and the covariance are

$$\begin{aligned}
\mathbf{P}^{p,xs}_{k+1|k} &= \mathbb{E}\left[\mathbf{x}_{k+1|k}\, \mathbf{s}^T_{p,k+1|k}\right] && (4.42)\\
&= (\mathbf{F}_x - \mathbf{K}_k \mathbf{G}_x)\, \mathbf{P}^{xx}_{k|k-1} \left(\mathbf{F}_{x,\theta,p} - \mathbf{K}_k \mathbf{G}_{x,\theta,p}\right)^T \\
&\quad + (\mathbf{F}_x - \mathbf{K}_k \mathbf{G}_x)\, \mathbf{P}^{xs}_{k|k-1}\, (\mathbf{F}_x - \mathbf{K}_k \mathbf{G}_x)^T, && (4.43)\\
\mathbf{P}^{p,sx}_{k+1|k} &= \left(\mathbf{P}^{p,xs}_{k+1|k}\right)^T. && (4.44)
\end{aligned}$$

This concludes the description of the nonlinear system (4.20), its linearization (4.24) and the observer (4.30) with its sensitivity (4.38).

4.2.2 Suboptimal filter

The extended exact desensitized Kalman filter is derived using the optimization criterion from the XDKF. The optimization criterion weights the minimum mean square error used in the standard Kalman filter with the minimum trace of the sensitivity second moment

$$\begin{gathered}
\min_{\mathbf{K}_k} J(\mathbf{K}_k), \;\; J(\mathbf{K}_k) = \alpha \operatorname{tr} \mathbf{P}^{xx}_{k+1|k} + \sum_p \gamma_p \operatorname{tr} \mathbf{P}^{p,ss}_{k+1|k}, \\
\alpha \equiv 1 - \sum_p \gamma_p, \;\; 0 \leq \sum_p \gamma_p < 1, \;\; \text{and} \;\; \gamma_p \geq 0,
\end{gathered} \tag{4.45}$$

where γ_p are the weights that control the trade-off between the optimization objectives. The optimal gain $\mathbf{K}_k$ is obtained by solving

$$\frac{\partial J(\mathbf{K}_k)}{\partial \mathbf{K}_k} = 0. \tag{4.46}$$

Solving requires to evaluate

$$\begin{aligned}\frac{\partial \mathrm{tr}\mathbf{P}^{p,ss}_{k+1|k}}{\partial \mathbf{K}_k} &= -2\mathbf{F}_{x,\theta,p}\mathbf{P}^{xx}_{k|k-1}\mathbf{G}^T_{x,\theta,p} - 2\mathbf{F}_x\mathbf{P}^{p,ss}_{k|k-1}\mathbf{G}^T_x \\ &\quad - 2\mathbf{F}_{x,\theta,p}\mathbf{P}^{p,xs}_{k|k-1}\mathbf{G}^T_x - 2\mathbf{F}_x\mathbf{P}^{p,sx}_{k|k-1}\mathbf{G}^T_{x,\theta,p} \\ &\quad + 2\mathbf{F}_x\hat{\mathbf{s}}_{p,k|k-1}\mathbf{g}^T_{\theta,p} + 2\mathbf{f}_{\theta,p}\hat{\mathbf{s}}^T_{p,k|k-1}\mathbf{G}^T_x - 2\mathbf{f}_{\theta,p}\mathbf{g}^T_{\theta,p} \\ &\quad + 2\mathbf{K}_k\mathbf{G}_{x,\theta,p}\mathbf{P}^{xx}_{k|k-1}\mathbf{G}^T_{x,\theta,p} + 2\mathbf{K}_k\mathbf{G}_x\mathbf{P}^{p,ss}_{k|k-1}\mathbf{G}^T_x \\ &\quad + 2\mathbf{K}_k\mathbf{G}_{x,\theta,p}\mathbf{P}^{p,xs}_{k|k-1}\mathbf{G}^T_x + 2\mathbf{K}_k\mathbf{G}_x\mathbf{P}^{p,sx}_{k|k-1}\mathbf{G}^T_{x,\theta,p} \\ &\quad - 2\mathbf{K}_k\mathbf{G}_x\hat{\mathbf{s}}_{p,k|k-1}\mathbf{g}^T_{\theta,p} - 2\mathbf{K}_k\mathbf{g}_{\theta,p}\hat{\mathbf{s}}^T_{p,k|k-1}\mathbf{G}^T_x \\ &\quad + 2\mathbf{K}_k\mathbf{g}_{\theta,p}\mathbf{g}^T_{\theta,p}, \end{aligned} \tag{4.47}$$

and

$$\frac{\partial \mathrm{tr}\mathbf{P}^{xx}_{k+1|k}}{\partial \mathbf{K}_k} = -2\mathbf{F}_x\mathbf{P}^{xx}_{k|k-1}\mathbf{G}^T_x + 2\mathbf{K}_k\left(\mathbf{G}_x\mathbf{P}^{xx}_{k|k-1}\mathbf{G}^T_x + \mathbf{R}\right). \tag{4.48}$$

Then solving (4.46) gives the solution

$$\mathbf{K}_k = \mathbf{L}\mathbf{M}^{-1}, \tag{4.49}$$

where

$$\begin{aligned}\mathbf{L} &= \alpha\,\mathbf{F}_x\mathbf{P}^{xx}_{k|k-1}\mathbf{G}^T_x \\ &\quad + \sum_p \gamma_p\Big(\mathbf{F}_{x,\theta,p}\mathbf{P}^{xx}_{k|k-1}\mathbf{G}^T_{x,\theta,p} + \mathbf{F}_x\mathbf{P}^{p,ss}_{k|k-1}\mathbf{G}^T_x \\ &\quad + \mathbf{F}_{x,\theta,p}\mathbf{P}^{p,xs}_{k|k-1}\mathbf{G}^T_x + \mathbf{F}_x\mathbf{P}^{p,sx}_{k|k-1}\mathbf{G}^T_{x,\theta,p} \\ &\quad - \mathbf{F}_x\hat{\mathbf{s}}_{p,k|k-1}\mathbf{g}^T_{\theta,p} - \mathbf{f}_{\theta,p}\hat{\mathbf{s}}^T_{p,k|k-1}\mathbf{G}^T_x + \mathbf{f}_{\theta,p}\mathbf{g}^T_{\theta,p}\Big), \end{aligned} \tag{4.50}$$

$$\begin{aligned}\mathbf{M} &= \alpha\left(\mathbf{G}_x\mathbf{P}^{xx}_{k|k-1}\mathbf{G}^T_x + \mathbf{R}\right) \\ &\quad + \sum_p \gamma_p\Big(\mathbf{G}_{x,\theta,p}\mathbf{P}^{xx}_{k|k-1}\mathbf{G}^T_{x,\theta,p} + \mathbf{G}_x\mathbf{P}^{p,ss}_{k|k-1}\mathbf{G}^T_x \\ &\quad + \mathbf{G}_{x,\theta,p}\mathbf{P}^{p,xs}_{k|k-1}\mathbf{G}^T_x + \mathbf{G}_x\mathbf{P}^{p,sx}_{k|k-1}\mathbf{G}^T_{x,\theta,p} \\ &\quad - \mathbf{G}_x\hat{\mathbf{s}}_{p,k|k-1}\mathbf{g}^T_{\theta,p} - \mathbf{g}_{\theta,p}\hat{\mathbf{s}}^T_{p,k|k-1}\mathbf{G}^T_x + \mathbf{g}_{\theta,p}\mathbf{g}^T_{\theta,p}\Big). \end{aligned} \tag{4.51}$$

The solution $\mathbf{K}_k$ exists if $\mathbf{M}$ is a positive definite matrix. The EXDKF is summed up in Algorithm 4.3.

Algorithm 4.3 (Extended eXact DKF — EXDKF).

- Given weights γ_p, stable weights $\gamma_{p,\mathrm{ub}}$ and prior estimates $\hat{\mathbf{x}}_{k|k-1}$, $\mathbf{P}^{xx}_{k|k-1}$, $\hat{\mathbf{s}}_{p,k|k-1}$, $\mathbf{P}^{ss}_{p,k|k-1}$, $\mathbf{P}^{sx}_{p,k|k-1}$.
- Evaluate the Jacobians (4.25) and (4.36) .
- Obtain gain (4.49) .
- Update the covariance (4.32) and the second moments (4.41), (4.44), (4.43) .
- Update state estimation mean (4.29) .
- Update sensitivity mean (4.39) .

4.2.3 Normalized objectives

The traces of second moments $\mathbf{P}^{xx}_{k+1|k}, \mathbf{P}^{ss}_{p,k+1|k}$ are not normalized in general. Therefore, the weight values must also capture the scaling of the objectives. This scaling would also need to be adaptive since the trace of second moments is time-variant. The adaptive normalization method for the XDKF was proposed to overcome this issue in Section 3.2.6. The same method can be applied to the EXDKF, which leads to the EXDKF with normalized objectives (EXDKF-N) summarized in Algorithm 4.4.

Algorithm 4.4 (Extended eXact DKF with Normalized objectives — EXDKF-N).

- Given weights γ_p, stable weights $\bar{\gamma}_p$ and prior estimates $\hat{\mathbf{x}}_{k|k-1}$, $\mathbf{P}^{xx}_{k|k-1}$, $\hat{\mathbf{s}}_{p,k|k-1}$, $\mathbf{P}^{ss}_{p,k|k-1}$, $\mathbf{P}^{sx}_{p,k|k-1}$.
- Evaluate the Jacobians (4.25) .
- Obtain $\mathbf{K}^{\star}_{\mathrm{KF}}$ by temporary setting α to zero and and running EXDKF update step in Algorithm 4.3 with the prior estimates .
- Obtain $\mathbf{K}^{\star}_{\mathrm{S},p}$ by temporary setting γ_p to $\bar{\gamma}_p$ and and running EXDKF update step in Algorithm 4.3 with the prior estimates.
- Evaluate utopia (3.115) and nadir (3.116) points for the $\mathbf{K}^{\star}_{\mathrm{KF}}, \mathbf{K}^{\star}_{\mathrm{S},p}$ obtained in the previous steps.
- Obtain posterior estimates $\hat{\mathbf{x}}_{k+1|k}$, $\mathbf{P}^{xx}_{k+1|k}$, $\hat{\mathbf{s}}_{p,k+1|k}$, $\mathbf{P}^{ss}_{p,k+1|k}$, $\mathbf{P}^{sx}_{p,k+1|k}$ by temporary setting α, γ_p to $\nu_\alpha, \nu_{\gamma,p}$ (3.119) respectively, and running EXDKF update step in Algorithm 4.3 with the prior estimates .

4.2.4 Numerical example

The performance of the EXDKF and the EXDKF-N is tested on the nonlinear system based on the example in Section 3.2.7. The performance is compared to the standard extended KF (EKF) and the desensitized extended Kalman filter (EDKF) from [17].

The nonlinear discrete-time system is defined as

$$\begin{aligned} x_{1,k+1} &= 0.9802 x_{1,k} + (0.0196 + 0.099\theta)\, x_{2,k}^2 + w_{1,k}, \\ x_{2,k+1} &= 0.9802 x_{2,k}^2 + w_{2,k}, \\ y &= x_{1,k} - x_{2,k} + e_k, \end{aligned} \tag{4.52}$$

where the noise covariances are

$$\mathbf{Q} = \begin{bmatrix} 1.9608 & 0.0195 \\ 0.0195 & 1.9605 \end{bmatrix}, \mathbf{R} = 1, \tag{4.53}$$

and the value of the uncertain parameter θ is expected to be within the interval $[-1, 1]$. The real parameter value is fixed during each simulation.

The test runs 500 Monte Carlo simulations for each of the nine true parameter values from the valid interval. Each simulation is 200 samples long, and it is initiated with the initial state value

$$x_{1,0} = x_{2,0} = 1. \tag{4.54}$$

Two versions of the EKF are used for the comparison. One version is denoted as the perfect EKF and contains information about the true parameter value. The other is denoted as the imperfect EKF, and it assumes that the parameter is zero. The

same parameter expected value $\hat{\theta} = 0$ is used by the desensitized algorithms – EDKF, EXDKF, EXDKF-N. All filters are initiated with the state estimate

$$\hat{x}_{1,0|-1} = \hat{x}_{2,0|-1} = 0,\ \mathbf{P}_{xx,0|-1} = \mathbf{I}, \tag{4.55}$$

and sensitivity

$$\hat{s}_{1,0|-1} = \mathbf{0},\ \mathbf{P}_{ss,0|-1} = \mathbf{I},\ \mathbf{P}_{sx,0|-1} = \mathbf{0}. \tag{4.56}$$

The EXDKF and the EXDKF-N are used with weight $\gamma = 0.5$, and the EXDKF-N uses $\gamma_{\text{ub}} = 0.99$ for objective normalization. The EDKF is used with the weighting matrix $\mathbf{W} = \mathbf{I}$, which corresponds to the EXDKF weighting. The corresponding EDKF weights are obtained by the diagonal matrix $W = \text{diag}([w, w])$, with values $w = \frac{\gamma}{\alpha}$ where γ, α are the XDKF weights.

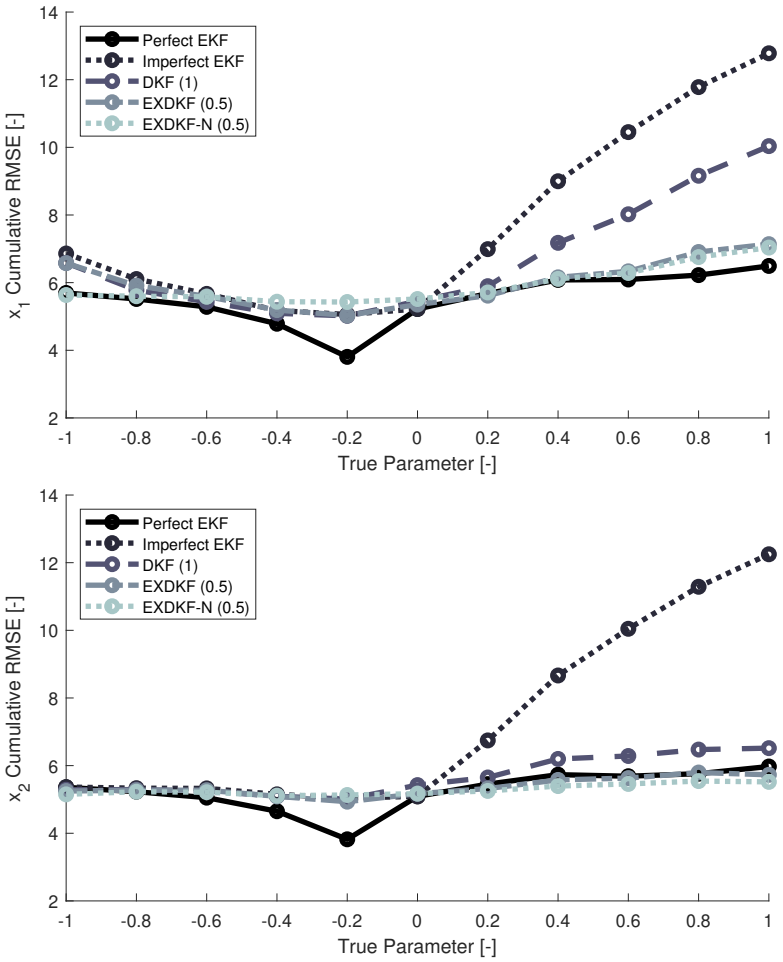

Figure 4.1. Cumulative RMSE comparison. Separate figures show the final cumulative RMSE for the states x_1, x_2 after 200 samples. The mean cumulative RMSE from 500 Monte Carlo simulations for each real parameter value is shown. The value in brackets represents the weight value.

The test results are shown in Fig. 4.1. The comparison is made using the root-mean-square error (RMSE) between true and estimated states. The results show that the imperfect EKF has significantly worse accuracy than the others. The desensitized filters prove their robustness to the uncertain parameter. Their accuracy is similar for all tested true parameter values. The accuracy of the desensitized filters is sometimes even better than the perfect EKF because the EKF is not the optimal filter for nonlinear systems due to the linearization error. Among the desensitized filters, the proposed filters EXDKF and EXDKF are more accurate than the original EDKF. Furthermore, the normalized objectives in the EXDKF-N improve the estimation accuracy.

Chapter 5
Desensitized fault detection and diagnosis

This chapter focuses on using desensitized filters for fault detection applications. Most of the results in this chapter were originally published in [O3, J2].

The Kalman filter can also be used for fault detection purposes by analyzing the statistics of a prediction error sequence. The prediction error sequence is also called the innovation sequence. In the nominal case, i.e., when no fault occurs in the system, the innovation sequence should have white noise properties. Most model-based fault detection methods rely on this property, so they make decisions based on various tests on the innovation sequence. One of these methods is the interacting multiple-model (IMM) method. The model in the IMM is described as a Markovian jump linear system (MJLS). The idea behind the MJLS is that a system behavior changes due to external conditions, but a linear model can describe it at each time. System behavior under fixed conditions is called a system mode. The system can be modeled by a set of linear models which describe all system modes. Mode changes are modeled as a discrete-time Markov chain.

Let us assume the discrete-time Markovian jump linear system

$$\begin{aligned} \mathbf{x}_{k+1}^{(m)} &= \mathbf{f}^{(m)}\left(\mathbf{x}_k^{(m)}, \mathbf{u}_k, \boldsymbol{\theta}\right) + \mathbf{v}_k^{(m)} = \mathbf{A}^{(m)}(\boldsymbol{\theta})\mathbf{x}_k^{(m)} + \mathbf{B}^{(m)}(\boldsymbol{\theta})\mathbf{u}_k + \mathbf{v}_k^{(m)}, \\ \mathbf{y}_k &= \mathbf{g}^{(m)}\left(\mathbf{x}_k^{(m)}, \boldsymbol{\theta}\right) + \mathbf{e}_k^{(m)} = \mathbf{C}^{(m)}\mathbf{x}_k^{(m)} + \mathbf{e}_k^{(m)}, \end{aligned} \tag{5.1}$$

where m in the superscript brackets denotes the mode number from the predefined set $\mathcal{M} = \{1, 2, \ldots, N_{\mathrm{m}}\}$ which is active at time k. The variables of the system at each mode have properties of the linear system in (3.39), namely $\mathbf{x}_k$ is the state vector of dimension N_x, $\mathbf{u}_k$ is the input vector of dimension N_u, $\mathbf{y}_k$ is the measurement vector of dimension N_y, $\boldsymbol{\theta}$ is the parameter vector of dimension N_θ, $\mathbf{A}^{(m)}(\boldsymbol{\theta})$ is $N_x \times N_x$ matrix, $\mathbf{B}^{(m)}(\boldsymbol{\theta})$ is $N_x \times N_u$ matrix and $\mathbf{C}^{(m)}$ is $N_y \times N_x$ matrix. The matrices $\mathbf{A}^{(m)}$, $\mathbf{B}^{(m)}$ are assumed to be linear matrix functions of parameters

$$\mathbf{A}^{(m)}(\boldsymbol{\theta}) = \mathbf{A}_0^{(m)} + \theta_1\mathbf{A}_1^{(m)} + \ldots + \theta_{N_\theta}\mathbf{A}_{N_\theta}^{(m)}, \tag{5.2}$$

$$\mathbf{B}^{(m)}(\boldsymbol{\theta}) = \mathbf{B}_0^{(m)} + \theta_1\mathbf{B}_1^{(m)} + \ldots + \theta_{N_\theta}\mathbf{B}_{N_\theta}^{(m)}. \tag{5.3}$$

The noise statistics of the system is

$$\begin{bmatrix} \mathbf{v}_k^{(m)} \\ \mathbf{e}_k^{(m)} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{Q}^{(m)} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}^{(m)} \end{bmatrix}\right). \tag{5.4}$$

The Markovian model mode transition probabilities

$$T(l, m) = P(M_{k+1} = m \,|M_k = l\,), \tag{5.5}$$

from the transition probability matrix $\mathbf{T}$ define the probability of transition from mode $l \in \mathcal{M}$ to mode $m \in \mathcal{M}$. Notation $T(l, m)$ denotes the element of $\mathbf{T}$ at l-th row and m-th column.

The IMM offers a suboptimal solution to state estimation of discrete-time Markovian jump linear systems (MJLS). Each mode of the MJLS requires a separate implementation of the Kalman filter. Together, they create a bank of Kalman filters. The algorithm is depicted in Fig. 5.1. The posterior state estimates from the last step $\hat{\mathbf{x}}^{(m)}_{k-1|k-1}$ are mixed by Markov mode transition probability $\mathbf{T}$ and mode probability $\boldsymbol{\mu}_{k-1}$. Then the filter bank evaluates the posterior state estimates $\hat{\mathbf{x}}^{(m)}_{k|k}$ and the likelihoods of system modes $\Lambda^{(m)}_k$. After that, the posterior state estimates are mixed by the posterior mode probability $\boldsymbol{\mu}_k$, which results in a single combined state estimate $\hat{\mathbf{x}}_{k|k}$.

The purpose of the output combination step is to give a single equivalent mixture distribution that represents the state system state. However, in some applications, a state of a single mode might be required. Then the combination step can be skipped.

If the dimensions of mode state vectors differ, the dimensions need to be unified before mixing and combination steps. Solutions to this problem are discussed in [66]. An unbiased solution replaces all missing estimates at the given mode with the best available estimate, e.g., from the last combination step. Other methods might replace the missing estimates with some default values. Note that missing estimates of state, sensitivity, and second moments must be provided. The choice of the method depends on the particular application.

The IMM can be easily used for fault detection and diagnosis by defining the MJLS as a model which consists of a mode for nominal system behavior and modes for system behavior when faults occur. The IMM applied to such a model simultaneously estimates the state and mode probability of the system. The mode probability gives information about the system's most probable mode, which indicates (detects) the probability of fault occurrence.

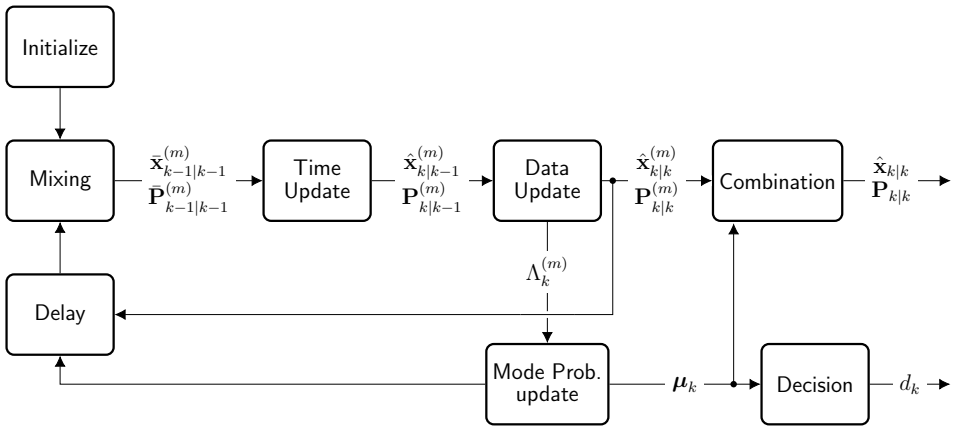

Figure 5.1. Interacting multiple-model method.

As in the desensitized filters, the true parameter vector $\boldsymbol{\theta}$ is unknown, so the estimated parameter vector $\hat{\boldsymbol{\theta}}$ is used in the filters. Also, the following compact notation is used for matrices $\mathbf{A}^{(m)}, \mathbf{B}^{(m)}$

$$\hat{\mathbf{A}}^{(m)} \equiv \mathbf{A}^{(m)}\left(\hat{\boldsymbol{\theta}}\right), \ \hat{\mathbf{B}}^{(m)} \equiv \mathbf{B}^{(m)}\left(\hat{\boldsymbol{\theta}}\right). \tag{5.6}$$

5.1 IMM with desensitized Kalman filtering

The IMM algorithm for the system in (5.1), which reduces sensitivity to parameters, can be created by modifying the standard IMM. First, the standard Kalman filters in

the filtering step are replaced by the desensitized filters. In this section, the SDKF is chosen. The IMM with the XDKF will be introduced in the next section. The filters in the IMM must have separate update steps due to mixing updates between the data-update step and the time-update step. That is why the SDKF is used as it was originally derived in [21]. To complete the IMM modification for the SDKF usage, the interaction step needs to be extended by mixing the state estimation error sensitivities to the parameter vector which are used in the filtering step. The IMM algorithm with SDKF is defined as follows.

Algorithm 5.1 (IMM with SDKF — IMM-SDKF).

- **Initialize.** The initial values need to be determined for the initial mode probability $\boldsymbol{\mu}_0$, initial estimate of state mean $\hat{\mathbf{x}}_{0|0}^{(m)}$, state estimation error covariance $\mathbf{P}_{0|0}^{(m)}$ and initial state estimation error sensitivity to parameters $\boldsymbol{\Sigma}_{0|0}^{(m)}$.
- **Interaction/Mixing.** The standard IMM step equations are applied together with mixing of sensitivities

$$\pi_{k-1}\left(m\,|l\right) = \frac{T(l,m)\,\mu_{k-1}\left(l\right)}{\sum_{l\in\mathcal{M}} T(l,m)\,\mu_{k-1}\left(l\right)}, \tag{5.7}$$

$$\bar{\mathbf{x}}_{k-1|k-1}^{(m)} = \sum_{l\in\mathcal{M}} \hat{\mathbf{x}}_{k-1|k-1}^{(l)}\,\pi_{k-1}\left(m\,|l\right), \tag{5.8}$$

$$\begin{aligned}\bar{\mathbf{P}}_{k-1|k-1}^{(m)} = \sum_{l\in\mathcal{M}} \pi_{k-1}\left(m\,|l\right)\Big[&\mathbf{P}_{k-1|k-1}^{(l)} + \\ &+ \left(\hat{\mathbf{x}}_{k-1|k-1}^{(l)} - \bar{\mathbf{x}}_{k-1|k-1}^{(m)}\right)\left(\hat{\mathbf{x}}_{k-1|k-1}^{(l)} - \bar{\mathbf{x}}_{k-1|k-1}^{(m)}\right)^T\Big],\end{aligned} \tag{5.9}$$

$$\bar{\boldsymbol{\Sigma}}_{k-1|k-1}^{(m)} = \sum_{l\in\mathcal{M}} \boldsymbol{\Sigma}_{k-1|k-1}^{(l)}\,\pi_{k-1}\left(m\,|l\right), \tag{5.10}$$

  where $\mu_{k-1}\left(m\right)$ denotes the m-th element in the mode probability vector μ_{k-1} at time $k-1$, and $\pi_{k-1}\left(m\,|l\right)$ is the mixing probability that denotes the transition from mode l to mode m. Linear combination of sensitivities in (5.10) comes from the definition of sensitivity in (3.24) where it is defined as the sensitivity of the state mean value to the parameter vector. This sensitivity definition can be applied to (5.8) and obtain $\bar{\boldsymbol{\Sigma}}_{k-1|k-1}^{(m)} = \partial\bar{\mathbf{x}}_{k-1|k-1}^{(m)}/\partial\hat{\boldsymbol{\theta}}$, which results in (5.10).
- **Mode state estimation.** The state estimation is done using separate SDKF for each mode. The updates are defined as follows
  - Time-update step

$$\hat{\mathbf{x}}_{k|k-1}^{(m)} = \hat{\mathbf{A}}^{(m)}\bar{\mathbf{x}}_{k-1|k-1}^{(m)} + \hat{\mathbf{B}}^{(m)}\mathbf{u}_{k-1}, \tag{5.11}$$

$$\boldsymbol{\Sigma}_{k|k-1}^{(m)} = \hat{\mathbf{A}}^{(m)}\bar{\boldsymbol{\Sigma}}_{k-1|k-1}^{(m)} + \left.\frac{\partial\mathbf{f}^{(m)}}{\partial\boldsymbol{\theta}}\right|_{\bar{\mathbf{x}}_{k-1|k-1}^{(m)},\mathbf{u}_{k-1},\hat{\boldsymbol{\theta}}}, \tag{5.12}$$

$$\mathbf{P}_{k|k-1}^{(m)} = \hat{\mathbf{A}}^{(m)}\bar{\mathbf{P}}_{k-1|k-1}^{(m)}\left(\hat{\mathbf{A}}^{(m)}\right)^T + \mathbf{Q}. \tag{5.13}$$

  - Data-update step

$$\boldsymbol{\Gamma}_k^{(m)} = \mathbf{C}^{(m)}\boldsymbol{\Sigma}_{k|k-1}^{(m)}, \tag{5.14}$$

$$\hat{\mathbf{y}}_{k|k-1}^{(m)} = \mathbf{C}^{(m)}\hat{\mathbf{x}}_{k|k-1}^{(m)}, \tag{5.15}$$

$$\mathbf{S}_{k|k-1}^{(m)} = \mathbf{C}^{(m)}\mathbf{P}_{k|k-1}^{(m)}\mathbf{C}^{(m)^T} + \mathbf{R} + \mathbf{\Gamma}_k^{(m)}\mathbf{W}^{(m)}\left(\mathbf{\Gamma}_k^{(m)}\right)^T, \tag{5.16}$$

$$\mathbf{K}_k^{(m)} = \left(\mathbf{P}_{k|k-1}^{(m)}\mathbf{C}^{(m)^T} + \mathbf{\Sigma}_{k|k-1}^{(m)}\mathbf{W}^{(m)}\left(\mathbf{\Gamma}_k^{(m)}\right)^T\right)\left(\mathbf{S}_{k|k-1}^{(m)}\right)^{-1}, \tag{5.17}$$

$$\hat{\mathbf{x}}_{k|k}^{(m)} = \hat{\mathbf{x}}_{k|k-1}^{(m)} + \mathbf{K}_k^{(m)}\left(\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}^{(m)}\right), \tag{5.18}$$

$$\mathbf{\Sigma}_{k|k}^{(m)} = \mathbf{\Sigma}_{k|k-1}^{(m)} + \mathbf{K}_k^{(m)}\mathbf{\Gamma}_k^{(m)}, \tag{5.19}$$

$$\mathbf{P}_{k|k}^{(m)} = \left(\mathbf{I} - \mathbf{K}_k^{(m)}\mathbf{C}^{(m)}\right)\mathbf{P}_{k|k-1}^{(m)}\left(\mathbf{I} - \mathbf{K}_k^{(m)}\mathbf{C}^{(m)}\right)^T + \mathbf{K}_k^{(m)}\mathbf{R}\left(\mathbf{K}_k^{(m)}\right)^T. \tag{5.20}$$

- **Probability update.** The probabilities of modes are updated using the standard IMM mode probability update

$$\Lambda_k^{(m)} = p\left(\mathbf{y}_k \middle| \hat{\mathbf{x}}_{k|k-1}^{(m)}, \mathbf{P}_{k|k-1}^{(m)}, \mathbf{\Sigma}_{k|k-1}^{(m)}\right),$$
$$\text{where } p\left(\mathbf{y}_k \middle| \hat{\mathbf{x}}_{k|k-1}^{(m)}, \mathbf{P}_{k|k-1}^{(m)}, \mathbf{\Sigma}_{k|k-1}^{(m)}\right) \sim \mathcal{N}\left(\hat{\mathbf{y}}_{k|k-1}^{(m)}, \mathbf{S}_{k|k-1}^{(m)}\right), \tag{5.21}$$

$$\mu_k(m) = \frac{1}{q}\Lambda_k^{(m)}\sum_{l\in\mathcal{M}} T(l,m)\,\mu_{k-1}(l), \tag{5.22}$$

  where q is the normalizing factor.
- **Output combination.** The standard IMM output combination step is extended with the combination of sensitivities derived using the approach explained in the interaction/mixing step.

$$\hat{\mathbf{x}}_{k|k} = \sum_{m\in\mathcal{M}} \hat{\mathbf{x}}_{k|k}^{(m)}\mu_k(m), \tag{5.23}$$

$$\mathbf{P}_{k|k} = \sum_{m\in\mathcal{M}} \mu_k(m)\left[\mathbf{P}_{k|k}^{(m)} + \left(\hat{\mathbf{x}}_{k|k}^{(m)} - \hat{\mathbf{x}}_{k|k}\right)\left(\hat{\mathbf{x}}_{k|k}^{(m)} - \hat{\mathbf{x}}_{k|k}\right)^T\right], \tag{5.24}$$

$$\mathbf{\Sigma}_{k|k} = \sum_{m\in\mathcal{M}} \mathbf{\Sigma}_{k|k}^{(m)}\mu_k(m). \tag{5.25}$$

To summarize, the IMM-SDKF is the IMM algorithm where the state estimation error sensitivity to parameters is considered. This modification can improve state estimation accuracy when uncertain parameters are present in the model. The IMM-SDKF algorithm maintains the efficiency of the standard IMM.

5.2 IMM with exact desensitized Kalman filtering

The IMM algorithm for the system in (5.1) can also be modified by the exact desensitizing filter. Since the IMM requires a two-step filtering algorithm, the XDKF-Z with separated steps in Algorithm 3.8 is the most suitable. The modification of the IMM starts with replacing the standard Kalman filters in the filtering step are replaced by the XDKF-Z. The replacement is simple, but the user must be careful with the time indices. The reason is that the order of the steps in Algorithm 3.8 must be flipped, so the algorithm starts with a time-update step. Consequently, the decorrelation matrices in the time-update step must be computed at the previous step, meaning they must be stored in memory together with the state, sensitivity, and covariance. An alternative solution is to reorder IMM algorithm steps which will be done in Chapter 7. Otherwise, the definition of the IMM algorithm with XDKF-Z (IMM-XDKF-Z) is straightforward. The IMM-XDKF-Z is summed up in Algorithm 5.2.

Algorithm 5.2 (IMM with XDKF-Z — IMM-XDKF-Z).

- **Initialize.** The initial values need to be determined for the initial mode probability $\boldsymbol{\mu}_0$, the initial estimate of state mean $\hat{\mathbf{x}}_{0|0}^{(m)}$, the state estimation error covariance $\mathbf{P}_{0|0}^{(m)}$ and the initial state estimation error sensitivity to parameters $\hat{\mathbf{s}}_{p,0|0}^{(m)}$.
- **Interaction/Mixing.** The standard IMM step equations are applied together with a mixing of sensitivities

$$\pi_{k-1}(m\,|l) = \frac{T(l,m)\,\mu_{k-1}(l)}{\sum_{l\in\mathcal{M}} T(l,m)\,\mu_{k-1}(l)}, \tag{5.26}$$

$$\bar{\mathbf{x}}_{k-1|k-1}^{(m)} = \sum_{l\in\mathcal{M}} \hat{\mathbf{x}}_{k-1|k-1}^{(l)}\,\pi_{k-1}(m\,|l)\,, \tag{5.27}$$

$$\begin{aligned}\bar{\mathbf{P}}_{k-1|k-1}^{\Sigma,(m)} = \sum_{l\in\mathcal{M}} \pi_{k-1}(m\,|l)\,\Big[&\mathbf{P}_{k-1|k-1}^{\Sigma,(l)} + \\ &+ \left(\hat{\mathbf{x}}_{k-1|k-1}^{(l)} - \bar{\mathbf{x}}_{k-1|k-1}^{(m)}\right)\left(\hat{\mathbf{x}}_{k-1|k-1}^{(l)} - \bar{\mathbf{x}}_{k-1|k-1}^{(m)}\right)^T\Big],\end{aligned} \tag{5.28}$$

$$\bar{\mathbf{s}}_{p,k-1|k-1}^{(m)} = \sum_{l\in\mathcal{M}} \hat{\mathbf{s}}_{p,k-1|k-1}^{(l)}\,\pi_{k-1}(m\,|l)\,, \tag{5.29}$$

where $\mu_{k-1}(m)$ denotes the m-th element in the mode probability vector μ_{k-1} at time $k-1$, and $\pi_{k-1}(m\,|l)$ denotes the element of the matrix with mixed transition probabilities $\boldsymbol{\pi}_{k-1}$. where it is defined as the sensitivity of the state mean value to the parameter vector.

- **Mode state estimation.** The filtering update is defined as follows
  - Time-update step

$$\hat{\mathbf{A}}_{\mathrm{dc}}^{(m)} = \hat{\mathbf{A}}^{(m)} - \mathbf{S}_{k-1}^{(m)}\left(\alpha^{(m)}\mathbf{R}^{(m)}\right)^{-1}\mathbf{C}^{(m)}, \tag{5.30}$$

$$\mathbf{Q}_{\mathrm{dc}}^{(m)} = \mathbf{Q}_{k-1}^{\Sigma,(m)} - \mathbf{S}_{k-1}^{(m)}\left(\alpha^{(m)}\mathbf{R}^{(m)}\right)^{-1}\left(\mathbf{S}_{k-1}^{(m)}\right)^T, \tag{5.31}$$

$$\hat{\mathbf{x}}_{k|k-1}^{(m)} = \hat{\mathbf{A}}_{\mathrm{dc}}^{(m)}\bar{\mathbf{x}}_{k-1|k-1}^{(m)} + \hat{\mathbf{B}}^{(m)}\mathbf{u}_{k-1} + \mathbf{S}_{k-1}^{(m)}\left(\alpha^{(m)}\mathbf{R}^{(m)}\right)^{-1}\mathbf{y}_{k-1}, \tag{5.32}$$

$$\hat{\mathbf{s}}_{p,k|k-1}^{(m)} = \hat{\mathbf{A}}_{\mathrm{dc}}^{(m)}\bar{\mathbf{s}}_{p,k-1|k-1}^{(m)} - \mathbf{b}_{p,k-1}^{(m)}, \tag{5.33}$$

$$\mathbf{P}_{k|k-1}^{\Sigma,(m)} = \hat{\mathbf{A}}_{\mathrm{dc}}^{(m)}\bar{\mathbf{P}}_{k-1|k-1}^{\Sigma,(m)}\left(\hat{\mathbf{A}}_{\mathrm{dc}}^{(m)}\right)^T + \mathbf{Q}_{\mathrm{dc}}^{(m)}. \tag{5.34}$$

  - Uncertainty-update step

$$\mathbf{b}_{p,k}^{(m)} = \mathbf{A}_p^{(m)}\hat{\mathbf{x}}_{k|k-1}^{(m)} + \mathbf{B}_p^{(m)}\mathbf{u}_k, \tag{5.35}$$

$$\mathbf{S}_k^{(m)} = -\sum_p \gamma_p^{(m)}\mathbf{b}_{p,k}^{(m)}\left(\mathbf{C}^{(m)}\hat{\mathbf{s}}_{p,k|k-1}^{(m)}\right)^T, \tag{5.36}$$

$$\begin{aligned}\mathbf{Q}_k^{\Sigma,(m)} = \alpha\mathbf{Q}^{(m)} + \sum_p \gamma_p^{(m)}\Big[&\mathbf{b}_{p,k}^{(m)}\left(\mathbf{b}_{p,k}^{(m)}\right)^T - \\ &- \mathbf{b}_{p,k}^{(m)}\left(\hat{\mathbf{A}}^{(m)}\hat{\mathbf{s}}_{p,k|k-1}^{(m)}\right)^T - \hat{\mathbf{A}}^{(m)}\hat{\mathbf{s}}_{p,k|k-1}^{(m)}\left(\mathbf{b}_{p,k}^{(m)}\right)^T\Big].\end{aligned} \tag{5.37}$$

  - Data-update step

$$\mathbf{K}_k^{(m)} = \mathbf{P}_{k|k-1}^{\Sigma,(m)}\left(\mathbf{C}^{(m)}\right)^T\left(\mathbf{C}^{(m)}\mathbf{P}_{k|k-1}^{\Sigma,(m)}\left(\mathbf{C}^{(m)}\right)^T + \alpha^{(m)}\mathbf{R}^{(m)}\right)^{-1}, \tag{5.38}$$

$$\hat{\mathbf{x}}_{k|k}^{(m)} = \hat{\mathbf{x}}_{k|k-1}^{(m)} + \mathbf{K}_k^{(m)} \left(\mathbf{y}_k - \mathbf{C}^{(m)} \hat{\mathbf{x}}_{k|k-1}^{(m)} \right), \tag{5.39}$$

$$\hat{\mathbf{s}}_{p,k|k}^{(m)} = \hat{\mathbf{s}}_{p,k|k-1}^{(m)} - \mathbf{K}_k^{(m)} \mathbf{C}^{(m)} \hat{\mathbf{s}}_{p,k|k-1}^{(m)}, \tag{5.40}$$

$$\mathbf{P}_{k|k}^{\Sigma,(m)} = \mathbf{P}_{k|k-1}^{\Sigma,(m)} - \mathbf{K}_k^{(m)} \mathbf{C}^{(m)} \mathbf{P}_{k|k-1}^{\Sigma,(m)}. \tag{5.41}$$

- **Probability update.** Before the probability update, the state error covariance needs to be extracted from the cumulative second moment by rewriting (3.82) into

$$\mathbf{P}_{k|k-1}^{(m)} = \frac{1}{\alpha^{(m)}} \left(\mathbf{P}_{k|k-1}^{\Sigma,(m)} - \sum_{p=1}^{N_\theta} \gamma_p^{(m)} \hat{\mathbf{s}}_{p,k|k-1}^{(m)} \left(\hat{\mathbf{s}}_{p,k|k-1}^{(m)} \right)^T \right). \tag{5.42}$$

Then the output error covariance $\mathbf{P}_{k|k-1}^{yy,(m)}$ is computed, and the probabilities of modes $\boldsymbol{\mu}_k$ are updated using the standard IMM mode probability update

$$\mathbf{P}_{k|k-1}^{yy,(m)} = \mathbf{C}^{(m)} \mathbf{P}_{k|k-1}^{(m)} \left(\mathbf{C}^{(m)} \right)^T + \mathbf{R}^{(m)}, \tag{5.43}$$

$$\Lambda_k^{(m)} = p\left(\mathbf{y}_k \left| \hat{\mathbf{x}}_{k|k-1}^{(m)}, \mathbf{P}_{k|k-1}^{(m)} \right. \right),$$

$$\text{where } p\left(\mathbf{y}_k \left| \hat{\mathbf{x}}_{k|k-1}^{(m)}, \mathbf{P}_{k|k-1}^{(m)} \right. \right) \sim \mathcal{N}\left(\hat{\mathbf{y}}_{k|k-1}^{(m)}, \mathbf{P}_{k|k-1}^{yy,(m)} \right), \tag{5.44}$$

$$\mu_k(m) = \frac{1}{q} \Lambda_k^{(m)} \sum_{l \in \mathcal{M}} T(l,m)\, \mu_{k-1}(l),$$

where q is the normalizing factor.

- **Output combination.** The standard IMM output combination step is extended with the combination of sensitivities derived using the approach explained in the interaction/mixing step.

$$\hat{\mathbf{x}}_{k|k} = \sum_{m \in \mathcal{M}} \hat{\mathbf{x}}_{k|k}^{(m)} \mu_k(m), \tag{5.45}$$

$$\mathbf{P}_{k|k}^{\Sigma} = \sum_{m \in \mathcal{M}} \mu_k(m) \left[\mathbf{P}_{k|k}^{\Sigma,(m)} + \left(\hat{\mathbf{x}}_{k|k}^{(m)} - \hat{\mathbf{x}}_{k|k} \right) \left(\hat{\mathbf{x}}_{k|k}^{(m)} - \hat{\mathbf{x}}_{k|k} \right)^T \right], \tag{5.46}$$

$$\hat{\mathbf{s}}_{p,k|k} = \sum_{m \in \mathcal{M}} \hat{\mathbf{s}}_{p,k|k}^{(m)} \mu_k(m).$$

To sum up, the IMM-XDKF is the IMM algorithm where the state estimation error sensitivity to parameters is considered in a stochastic sense.

5.3 Application to buildings

The FDD using the IMM can be done by simply analyzing posterior mode probabilities. The fault modeled in mode m is detected when its mode probability $\mu_k(m)$ is the highest among all modes. In practical applications, various thresholds can be set to minimize false positive or false negative detections.

This section focuses on detecting faults in buildings that could cause the inefficiency of a control algorithm. The inefficiencies of the control algorithm could be caused by inaccurate state estimation or abrupt changes in a zone. The inefficiencies can be autonomously removed, and energy consumption can be decreased when accurate state estimation and fault detection are provided to a control algorithm.

A simple building model is used for testing the performance. The building model captures the thermal dynamics of the building using the RC (Resistance-Capacitance) equivalent method, which is based on the first-principles approach. The building is depicted in Fig. 5.2, consisting of three rooms connected in a row. Each room has a window and a controlled heating source. The inner wall length is the same in all rooms, but the outer walls differ. The rooms are modeled as first-order systems, and each wall is modeled as a third-order system. The adjacent walls with the same zones are merged into one wall model, resulting in three models for walls in each room and two for walls between rooms. Then the complete model consists of three room models and five wall models.

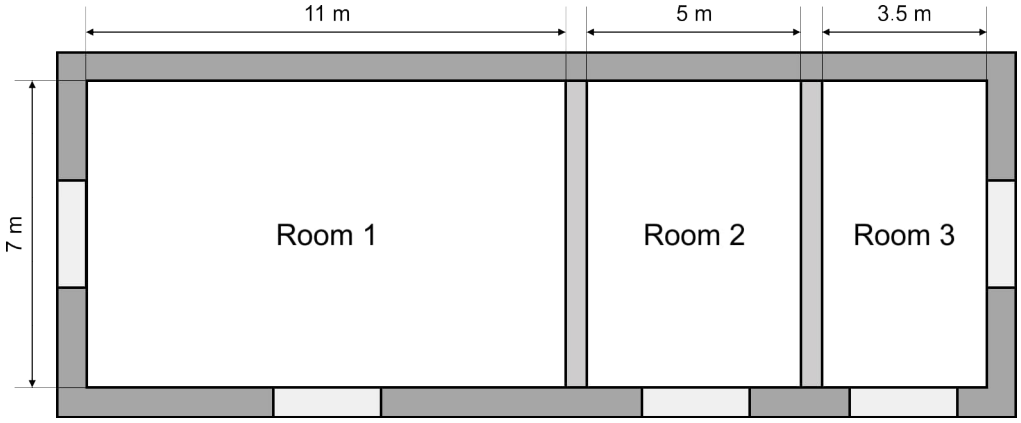

Figure 5.2. Plan of the building model used in simulations.

The entire building model is a linear system of the 18th order, with the state vector defined as

$$\mathbf{x} = [T_{\mathrm{R1}}, T_{\mathrm{R2}}, T_{\mathrm{R3}}, T_{\mathrm{W1i}}, \ldots, T_{\mathrm{W5i}}, T_{\mathrm{W1e}}, \ldots, T_{\mathrm{W5e}}, T_{\mathrm{W1c}}, \ldots, T_{\mathrm{W5c}}]^T, \tag{5.47}$$

where $T_{\mathrm{R}r}$ represents the temperature in room r. Furthermore, $T_{\mathrm{W}wi}, T_{\mathrm{W}we}, T_{\mathrm{W}wc}$ are the w-th wall temperatures of interior/exterior surfaces and core respectively. The inputs to the system include the outside (ambient) air temperature T_{amb} and the heat flow input q_{in} to each room which is controlled using a set of feedback PI controllers with limited resource. The input vector is defined as

$$\mathbf{u} = [q_{\mathrm{in1}}, q_{\mathrm{in2}}, q_{\mathrm{in3}}, T_{\mathrm{amb}}]^T, \tag{5.48}$$

where $q_{\mathrm{in1}}, q_{\mathrm{in2}}, q_{\mathrm{in3}}$ are the manipulated variables and T_{amb} is the disturbance variable. The outputs of the system are the temperature measurements in rooms

$$\mathbf{y} = [T_{\mathrm{R1}}, T_{\mathrm{R2}}, T_{\mathrm{R3}}]^T. \tag{5.49}$$

The state space system matrices are not given in this text since they are too large, and this work is not focused on creating a building model.

The testing is designed to compare the standard IMM to the IMM-SDKF and the IMM-XDKF-Z using the "average zone" models. The average zone model means that each room (zone) is modeled by the same [1] average room model. The goal is to show that

[1] In practice, several average zone models would need to be created to distinguish between the dynamics of distinct zones. Usually, the number of average zone models needed to create a building model is low compared to the number of all zones in a building.

the faults can be detected even without complete information about room parameters, which would improve the scalability of the FDD system. Additionally, the standard IMM with the building model with full information is used as the benchmark for the best achievable performance. The building model with full information is also used for generating the measurement data.

Two types of faults are tested. The faults are modeled as model disturbances which extend the original building model. A separate augmented building model is created for each fault. These models are then used together with the nominal model as the system modes in the IMM algorithm. The first disturbance (fault 1) represents a situation when a window is opened in the first room, which can cause energy waste in order to satisfy a temperature setpoint. This situation can be modeled by augmenting the state space model with the unknown input state that represents the additional heat flow for room 1

$$q_{\mathrm{d},k+1} = q_{\mathrm{d},k} + w_{q_\mathrm{d}}, \quad w_{q_\mathrm{d}} \sim \mathcal{N}\left(0, Q_{q_\mathrm{d}}\right). \tag{5.50}$$

Also, the new state is added to the heat flow balance equation of room 1. The second disturbance (fault 2) is the ambient temperature measurement fault, representing a situation with sun radiation heating an ambient temperature thermometer. In this case, the thermometer provides information about outside air temperature. This disturbance is modeled as the extra state

$$T_{\mathrm{d},k+1} = T_{\mathrm{d},k} + w_{T_\mathrm{d}}, \quad w_{T_\mathrm{d}} \sim \mathcal{N}\left(0, Q_{T_\mathrm{d}}\right). \tag{5.51}$$

Then the real ambient temperature is computed as the disturbed measurement subtracted by disturbance

$$T_{\mathrm{amb}} = T_{\mathrm{amb,sensor}} - T_\mathrm{d}. \tag{5.52}$$

The two disturbance modes were selected to illustrate both local and global disturbances. The heat flow disturbance represents the local disturbance because it mainly affects one room. The ambient temperature has an impact on all rooms. Distinguishing the disturbances will become more relevant in Chapter 7, which is devoted to distributed FDD methods.

In the experiment, the MJLS model with four system modes is used: the nominal model (model without fault), the model with fault 1 (opened window), the model with fault 2 (disturbed ambient temperature), and the model with both faults (faults 1 and 2). In that order, the state vectors of modes are defined as

$$\mathbf{x}^{(1)} \triangleq \begin{bmatrix} \mathbf{x} \\ 0 \\ 0 \end{bmatrix}, \mathbf{x}^{(2)} \triangleq \begin{bmatrix} \mathbf{x} \\ q_\mathrm{d} \\ 0 \end{bmatrix}, \mathbf{x}^{(3)} \triangleq \begin{bmatrix} \mathbf{x} \\ 0 \\ T_\mathrm{d} \end{bmatrix}, \mathbf{x}^{(4)} \triangleq \begin{bmatrix} \mathbf{x} \\ q_\mathrm{d} \\ T_\mathrm{d} \end{bmatrix}. \tag{5.53}$$

The state vector dimensions must be unified to mix them in the interaction and combination steps. The unbiased method is used in the interaction step, where the full states are completed with the best available estimates from the last combination step. Completion is done in the combination step using zero mean value and expected variance. This setting proved to be the best in accurately estimating the disturbance state. The mode transition probability matrix is defined as

$$\mathbf{T} = \begin{bmatrix} 0.998 & 0.001 & 0.001 & 0 \\ 0.14 & 0.859 & 0 & 0.001 \\ 0.001 & 0 & 0.998 & 0.001 \\ 0 & 0.001 & 0.14 & 0.859 \end{bmatrix}, \tag{5.54}$$

where the modes are ordered as mentioned above (nominal, fault 1, fault 2, both faults). The matrix is created by determining the probabilities of opening/closing the window (0.1%/14%) and heated/non-heated thermometer (0.1%). Transition to the mode with both faults can be done only from single fault modes and vice versa. Also, the transitions between single-fault modes and from no-fault to combined-fault mode (and vice versa) have zero probability.

The performance of the proposed algorithms was tested in the MATLAB Simulink. The system was tested using Monte Carlo simulations for various faults. The simulations were run with the constant ambient temperature between $-10\,°C$ to $10\,°C$, and heating setpoints were always $23\,°C$. The initial conditions of the system and estimators were $20\,°C$ for all zone and wall temperatures in the state vector. The white Gaussian noise with variance $\sigma^2 = 0.05^2$ was added to all temperature measurements in rooms to simulate real measurements.

All experiments consisted of responding to initial conditions and at least one fault occurrence. The total simulation time of each experiment was one hour. Each experiment was repeated 100 times using the Monte Carlo simulations. The steady states of the simulated disturbances depend on the ambient conditions that are generated randomly for each run.

The results are interpreted by the mode probabilities because the decision mechanism can differ from application to application but it is always linked to the mode probability. The figures with mode probability results show the mean and standard deviation computed over mode probability responses in 100 Monte Carlo simulations. To improve the readability, the standard IMM the accurate model will be referred to as the perfect IMM-KF, and the standard IMM with the average model will be referred to as the imperfect IMM-KF.

In the first scenario, the window was opened at time $t = 30\,\text{min}$ and then closed at time $t = 42\,\text{min}$. The disturbance simulations are shown in Fig. 5.3.

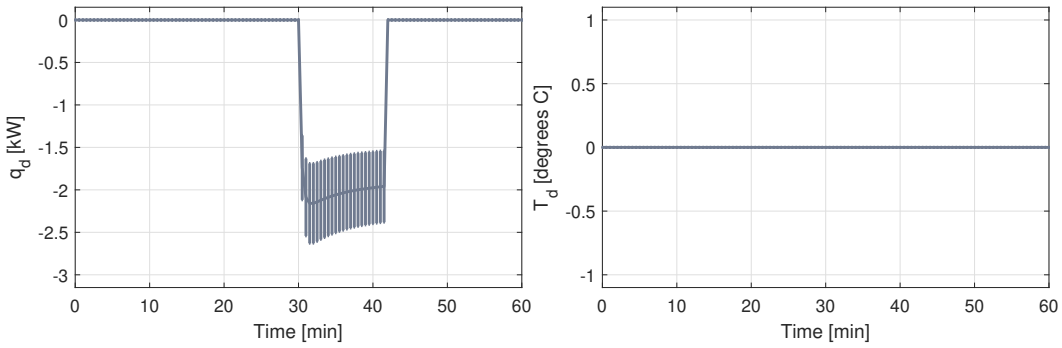

Figure 5.3. Opened window fault. The figure shows the simulated disturbances. The vertical lines represent standard deviations of disturbances from 100 Monte Carlo simulations.

The difference between using the accurate model and the average model is clear from the results of the perfect IMM-KF in Fig. 5.4 and the results of the imperfect IMM-KF in Fig. 5.5. The results of the imperfect IMM-KF show that before the fault occurrence, the Fault 2 mode is the most probable, which is wrong. Also, the standard deviation is high, meaning the results are inconsistent. Therefore, the imperfect IMM-KF is not satisfactory for detecting this fault.

The results of the IMM-SDKF and IMM-XDKF-Z are shown in Fig. 5.6 and Fig. 5.7 respectively. The high mode probability of the correct modes, a clear difference between

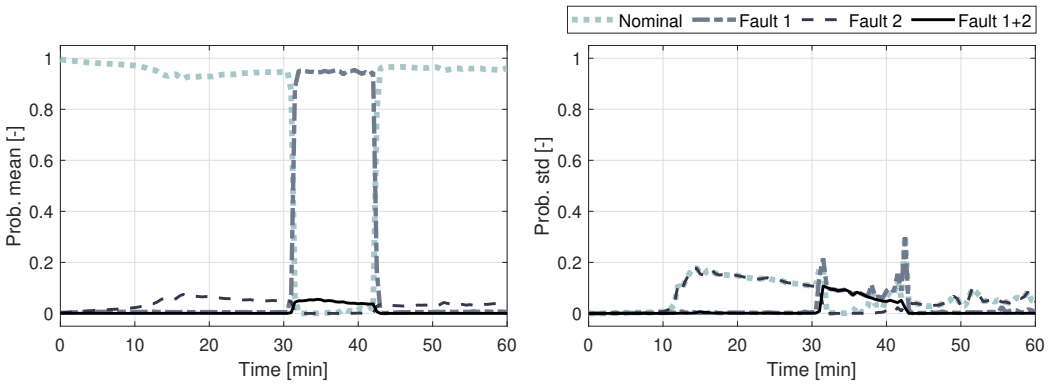

Figure 5.4. Opened window fault. The figure shows the mode probability of the standard IMM with the accurate zone model (perfect IMM-KF).

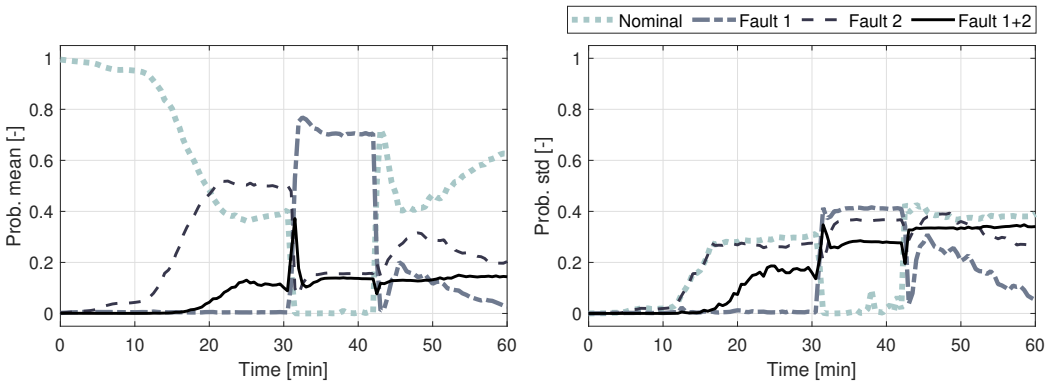

Figure 5.5. Opened window fault. The figure shows the mode probabilities of the standard IMM with the average zone model (imperfect IMM-KF).

the mean values, and a low standard deviation indicate that both desensitized IMM algorithms can detect the fault clearly and consistently. The performance of the IMM-SDKF and IMM-XDKF-Z is similar to the perfect IMM-KF, so the average zone model does not deteriorate it.

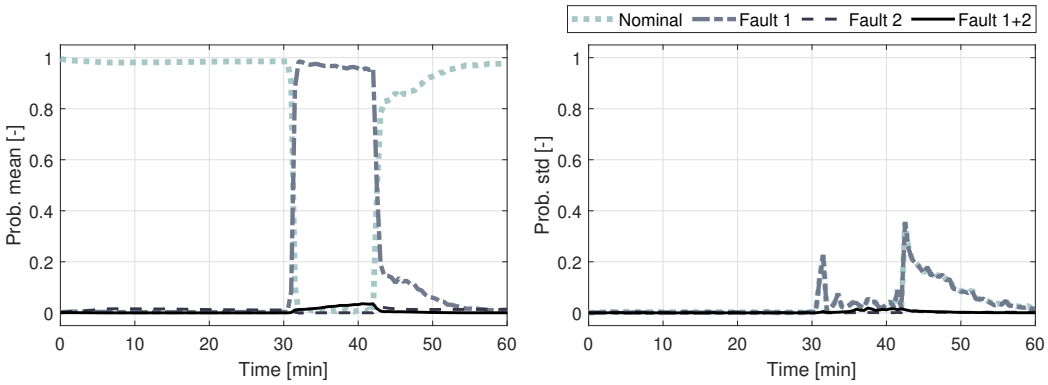

Figure 5.6. Opened window fault. The figure shows the mode probabilities of the IMM-SDKF with the average zone model.

The second scenario simulates the gradually heated thermometer of outside air temperature. The disturbance begins at time $t = 20\,\text{min}$ and the disturbance temperature gradually goes to $10\,°\text{C}$ as it is shown in Fig. 5.8.

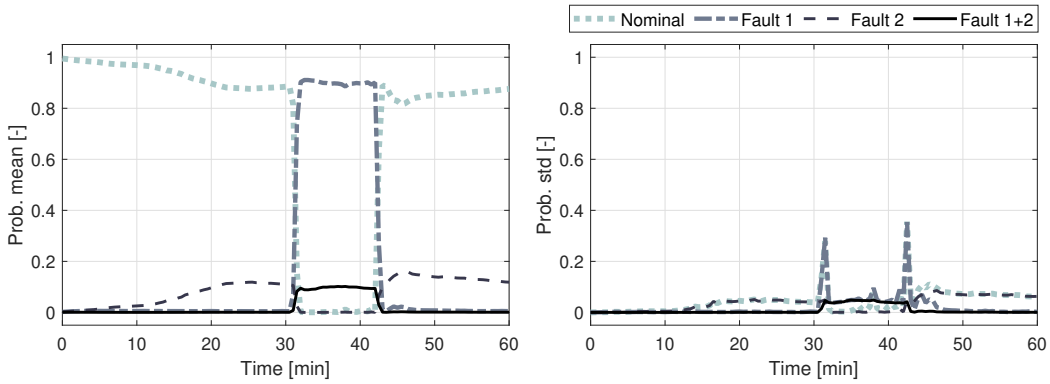

Figure 5.7. Opened window fault. The figure shows the mode probabilities of the IMM-XDKF-Z with the average zone model.

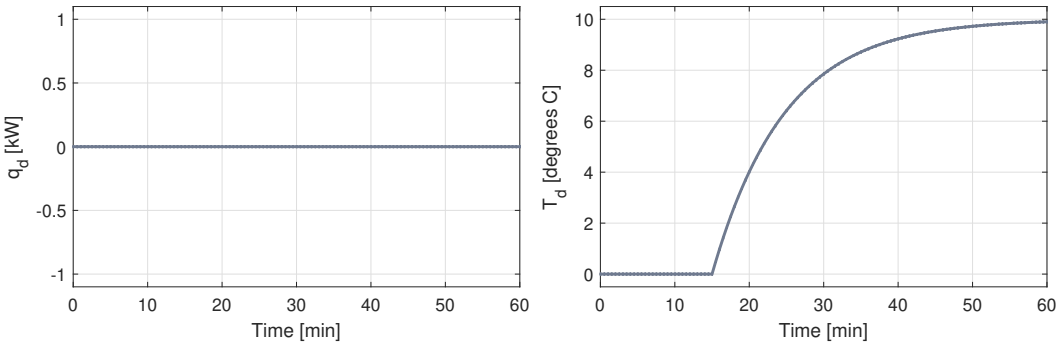

Figure 5.8. Heated thermometer fault. The figure shows the simulated disturbances.

The result of the perfect IMM-KF and the imperfect IMM-KF are shown in Fig. 5.9 and Fig. 5.10, respectively. The heated thermometer is detectable using the perfect IMM-KF, whereas, in the imperfect IMM-KF, the combined fault has the highest probability during the fault occurrence. Therefore the average model in the imperfect IMM-KF significantly deteriorates detection performance.

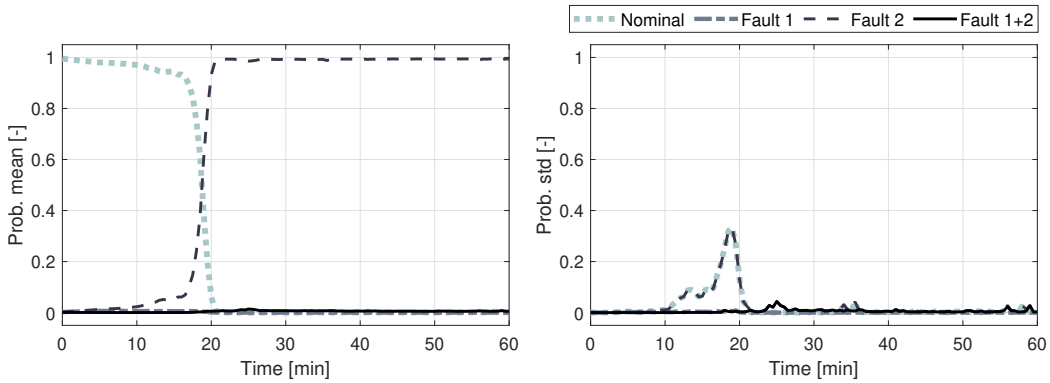

Figure 5.9. Heated thermometer fault. The figure shows the mode probabilities of the standard IMM with the accurate zone model (perfect IMM-KF).

The results for desensitized IMM algorithms are in Fig. 5.11 and Fig. 5.12. The IMM-SDKF and the IMM-XDKF-Z have delayed detection by a few minutes compared to the perfect IMM-KF. Otherwise, their performance results are again very similar.

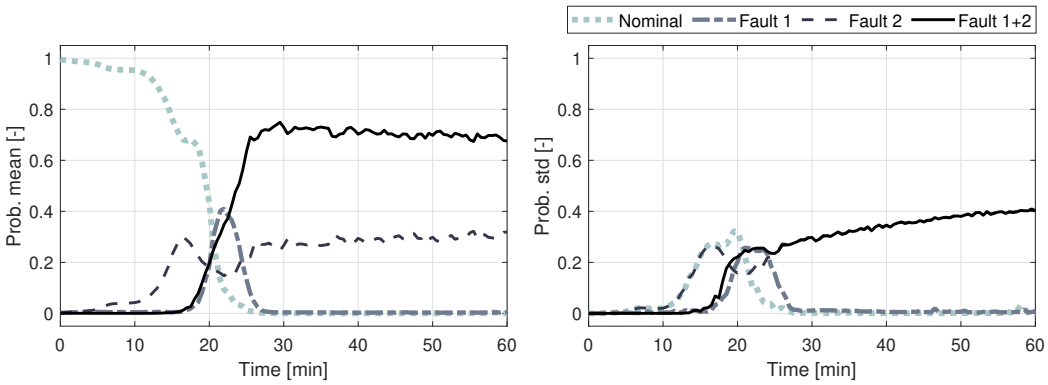

Figure 5.10. Heated thermometer fault. The figure shows the mode probabilities of the standard IMM with the average zone model (imperfect IMM-KF).

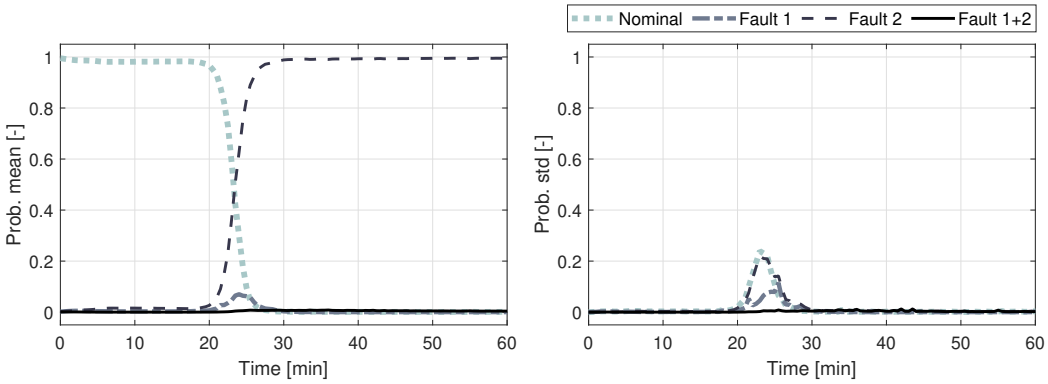

Figure 5.11. Heated thermometer fault. The figure shows the mode probabilities of the IMM-SDKF with the average zone model.

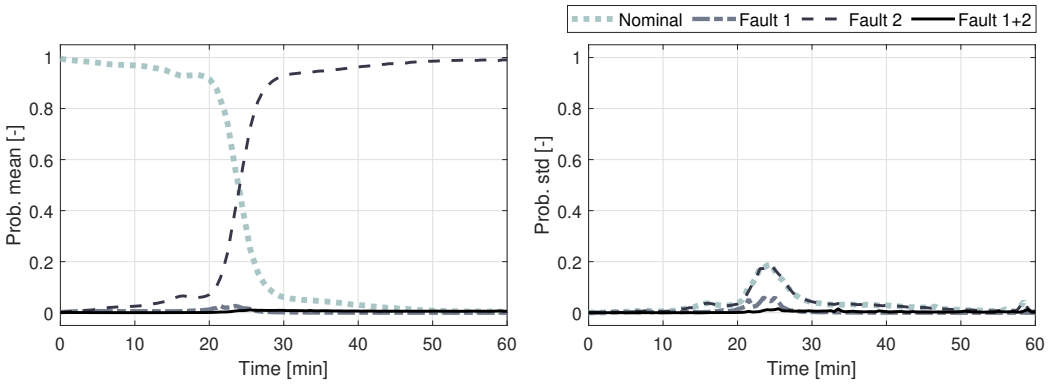

Figure 5.12. Heated thermometer fault. The figure shows the mode probabilities of the IMM-XDKF-Z with the average zone model.

The fault can be clearly and consistently detected by both the IMM-SDKF and the IMM-XDKF-Z.

The last experiment presents the detection of simultaneous faults. In the simulation, the Fault 1 and Fault 2 scenarios run simultaneously, i.e., the thermometer is heated, then the window is opened for 12 min. The simulation of disturbances is shown in Fig. 5.13.

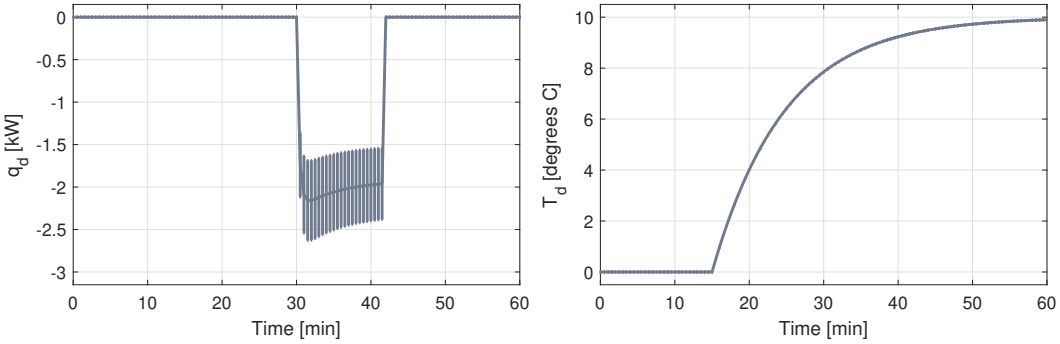

Figure 5.13. Simultaneous faults. The figure shows the simulated disturbances. The vertical lines represent standard deviations computed from 100 Monte Carlo simulations.

The faults are accurately detected using the perfect IMM-KF shown in Fig. 5.14. In the previous results, the imperfect IMM-KF could not detect the separate faults, so the detection of both faults was not expected. These expectations are confirmed by the results of the imperfect IMM-KF in Fig. 5.15. The mode probabilities do not correspond to the real modes, so the imperfect IMM-KF cannot detect the faults.

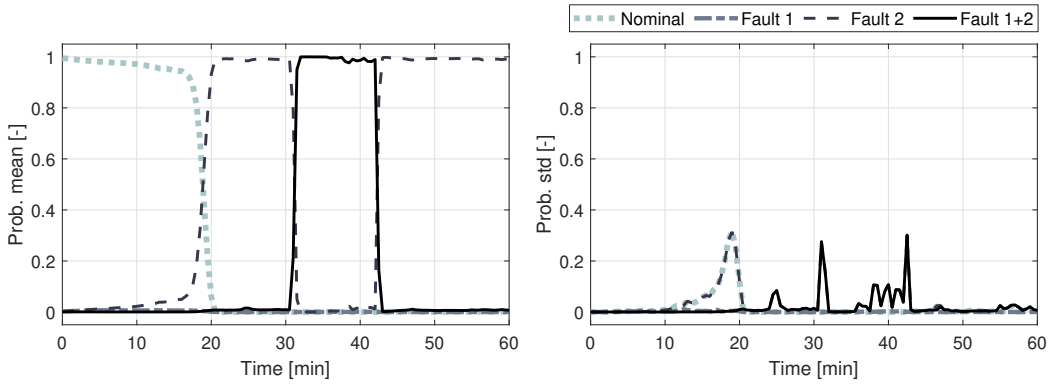

Figure 5.14. Simultaneous faults. The figure shows the mode probabilities of the standard IMM with the accurate zone model (perfect IMM-KF).

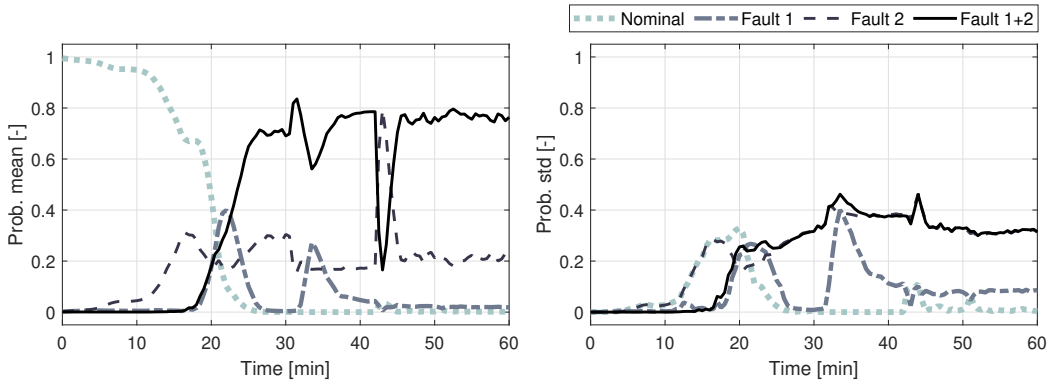

Figure 5.15. Simultaneous faults. The figure shows the mode probabilities of the standard IMM with the average zone model (imperfect IMM-KF).

Finally, the results of the IMM-SDKF and the IMM-XDKF-Z are shown in Fig. 5.16 and Fig. 5.17 respectively. The mode probabilities mean values of the IMM-SDKF

correspond to the real modes. In the recovery phase after the windows are closed (end of Fault 1), the mode probability of the combined faults (Fault 1+2) is close to the mode probability of the heated thermometer (Fault 2), and the standard deviation is higher. In this phase, the distinguishing between modes might not be clear. However, this unclear phase lasts only for about 10 min. Then the modes are correctly and consistently distinguished. The mode probabilities of the IMM-XDKF-Z are estimated correctly and consistently. Its performance is similar to the perfect IMM-KF. Both the IMM-SDKF and the IMM-XDKF-Z proved suitable for detection in the scenario with simultaneous faults.

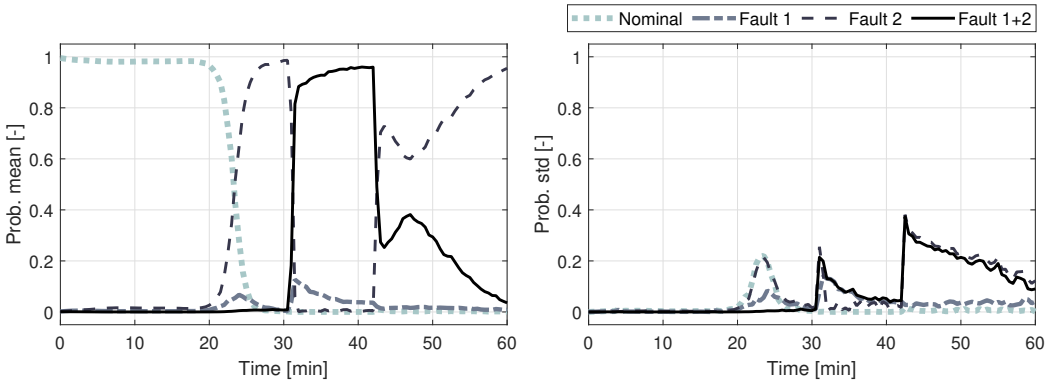

Figure 5.16. Simultaneous faults. The figure shows the mode probabilities of the IMM-SDKF with the average zone model.

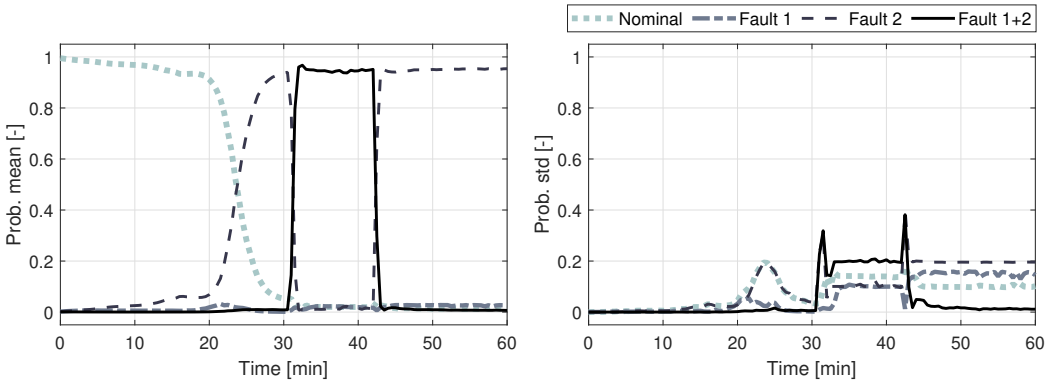

Figure 5.17. Simultaneous faults. The figure shows the mode probabilities of the IMM-XDKF-Z with the average zone model.

Chapter 6
Distributed desensitized state estimation

Distributed formulation of the exact desensitized Kalman filter can be very useful for application on large-scale networks. The goal is to obtain state estimates of local subsystems given the information from their closest neighbors. There are two areas where the desensitized algorithm can improve the local state estimation. First, the local estimation can be improved when local subsystems have uncertain parameters. Second, the local estimation can be improved by reducing the sensitivity to the information obtained from neighbors. This approach helps keep low communication burden between neighbors while considering the uncertainty of neighbor estimates. The results in this chapter were published in [J2].

6.1 Distributed linear stochastic system

The distributed state estimation method for linear systems interconnected by linear combinations of neighbors' state vectors is proposed in this section. The method assumes a network of linear interconnected non-overlapping systems with the local node system

$$\begin{aligned} \mathbf{x}_{k+1}^{(i)} &= \mathbf{A}^{(i)}\left(\boldsymbol{\theta}^{(i)}\right)\mathbf{x}_k^{(i)} + \mathbf{B}^{(i)}\left(\boldsymbol{\theta}^{(i)}\right)\mathbf{u}_k^{(i)} + \sum_{j\in\mathcal{G}^{(i)}} \mathbf{G}_j^{(i)}\mathbf{x}_k^{(j)} + \mathbf{v}_k^{(i)}, \\ \mathbf{y}_k^{(i)} &= \mathbf{C}^{(i)}\mathbf{x}_k^{(i)} + \mathbf{e}_k^{(i)}, \end{aligned} \tag{6.1}$$

where the integer i in the superscript brackets identifies the particular network node, $\mathbf{x}_k^{(i)}$ is the node state vector at time k, $\mathbf{u}_k^{(i)}$ is the node input vector, $\mathbf{y}_k^{(i)}$ is the node measured output vector, $\mathbf{v}_k^{(i)}, \mathbf{e}_k^{(i)}$ are the process and measurement white noise sequences, j denotes the adjacent node from the neighborhood set $\mathcal{G}^{(i)} = \{j_1^{(i)}, \ldots, j_{N_{\mathrm{g},i}}^{(i)}\}$, and $N_{\mathrm{g},i}$ is the number of neighbors of node i. The process and measurement noise statistics are

$$\begin{bmatrix} \mathbf{v}_k^{(i)} \\ \mathbf{e}_k^{(i)} \end{bmatrix} = \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{Q}^{(i)} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}^{(i)} \end{bmatrix}\right). \tag{6.2}$$

Furthermore, noise inputs are not correlated across nodes

$$\mathrm{E}\left[\mathbf{v}_k^{(i)}\left(\mathbf{v}_k^{(j)}\right)^T\right] = \mathbf{0},\ \mathrm{E}\left[\mathbf{v}_k^{(i)}\left(\mathbf{e}_k^{(j)}\right)^T\right] = \mathbf{0},\ \mathrm{E}\left[\mathbf{e}_k^{(i)}\left(\mathbf{e}_k^{(j)}\right)^T\right] = \mathbf{0}, \quad \text{for all } i \neq j. \tag{6.3}$$

The local systems depend on vectors of local uncertain parameters $\boldsymbol{\theta}^{(i)} \in \mathbb{R}^{N_{\theta,i}}$, where $N_{\theta,i}$ denotes the dimension of the local parameter vector at node i. The matrices $\mathbf{A}^{(i)}$, $\mathbf{B}^{(i)}$ are assumed to be linear matrix functions of parameters

$$\mathbf{A}^{(i)}\left(\boldsymbol{\theta}^{(i)}\right) = \mathbf{A}_0^{(i)} + \theta_1^{(i)}\mathbf{A}_1^{(i)} + \theta_2^{(i)}\mathbf{A}_2^{(i)} + \ldots + \theta_{N_{\theta,i}}^{(i)}\mathbf{A}_{N_{\theta,i}}^{(i)}, \tag{6.4}$$

$$\mathbf{B}^{(i)}\left(\boldsymbol{\theta}^{(i)}\right) = \mathbf{B}_0^{(i)} + \theta_1^{(i)}\mathbf{B}_1^{(i)} + \theta_2^{(i)}\mathbf{B}_2^{(i)} + \ldots + \theta_{N_{\theta,i}}^{(i)}\mathbf{B}_{N_{\theta,i}}^{(i)}. \tag{6.5}$$

The linear matrix function descriptions in (6.4), (6.5) are inspired by Taylor series expansion of a nonlinear function in the operating point $\boldsymbol{\theta}_{\text{op}}$

$$f(\boldsymbol{\theta}) = f(\boldsymbol{\theta}_{\text{op}}) + (\partial f(\boldsymbol{\theta})/\partial\theta_1)\,|_{\boldsymbol{\theta}_{\text{op}}}\Delta\theta_1 + (\partial f(\boldsymbol{\theta})/\partial\theta_2)\,|_{\boldsymbol{\theta}_{\text{op}}}\Delta\theta_2 + \ldots, \tag{6.6}$$

which is often used for linearization of nonlinear systems.

The local state estimation of the system (6.1) requires sharing state estimates and error covariances among neighbors. Sharing the state estimation error covariances increases the size of transferred data related to the state estimation information from $\mathcal{O}(N_x)$ to $\mathcal{O}(N_x^2)$, where N_x is a state vector dimension. Instead of using the error covariances, the uncertainty of neighbor state estimates can be expressed by additional uncertain parameters, which results in the modified local system description

$$\begin{aligned}\mathbf{x}_{k+1}^{(i)} &= \mathbf{A}^{(i)}\left(\boldsymbol{\theta}^{(i)}\right)\mathbf{x}_k^{(i)} + \mathbf{B}^{(i)}\left(\boldsymbol{\theta}^{(i)}\right)\mathbf{u}_k^{(i)} + \sum_{j\in\mathcal{G}^{(i)}}\rho_j^{(i)}\mathbf{G}_j^{(i)}\hat{\mathbf{x}}_k^{(j)} + \mathbf{v}_k^{(i)},\\ \mathbf{y}_k^{(i)} &= \mathbf{C}^{(i)}\mathbf{x}_k^{(i)} + \mathbf{e}_k^{(i)},\end{aligned} \tag{6.7}$$

where the estimates from neighbors are considered to be deterministic inputs multiplied by unknown uncertain parameters $\rho_j^{(i)}, j \in \mathcal{G}^{(i)}$. Putting these parameters in the parameter vector $\boldsymbol{\rho}^{(i)} \in \mathbb{R}^{N_{\text{g},i}}$ allows to formulate the extended parameter vector of the i-th local system as

$$\boldsymbol{\Theta}^{(i)} = \begin{bmatrix}\boldsymbol{\rho}^{(i)}\\ \boldsymbol{\theta}^{(i)}\end{bmatrix}, \tag{6.8}$$

with dimension $N_{\Theta,i} = N_{\text{g},i} + N_{\theta,i}$. The state estimation method for the local system (6.7) should include uncertainty of the parameters to reduce the impact of the parameter uncertainty on the state estimation error, and consequently, reduce the impact of the neighbor estimate uncertainty. In this case, the desensitized approach is chosen because it does not require a prior definition of the uncertainty, such as a probability distribution or an interval, but only the expected value is needed. In this particular application to the local system (6.7), the expected values of parameter vectors $\boldsymbol{\rho}^{(i)}$ are all one. In the desensitized approach, the reduction of the state estimation error sensitivity to uncertain parameters is tuned using weights. The proposed local desensitized Kalman filter algorithm (LXDKF) is derived in the following section.

6.2 Local desensitized Kalman filter

The local exact desensitized Kalman filter (LXDKF) for the local system (6.7) is derived in this section. The LXDKF is based on the exact desensitized Kalman filter with zero gain sensitivity and cumulative update in Algorithm 3.5.

When the true parameter values are known, a linear observer for the local system (6.7) estimates the state as

$$\begin{aligned}\hat{\mathbf{x}}_{k+1|k}^{(i)*} &= \mathbf{A}^{(i)}\left(\boldsymbol{\theta}^{(i)}\right)\hat{\mathbf{x}}_{k|k-1}^{(i)*} + \mathbf{B}^{(i)}\left(\boldsymbol{\theta}^{(i)}\right)\mathbf{u}_k + \sum_{j\in\mathcal{G}^{(i)}}\rho_j^{(i)}\mathbf{G}_j^{(i)}\hat{\mathbf{x}}_{k|k-1}^{(j)}\\ &\quad + \mathbf{K}_k^{(i)}\left(\mathbf{y}_k^{(i)} - \mathbf{C}^{(i)}\hat{\mathbf{x}}_{k|k-1}^{(i)}\right),\end{aligned} \tag{6.9}$$

and generates the state prediction error

$$\tilde{\mathbf{x}}_{k+1|k}^{(i)*} = \mathbf{x}_{k+1}^{(i)} - \hat{\mathbf{x}}_{k+1|k}^{(i)*}, \tag{6.10}$$

$$= \left(\mathbf{A}^{(i)}\left(\boldsymbol{\theta}^{(i)}\right) - \mathbf{K}_k^{(i)}\mathbf{C}^{(i)}\right)\tilde{\mathbf{x}}_{k|k-1}^{(i)*} + \mathbf{v}_k^{(i)} - \mathbf{K}_k^{(i)}\mathbf{e}_k^{(i)}, \tag{6.11}$$

where $\mathbf{K}_k^{(i)}$ is a time-variant observer gain. The mean value of the observer error (6.10) is zero. Therefore minimizing its covariance will result in the optimal observer in the minimum mean square error (MMSE) sense.

When the observer is designed using some estimated parameter vector $\hat{\boldsymbol{\Theta}}^{(i)}$, the true value is defined as

$$\boldsymbol{\Theta}^{(i)} = \hat{\boldsymbol{\Theta}}^{(i)} + \tilde{\boldsymbol{\Theta}}^{(i)} = \begin{bmatrix} \hat{\boldsymbol{\rho}}^{(i)} \\ \hat{\boldsymbol{\theta}}^{(i)} \end{bmatrix} + \begin{bmatrix} \hat{\boldsymbol{\rho}}^{(i)} \\ \hat{\boldsymbol{\theta}}^{(i)} \end{bmatrix}, \tag{6.12}$$

where $\tilde{\boldsymbol{\Theta}}^{(i)}$ denotes the parameter error, which is assumed to be unknown. The observer state prediction mean becomes

$$\begin{aligned} \hat{\mathbf{x}}_{k+1|k}^{(i)} &= \mathbf{A}^{(i)}\left(\hat{\boldsymbol{\theta}}^{(i)}\right)\hat{\mathbf{x}}_{k|k-1}^{(i)} + \mathbf{B}^{(i)}\left(\hat{\boldsymbol{\theta}}^{(i)}\right)\mathbf{u}_k^{(i)} + \sum_{j\in\mathcal{G}^{(i)}} \hat{\rho}_j^{(i)}\mathbf{G}_j^{(i)}\hat{\mathbf{x}}_{k|k-1}^{(j)} \\ &\quad + \mathbf{K}_k^{(i)}\left(\mathbf{y}_k^{(i)} - \mathbf{C}^{(i)}\hat{\mathbf{x}}_{k|k-1}^{(i)}\right), \end{aligned} \tag{6.13}$$

which generates the prediction error

$$\tilde{\mathbf{x}}_{k+1|k}^{(i)} = \mathbf{x}_{k+1}^{(i)} - \hat{\mathbf{x}}_{k+1|k}^{(i)}, \tag{6.14}$$

$$\begin{aligned} &= \left(\mathbf{A}^{(i)}\left(\hat{\boldsymbol{\theta}}^{(i)}\right) - \mathbf{K}_k^{(i)}\mathbf{C}^{(i)}\right)\tilde{\mathbf{x}}_{k|k-1}^{(i)} + \mathbf{v}_k^{(i)} - \mathbf{K}_k^{(i)}\mathbf{e}_k^{(i)} \\ &\quad + \tilde{\mathbf{A}}^{(i)}\mathbf{x}_k^{(i)} + \tilde{\mathbf{B}}^{(i)}\mathbf{u}_k^{(i)} + \sum_{j\in\mathcal{G}^{(i)}} \tilde{\rho}_j^{(i)}\mathbf{G}_j^{(i)}\hat{\mathbf{x}}_{k|k-1}^{(j)}, \end{aligned} \tag{6.15}$$

where the matrices $\tilde{\mathbf{A}}^{(i)}, \tilde{\mathbf{B}}^{(i)}$ are defined as

$$\begin{aligned} \tilde{\mathbf{A}}^{(i)} &\equiv \mathbf{A}^{(i)}\left(\boldsymbol{\theta}^{(i)}\right) - \mathbf{A}^{(i)}\left(\hat{\boldsymbol{\theta}}^{(i)}\right), \\ \tilde{\mathbf{B}}^{(i)} &\equiv \mathbf{B}^{(i)}\left(\boldsymbol{\theta}^{(i)}\right) - \mathbf{B}^{(i)}\left(\hat{\boldsymbol{\theta}}^{(i)}\right). \end{aligned} \tag{6.16}$$

The notations $\hat{\mathbf{A}}^{(i)} \equiv \mathbf{A}^{(i)}\left(\hat{\boldsymbol{\theta}}^{(i)}\right)$ and $\hat{\mathbf{B}}^{(i)} \equiv \mathbf{B}^{(i)}\left(\hat{\boldsymbol{\theta}}^{(i)}\right)$ will be used to keep the text compact.

Low sensitivity of prediction error to parameters guarantees that even large parameter variation $\tilde{\boldsymbol{\Theta}}^{(i)}$ will have a limited impact on the state estimate error $\tilde{\mathbf{x}}_{k+1|k}^{(i)}$. The sensitivity of prediction error (6.14) to a selected parameter $\hat{\Theta}_p^{(i)}$, where $p = 1, \ldots, N_{\Theta,i}$ and $N_{\Theta,i} = N_{\mathrm{g},i} + N_{\theta,i}$, is obtained as

$$\mathbf{s}_{p,k+1|k}^{(i)} \equiv \frac{d\tilde{\mathbf{x}}_{k+1|k}^{(i)}}{d\hat{\Theta}_p^{(i)}}, \tag{6.17}$$

$$\begin{aligned} &= \left(\hat{\mathbf{A}}^{(i)} - \mathbf{K}_k^{(i)}\mathbf{C}^{(i)}\right)\mathbf{s}_{p,k|k-1}^{(i)} + \frac{\partial\hat{\mathbf{A}}^{(i)}}{\partial\hat{\Theta}_p^{(i)}}\tilde{\mathbf{x}}_{k|k-1}^{(i)} \\ &\quad - \frac{\partial\hat{\mathbf{A}}^{(i)}}{\partial\hat{\Theta}_p^{(i)}}\mathbf{x}_k^{(i)} - \frac{\partial\hat{\mathbf{B}}^{(i)}}{\partial\hat{\Theta}_p^{(i)}}\mathbf{u}_k^{(i)} - \sum_{j\in\mathcal{G}^{(i)}} \frac{\partial\hat{\rho}_j^{(i)}}{\partial\hat{\Theta}_p^{(i)}}\mathbf{G}_j^{(i)}\hat{\mathbf{x}}_{k|k-1}^{(j)}, \end{aligned} \tag{6.18}$$

$$\begin{aligned} &= \left(\hat{\mathbf{A}}^{(i)} - \mathbf{K}_k^{(i)}\mathbf{C}^{(i)}\right)\mathbf{s}_{p,k|k-1}^{(i)} \\ &\quad - \mathbf{A}_p^{(i)}\hat{\mathbf{x}}_{k|k-1}^{(i)} - \mathbf{B}_p^{(i)}\mathbf{u}_k^{(i)} - \sum_{j\in\mathcal{G}^{(i)}} \delta_{j\,j_p^{(i)}}\mathbf{G}_j^{(i)}\hat{\mathbf{x}}_{k|k-1}^{(j)}, \end{aligned} \tag{6.19}$$

where $\mathbf{A}_p^{(i)}, \mathbf{B}_p^{(i)}$ are the matrix coefficients of linear matrix functions defined in (6.4) and (6.5), and $\delta_{j\,j_p^{(i)}}$ denotes the Kronecker delta, i.e. $\delta_{j\,j_p^{(i)}}$ is one if $j = j_p^{(i)}$ and zero otherwise. Also, when $p > N_{\mathrm{g},i}$, neighbor $j_p^{(i)}$ does not exist and $\delta_{j\,j_p^{(i)}}$ is zero. Furthermore, it is assumed that the impact of the gain sensitivity is negligible

$$\frac{d\mathbf{K}_k^{(i)}}{d\hat{\Theta}_p^{(i)}} = 0. \tag{6.20}$$

The assumption (6.20) is fully justified in Section 3.2. The mean value notation will be used to emphasize that the sensitivity in (6.19) is a deterministic variable

$$\mathbf{s}_{p,k+1|k}^{(i)} = \mathbb{E}\left[\hat{\mathbf{s}}_{p,k+1|k}^{(i)}\right] = \hat{\mathbf{s}}_{p,k+1|k}^{(i)}. \tag{6.21}$$

As mentioned in the previous sections, the desensitized approach uses two objectives. First, the state estimation error is minimized as in the standard Kalman filter. Second, the state prediction error sensitivity to uncertain parameters is minimized. The goal is achieved by minimizing the optimization criterion defined as a convex combination of individual objectives

$$\begin{aligned} J^{(i)} &= \alpha^{(i)}\,\mathrm{tr}\,\mathbb{E}\left[\tilde{\mathbf{x}}_{k+1|k}^{(i)}\left(\tilde{\mathbf{x}}_{k+1|k}^{(i)}\right)^T\right] + \sum_{p=1}^{N_{\Theta,i}} \gamma_p^{(i)}\mathrm{tr}\left(\hat{\mathbf{s}}_{p,k+1|k}^{(i)}\left(\hat{\mathbf{s}}_{p,k+1|k}^{(i)}\right)^T\right), \\ &\text{where } \alpha^{(i)} \equiv 1 - \sum_{p=1}^{N_{\Theta,i}} \gamma_p^{(i)},\ 0 \le \sum_{p=1}^{N_{\Theta,i}} \gamma_p^{(i)} < 1,\ \text{and } \gamma_p^{(i)} \ge 0. \end{aligned} \tag{6.22}$$

The following notation is used for the covariance notation in the next text

$$\mathbf{P}_k^{(i)} \equiv \mathbb{E}\left[\tilde{\mathbf{x}}_k^{(i)}\left(\tilde{\mathbf{x}}_k^{(i)}\right)^T\right]. \tag{6.23}$$

The state estimation error covariance is defined as

$$\begin{aligned} \mathbf{P}_{k+1|k}^{(i)} = &\left(\hat{\mathbf{A}}^{(i)} - \mathbf{K}_k^{(i)}\mathbf{C}^{(i)}\right)\mathbf{P}_{k|k-1}^{(i)}\left(\hat{\mathbf{A}}^{(i)} - \mathbf{K}_k^{(i)}\mathbf{C}^{(i)}\right)^T \\ &+ \mathbf{K}_k^{(i)}\mathbf{R}^{(i)}\left(\mathbf{K}_k^{(i)}\right)^T + \mathbf{Q}^{(i)}. \end{aligned} \tag{6.24}$$

The sensitivity product is evaluated as

$$\begin{aligned} \hat{\mathbf{s}}_{p,k+1|k}^{(i)}\left(\hat{\mathbf{s}}_{p,k+1|k}^{(i)}\right)^T = &\left(\hat{\mathbf{A}}^{(i)} - \mathbf{K}_k^{(i)}\mathbf{C}^{(i)}\right)\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\left(\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\right)^T\left(\hat{\mathbf{A}}^{(i)} - \mathbf{K}_k^{(i)}\mathbf{C}^{(i)}\right)^T \\ &+ \mathbf{b}_{p,k}^{(i)}\left(\mathbf{b}_{p,k}^{(i)}\right)^T \\ &- \mathbf{b}_{p,k}^{(i)}\left(\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\right)^T\left(\hat{\mathbf{A}}^{(i)} - \mathbf{K}_k^{(i)}\mathbf{C}^{(i)}\right)^T \\ &- \left(\hat{\mathbf{A}}^{(i)} - \mathbf{K}_k^{(i)}\mathbf{C}^{(i)}\right)\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\left(\mathbf{b}_{p,k}^{(i)}\right)^T, \end{aligned} \tag{6.25}$$

where

$$\mathbf{b}_{p,k}^{(i)} = \mathbf{A}_p^{(i)}\hat{\mathbf{x}}_{k|k-1}^{(i)} + \mathbf{B}_p^{(i)}\mathbf{u}_k^{(i)} + \sum_{j\in\mathcal{G}^{(i)}} \delta_{j\,j_p}\mathbf{G}_j^{(i)}\hat{\mathbf{x}}_{k|k-1}^{(j)}. \tag{6.26}$$

The optimal gain $\mathbf{K}_k^{(i)}$ can be found from the first order optimality condition

$$\frac{\partial J}{\partial \mathbf{K}_k^{(i)}} = 0. \tag{6.27}$$

The derivatives of the elements of criterion (6.22) required to evaluate (6.27) are

$$\begin{aligned}\frac{\partial \operatorname{tr}\mathbf{P}_{k+1|k}^{(i)}}{\partial \mathbf{K}_k^{(i)}} &= -2\hat{\mathbf{A}}^{(i)}\mathbf{P}_{k|k-1}^{(i)}\left(\mathbf{C}^{(i)}\right)^T \\ &\quad + 2\mathbf{K}_k^{(i)}\left(\mathbf{C}^{(i)}\mathbf{P}_{k|k-1}^{(i)}\left(\mathbf{C}^{(i)}\right)^T + \mathbf{R}^{(i)}\right), \end{aligned} \tag{6.28}$$

$$\begin{aligned}\frac{\partial \operatorname{tr}\left(\hat{\mathbf{s}}_{p,k+1|k}^{(i)}\left(\hat{\mathbf{s}}_{p,k+1|k}^{(i)}\right)^T\right)}{\partial \mathbf{K}_k^{(i)}} &= -2\hat{\mathbf{A}}^{(i)}\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\left(\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\right)^T\left(\mathbf{C}^{(i)}\right)^T \\ &\quad + 2\mathbf{K}_k^{(i)}\mathbf{C}^{(i)}\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\left(\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\right)^T\left(\mathbf{C}^{(i)}\right)^T \\ &\quad + 2\mathbf{b}_{p,k}^{(i)}\left(\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\right)^T\left(\mathbf{C}^{(i)}\right)^T. \end{aligned} \tag{6.29}$$

The resulting gain equation is obtained by substituting (6.28) and (6.29) to (6.27)

$$\begin{aligned}&\mathbf{K}_k^{(i)}\left[\mathbf{C}^{(i)}\left(\alpha^{(i)}\mathbf{P}_{k|k-1}^{(i)} + \sum_{p=1}^{N_{\Theta,i}}\gamma_p^{(i)}\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\left(\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\right)^T\right)\left(\mathbf{C}^{(i)}\right)^T + \alpha^{(i)}\mathbf{R}^{(i)}\right] = \\ &= \hat{\mathbf{A}}^{(i)}\left(\alpha^{(i)}\mathbf{P}_{k|k-1}^{(i)} + \sum_{p=1}^{N_{\Theta,i}}\gamma_p^{(i)}\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\left(\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\right)^T\right)\left(\mathbf{C}^{(i)}\right)^T - \\ &\quad - \sum_{p=1}^{N_{\Theta,i}}\gamma_p^{(i)}\mathbf{b}_{p,k}^{(i)}\left(\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\right)^T\left(\mathbf{C}^{(i)}\right)^T. \end{aligned} \tag{6.30}$$

A rank-1 update of the covariance results in a positive semi-definite matrix. Therefore, the multiplier of the gain $\mathbf{K}_k^{(i)}$ in (6.30) is a positive definite matrix, and the gain can be defined as

$$\mathbf{K}_k^{(i)} = \left[\hat{\mathbf{A}}^{(i)}\mathbf{P}_{k|k-1}^{\Sigma,(i)}\left(\mathbf{C}^{(i)}\right)^T + \mathbf{S}_k^{(i)}\right]\left[\mathbf{C}^{(i)}\mathbf{P}_{k|k-1}^{\Sigma,(i)}\left(\mathbf{C}^{(i)}\right)^T + \alpha^{(i)}\mathbf{R}^{(i)}\right]^{-1}, \tag{6.31}$$

where

$$\mathbf{P}_{k|k-1}^{\Sigma,(i)} = \alpha^{(i)}\mathbf{P}_{k|k-1}^{(i)} + \sum_{p=1}^{N_{\Theta,i}}\gamma_p^{(i)}\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\left(\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\right)^T, \tag{6.32}$$

$$\mathbf{S}_k^{(i)} = -\sum_{p=1}^{N_{\Theta,i}}\gamma_p^{(i)}\mathbf{b}_{p,k}^{(i)}\left(\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\right)^T\left(\mathbf{C}^{(i)}\right)^T. \tag{6.33}$$

The cumulative matrix $\mathbf{P}_{k|k-1}^{\Sigma,(i)}$ in (6.32) is a positive semi-definite matrix. Then it is sufficient and convenient to update the cumulative moment $\mathbf{P}_{k|k-1}^{\Sigma,(i)}$ instead of the separate covariance update $\mathbf{P}_{k|k-1}^{(i)}$. The cumulative moment update equation is obtained as

$$\begin{aligned}\mathbf{P}_{k+1|k}^{\Sigma,(i)} &= \hat{\mathbf{A}}^{(i)}\mathbf{P}_{k|k-1}^{\Sigma,(i)}\left(\hat{\mathbf{A}}^{(i)}\right)^T + \mathbf{Q}_{k|k-1}^{\Sigma,(i)} \\ &\quad - \mathbf{K}_k^{(i)}\left[\mathbf{C}^{(i)}\mathbf{P}_{k|k-1}^{\Sigma,(i)}\left(\hat{\mathbf{A}}^{(i)}\right)^T + \left(\mathbf{S}_k^{(i)}\right)^T\right], \end{aligned} \tag{6.34}$$

where the cumulative process noise matrix is

$$\mathbf{Q}_{k|k-1}^{\Sigma,(i)} = \alpha^{(i)}\mathbf{Q}^{(i)} + \sum_{p=1}^{N_{\Theta,i}} \gamma_p^{(i)} \Bigg[\mathbf{b}_{p,k}^{(i)} \left(\mathbf{b}_{p,k}^{(i)}\right)^T - \\ - \mathbf{b}_{p,k}^{(i)} \left(\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\right)^T \left(\hat{\mathbf{A}}^{(i)}\right)^T - \hat{\mathbf{A}}^{(i)} \hat{\mathbf{s}}_{p,k|k-1}^{(i)} \left(\mathbf{b}_{p,k}^{(i)}\right)^T \Bigg]. \tag{6.35}$$

The weighted cumulative update in (6.34) is a Riccati difference equation (RDE). This update can be interpreted as a covariance update of the standard Kalman filter with the correlated process and measurement noise and time-varying covariance matrices

$$\begin{bmatrix} \mathbf{v}_k^{(i)} \\ \mathbf{e}_k^{(i)} \end{bmatrix} = \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{Q}_{k|k-1}^{\Sigma,(i)} & \mathbf{S}_k^{(i)} \\ \left(\mathbf{S}_k^{(i)}\right)^T & \alpha^{(i)}\mathbf{R}^{(i)} \end{bmatrix} \right). \tag{6.36}$$

The algorithms with separate data-update and time-update steps are more suitable for distributed state estimation since the information can be shared after each time iteration and between the steps. The particular steps are described as follows. The data-update step obtains the state estimate $\mathbf{x}_{k|k}$ based on the measurements at time k and the prior state estimate $\mathbf{x}_{k|k-1}$. The time-update step obtains the state estimate $\mathbf{x}_{k+1|k}$ based on the model and the posterior state estimate $\mathbf{x}_{k|k}$.

Thanks to the LXDKF similarity with the standard Kalman filter, the single-step LXDKF can be separated by decorrelating the process and measurement noise with mutual correlation defined by (6.33). The steps can be separated using the approach in Section 3.2.4, which results in the decorrelated process and measurement noise

$$\begin{bmatrix} \mathbf{v}_k^{\mathrm{dc},(i)} \\ \mathbf{e}_k^{(i)} \end{bmatrix} = \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{Q}_{k|k}^{\Sigma,(i)} & \mathbf{0} \\ \mathbf{0} & \alpha^{(i)}\mathbf{R}^{(i)} \end{bmatrix} \right), \tag{6.37}$$

where

$$\mathbf{Q}_{k|k}^{\Sigma,(i)} = \mathbf{Q}_{k|k-1}^{\Sigma,(i)} - \mathbf{S}_k^{(i)} \left(\alpha^{(i)}\mathbf{R}^{(i)}\right)^{-1} \left(\mathbf{S}_k^{(i)}\right)^T. \tag{6.38}$$

The state propagation equation (6.7) is transformed into

$$\mathbf{x}_{k+1}^{(i)} = \left(\mathbf{A}^{(i)}\left(\boldsymbol{\theta}^{(i)}\right) - \mathbf{S}_k^{(i)} \left(\alpha^{(i)}\mathbf{R}^{(i)}\right)^{-1} \mathbf{C}^{(i)}\right) \mathbf{x}_k^{(i)} + \mathbf{B}^{(i)}\left(\boldsymbol{\theta}^{(i)}\right) \mathbf{u}_k^{(i)} + \\ + \sum_{j \in \mathcal{G}^{(i)}} \rho_j^{(i)} \mathbf{G}_j^{(i)} \hat{\mathbf{x}}_k^{(j)} - \mathbf{S}_k^{(i)} \left(\alpha^{(i)}\mathbf{R}^{(i)}\right)^{-1} \mathbf{y}_k^i + \mathbf{v}_k^{\mathrm{dc},(i)}. \tag{6.39}$$

Then the algorithm can be split into a data-update step and a time-update step, as shown in Section 3.2.4. The final two-step algorithm is described as follows.

Algorithm 6.1 (Local eXact Desensitized Kalman Filter — LXDKF). The LXDKF algorithm requires initialization of the weights $\gamma_p^{(i)}$ and the prior state mean $\hat{\mathbf{x}}_{k|k-1}^{(i)}$, the cumulative moment $\mathbf{P}_{k|k-1}^{\Sigma,(i)}$, and the state estimation error sensitivities to parameters $\hat{\mathbf{s}}_{p,k|k-1}^{(i)}$. The following LXDKF update steps need to be executed at each local node at each time iteration

- **Exchange data.** The prior state estimates $\hat{\mathbf{x}}_{k|k-1}^{(i)}$ are exchanged with neighbors. The local input and measurement data are obtained.
- **Uncertainty-update step.** The process noise covariance matrix and the cross-covariance matrix are updated.

$$\mathbf{b}_{p,k}^{(i)} = \mathbf{A}_p^{(i)}\hat{\mathbf{x}}_{k|k-1}^{(i)} + \mathbf{B}_p^{(i)}\mathbf{u}_k^{(i)} + \sum_{j\in\mathcal{G}^{(i)}} \delta_{j\,j_p}\mathbf{G}_j^{(i)}\hat{\mathbf{x}}_{k|k-1}^{(j)}, \tag{6.40}$$

$$\begin{aligned}\mathbf{Q}_{k|k-1}^{\Sigma,(i)} = \alpha^{(i)}\mathbf{Q}^{(i)} + \sum_{p=1}^{N_{\Theta,i}} \gamma_p^{(i)} \Bigg[& \mathbf{b}_{p,k}^{(i)} \left(\mathbf{b}_{p,k}^{(i)}\right)^T - \\ & - \mathbf{b}_{p,k}^{(i)} \left(\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\right)^T \left(\hat{\mathbf{A}}^{(i)}\right)^T - \hat{\mathbf{A}}^{(i)}\hat{\mathbf{s}}_{p,k|k-1}^{(i)} \left(\mathbf{b}_{p,k}^{(i)}\right)^T \Bigg],\end{aligned} \tag{6.41}$$

$$\mathbf{S}_k^{(i)} = -\sum_{p=1}^{N_{\Theta,i}} \gamma_p^{(i)}\mathbf{b}_{p,k}^{(i)} \left(\hat{\mathbf{s}}_{p,k|k-1}^{(i)}\right)^T \left(\mathbf{C}^{(i)}\right)^T, \tag{6.42}$$

  where $\mathbf{b}_{p,k}^{(i)}$ is an auxiliary variable used just to simplify the formulas (6.41) and (6.42).
- **Data-update step.** The state estimate, the sensitivity, and the cumulative moment are updated based on measurements.

$$\mathbf{K}_k^{(i)} = \mathbf{P}_{k|k-1}^{\Sigma,(i)} \left(\mathbf{C}^{(i)}\right)^T \left[\mathbf{C}^{(i)}\mathbf{P}_{k|k-1}^{\Sigma,(i)} \left(\mathbf{C}^{(i)}\right)^T + \alpha^{(i)}\mathbf{R}^{(i)}\right]^{-1}, \tag{6.43}$$

$$\hat{\mathbf{x}}_{k|k}^{(i)} = \hat{\mathbf{x}}_{k|k-1}^{(i)} + \mathbf{K}_k^{(i)} \left(\mathbf{y}_k^{(i)} - \mathbf{C}^{(i)}\hat{\mathbf{x}}_{k|k-1}^{(i)}\right), \tag{6.44}$$

$$\hat{\mathbf{s}}_{p,k|k}^{(i)} = \hat{\mathbf{s}}_{p,k|k-1}^{(i)} - \mathbf{K}_k^{(i)}\mathbf{C}^{(i)}\hat{\mathbf{s}}_{p,k|k-1}^{(i)}, \tag{6.45}$$

$$\mathbf{P}_{k|k}^{\Sigma,(i)} = \mathbf{P}_{k|k-1}^{\Sigma,(i)} - \mathbf{K}_k^{(i)}\mathbf{C}^{(i)}\mathbf{P}_{k|k-1}^{\Sigma,(i)}. \tag{6.46}$$

- **Exchange data.** The posterior state estimates $\hat{\mathbf{x}}_{k|k}^{(i)}$ are exchanged with neighbors.
- **Time-update step.** The state estimate, the sensitivity, and the cumulative moment are propagated in time using local models. Notice that the updates are done using the decorrelated system and noise matrices since the process and measurement noise are correlated by $\mathbf{S}_k^{(i)}$.

$$\hat{\mathbf{A}}_{\mathrm{dc}}^{(i)} = \hat{\mathbf{A}}^{(i)} - \mathbf{S}_k^{(i)} \left(\alpha^{(i)}\mathbf{R}^{(i)}\right)^{-1} \mathbf{C}^{(i)}, \tag{6.47}$$

$$\mathbf{Q}_{k|k}^{\Sigma,(i)} = \mathbf{Q}_{k|k-1}^{\Sigma,(i)} - \mathbf{S}_k^{(i)} \left(\alpha^{(i)}\mathbf{R}^{(i)}\right)^{-1} \left(\mathbf{S}_k^{(i)}\right)^T, \tag{6.48}$$

$$\hat{\mathbf{x}}_{k+1|k}^{(i)} = \hat{\mathbf{A}}_{\mathrm{dc}}^{(i)}\hat{\mathbf{x}}_{k|k}^{(i)} + \hat{\mathbf{B}}^{(i)}\mathbf{u}_k^{(i)} + \sum_{j\in\mathcal{G}^{(i)}} \hat{\rho}_j^{(i)}\mathbf{G}_j^{(i)}\hat{\mathbf{x}}_{k|k}^{(j)} + \mathbf{S}_k^{(i)} \left(\alpha^{(i)}\mathbf{R}^{(i)}\right)^{-1} \mathbf{y}_k^{(i)}, \tag{6.49}$$

$$\hat{\mathbf{s}}_{p,k+1|k}^{(i)} = \hat{\mathbf{A}}_{\mathrm{dc}}^{(i)}\hat{\mathbf{s}}_{p,k|k}^{(i)} - \mathbf{b}_{p,k}^{(i)}, \tag{6.50}$$

$$\mathbf{P}_{k+1|k}^{\Sigma,(i)} = \hat{\mathbf{A}}_{\mathrm{dc}}^{(i)}\mathbf{P}_{k|k}^{\Sigma,(i)} \left(\hat{\mathbf{A}}_{\mathrm{dc}}^{(i)}\right)^T + \mathbf{Q}_{k|k}^{\Sigma,(i)}. \tag{6.51}$$

The LXDKF in Algorithm 6.1 requires communication only between the adjacent nodes. The communication transfer between any two neighbors is shown in Fig. 6.1. It is required to transfer only the state estimate information twice at each iteration. In the case of a double-precision, a node needs to transfer $16 \times N_x$ bytes per step to each neighbor, which corresponds to complexity $\mathcal{O}(N_x)$. For comparison, the PKF need to transfer $16 \times (N_x + N_x^2)$ bytes per step, i.e. its complexity is $\mathcal{O}(N_x^2)$, because it needs to share the covariance matrix. Such transfer rate is not negligible because the network bandwidth is usually also needed for other purposes such as a distributed control algorithm, data logging, etc.

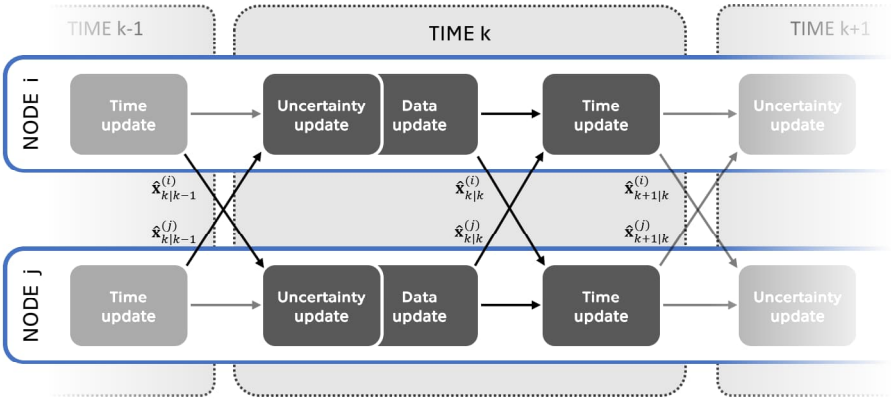

Figure 6.1. The LXDKF algorithm communication transfer between neighbors.

Chapter 7
Distributed desensitized fault detection and diagnosis

The IMM method that uses filtering methods for the FDD was discussed in Chapter 5. Similarly, the distributed FDD methods can be developed based on the distributed state estimation algorithms. This chapter describes a distributed model-based FDD method that is scalable, reusable, accurate and has a low communication burden. The results in this chapter were published in [J2].

7.1 Local IMM with desensitized filtering

Application of the IMM-based FDD to the network (6.7) consists of two parts. First, a set of local modes needs to be defined for each network node. Second, a local IMM is used for state estimation of a local system, where the LXDKF replaces Kalman filters.

The local Markov jump linear system is defined as

$$\begin{aligned} \mathbf{x}_{k+1}^{(i,m_i)} &= \mathbf{A}^{(i,m_i)}(\boldsymbol{\theta}^{(i)})\mathbf{x}_k^{(i,m_i)} + \mathbf{B}^{(i,m_i)}(\boldsymbol{\theta}^{(i)})\mathbf{u}_k^{(i)} + \sum_{j\in\mathcal{G}^{(i)}} \rho_j^{(i)}\mathbf{G}_j^{(i,m_i)}\hat{\mathbf{x}}_k^{(j)} + \mathbf{v}_k^{(i)}, \\ \mathbf{y}_k^{(i)} &= \mathbf{C}^{(i,m_i)}\mathbf{x}_k^{(i,m_i)} + \mathbf{e}_k^{(i)}, \end{aligned} \tag{7.1}$$

where i in the superscript brackets denotes the local network node number, j defines the neighbor network node number from neighborhood set $\mathcal{G}^{(i)} = \{j_1^{(i)}, j_2^{(i)}, \ldots, j_{N_{\mathrm{g},i}}^{(i)}\}$, m_i in the superscript brackets denotes the mode number at node i at time k, and m_i is from the predefined set $\mathcal{M}^{(i)} = \{1, 2, \ldots, N_{\mathrm{m},i}\}$. Other properties follow the properties of the local system (6.7). The neighbor estimate $\hat{\mathbf{x}}_k^{(j)}$ do not have any mode superscript m_j because the local node uses the estimate from neighbor as it is sent and have no information about adjacent node mode set $\mathcal{M}^{(j)}$.

The accuracy of the IMM can be improved if the mode set $\mathcal{M}^{(i)}$ is variable in time [67]. Such an approach can overcome a situation where a mode has a high probability even though a quantifiable reason exists that this mode cannot occur in a real scenario. In this thesis, the modes from the predefined set $\mathcal{M}^{(i)}$ are deactivated if they do not satisfy predefined local or global constraints. The examples of the local constraint $c_k^{(i,m_i)}$ are

- time constraint: the time has passed the initialization phase $t_k \geq t_{\mathrm{init}}$
- input constraint: an input value is within predefined bounds $u_{\mathrm{LB}} \geq u_k \geq u_{\mathrm{UB}}$
- state estimate constraint: a state estimate is within predefined bounds $x_{\mathrm{LB}} \geq \hat{x}_k \geq x_{\mathrm{UB}}$
- likelihood constraint: a difference between local mode likelihood values is greater than threshold $\Lambda_k^{(i,m_i)} - \Lambda_k^{(i,l_i)} \geq \epsilon$, where $l_i \in \mathcal{M}^{(i)}$.

The local constraint definition is not fixed to the given examples and can also be defined as a combination of constraints. The global constraints $C_k^{(i,m_i)}$ are defined as the logical

conjunction of the local constraints and selected local constraints in the neighbors' nodes If the constraints are not satisfied, the mode is deactivated by modifying the transition probability matrix. The modification starts with setting the probability of staying in the deactivated mode to zero. Then, the probability of the transition to deactivated mode is significantly reduced compared to the transition to active mode. Finally, the transition probability matrix is normalized after the modifications. The constraint mode activation results in the current transition probability matrix where the elements define the probability of transition from the mode l_i to the mode m_i

$$T_k^{(i)}(l_i, m_i) = P(M_{k+1} = m_i \,|M_k = l_i\,), \tag{7.2}$$

where M_k denotes the mode at time k. The local IMM algorithm needs to evaluate the constraint mode activation before mixing the mode state estimates.

The desensitized approach from the LXDKF in Algorithm 6.1 is used for the mode state estimation step to increase robustness against neighbor estimate errors This substitution results in the local IMM with constrained mode activation and filtering done by the LXDKF, which will be referred to as the LIMM-XDKF. The LIMM-XDKF requires initialization of the weights $\gamma_p^{(i)}$ and the prior estimates of the state mean $\hat{\mathbf{x}}_{k|k-1}^{(i,m_i)}$, the cumulative state estimation error covariance $\mathbf{P}_{k|k-1}^{\Sigma,(i,m_i)}$, the state estimation error sensitivities to parameters $\hat{\mathbf{s}}_{p,k|k-1}^{(i,m_i)}$, the prior mode probability $\boldsymbol{\mu}_{k-1}^{(i)}$, and the initial transition probability matrix $\mathbf{T}_0^{(i)}$. The following LIMM-XDKF update steps need to be executed at each local node at each time iteration:

Algorithm 7.1 (Local IMM with eXact Desensitized Kalman Filtering – LIMM-XDKF).

- **Exchange data.** The prior state estimates $\hat{\mathbf{x}}_{k|k-1}^{(i)}$ are exchanged with neighbors. The local input and measurement data are obtained.
- **Uncertainty-update step.** The variable noise covariance matrices are updated for each mode $m_i \in \mathcal{M}^{(i)}$.

$$\mathbf{b}_{p,k}^{(i,m_i)} = \mathbf{A}_p^{(i,m_i)}\hat{\mathbf{x}}_{k|k-1}^{(i,m_i)} + \mathbf{B}_p^{(i,m_i)}\mathbf{u}_k^{(i)} + \sum_{j\in\mathcal{G}^{(i)}} \delta_{jj_p}\mathbf{G}_j^{(i,m_i)}\hat{\mathbf{x}}_{k|k-1}^{(j)}, \tag{7.3}$$

$$\begin{aligned}\mathbf{Q}_{k|k-1}^{\Sigma,(i,m_i)} = \alpha^{(i)}\mathbf{Q}^{(i,m_i)} + \sum_{p=1}^{N_{\Theta,i}} \gamma_p^{(i)} \Bigg[& \mathbf{b}_{p,k}^{(i,m_i)} \left(\mathbf{b}_{p,k}^{(i,m_i)}\right)^T - \\ & - \mathbf{b}_{p,k}^{(i,m_i)} \left(\hat{\mathbf{s}}_{p,k|k-1}^{(i,m_i)}\right)^T \left(\hat{\mathbf{A}}^{(i,m_i)}\right)^T - \\ & - \hat{\mathbf{A}}^{(i,m_i)}\hat{\mathbf{s}}_{p,k|k-1}^{(i,m_i)} \left(\mathbf{b}_{p,k}^{(i,m_i)}\right)^T \Bigg],\end{aligned} \tag{7.4}$$

$$\mathbf{S}_k^{(i,m_i)} = -\sum_{p=1}^{N_{\Theta,i}} \gamma_p^{(i)}\mathbf{b}_{p,k}^{(i,m_i)} \left(\hat{\mathbf{s}}_{p,k|k-1}^{(i,m_i)}\right)^T \left(\mathbf{C}^{(i,m_i)}\right)^T. \tag{7.5}$$

- **Data-update step.**
  - The state estimate, the sensitivity, and the cumulative covariance are updated for each mode based on measurements

$$\mathbf{K}_k^{(i,m_i)} = \mathbf{P}_{k|k-1}^{\Sigma,(i,m_i)} \left(\mathbf{C}^{(i,m_i)}\right)^T \left[\mathbf{C}^{(i,m_i)}\mathbf{P}_{k|k-1}^{\Sigma,(i,m_i)} \left(\mathbf{C}^{(i,m_i)}\right)^T + \alpha^{(i)}\mathbf{R}^{(i,m_i)}\right]^{-1}, \tag{7.6}$$

$$\hat{\mathbf{x}}_{k|k}^{(i,m_i)} = \hat{\mathbf{x}}_{k|k-1}^{(i,m_i)} + \mathbf{K}_k^{(i,m_i)} \left(\mathbf{y}_k^{(i,m_i)} - \mathbf{C}^{(i,m_i)}\hat{\mathbf{x}}_{k|k-1}^{(i,m_i)}\right), \tag{7.7}$$

$$\hat{\mathbf{s}}_{p,k|k}^{(i,m_i)} = \hat{\mathbf{s}}_{p,k|k-1}^{(i,m_i)} - \mathbf{K}_k^{(i,m_i)}\mathbf{C}^{(i,m_i)}\hat{\mathbf{s}}_{p,k|k-1}^{(i,m_i)}, \tag{7.8}$$

$$\mathbf{P}_{k|k}^{\Sigma,(i,m_i)} = \mathbf{P}_{k|k-1}^{\Sigma,(i,m_i)} - \mathbf{K}_k^{(i,m_i)}\mathbf{C}^{(i,m_i)}\mathbf{P}_{k|k-1}^{\Sigma,(i,m_i)}. \tag{7.9}$$

- Before the mode probability update, the state error covariance needs to be extracted from the cumulative second moment

$$\mathbf{P}_{k|k-1}^{(i,m_i)} = \frac{1}{\alpha^{(i)}} \left(\mathbf{P}_{k|k-1}^{\Sigma,(i,m_i)} - \sum_{p=1}^{N_{\Theta,i}} \gamma_p^{(i)} \hat{\mathbf{s}}_{p,k|k-1}^{(i,m_i)} \left(\hat{\mathbf{s}}_{p,k|k-1}^{(i,m_i)} \right)^T \right). \tag{7.10}$$

Then likelihoods $\Lambda_k^{(i,m_i)}$ of the measurement being from the particular mode are computed and used to update the mode probabilities $\mu_k^{(i)}(m_i)$

$$\Lambda_k^{(i,m_i)} = p\left(\mathbf{y}_k \middle| \hat{\mathbf{x}}_{k|k-1}^{(i,m_i)}, \mathbf{P}_{k|k-1}^{(i,m_i)} \right), \tag{7.11}$$

$$\boldsymbol{\mu}_k^{(i)}(m_i) = \frac{1}{q} \Lambda_k^{(i,m_i)} \sum_{l_i \in \mathcal{M}^{(i)}} T_{k-1}^{(i)}(l_i, m_i) \mu_{k-1}^{(i)}(l_i), \tag{7.12}$$

where q is the normalizing factor, $\mu_k^{(i)}(m_i)$ is the m_i-th element of the vector $\boldsymbol{\mu}_k^{(i)}$, and $T_{k-1}^{(i)}(l_i, m_i)$ is the element in m_i-th row and l_i-th column of the matrix $\mathbf{T}_{k-1}^{(i)}$.

- The state estimate is updated as the linear combination of the mode state estimates weighted by the mode probabilities

$$\hat{\mathbf{x}}_{k|k}^{(i)} = \sum_{m_i \in \mathcal{M}^{(i)}} \hat{\mathbf{x}}_{k|k}^{(i,m_i)} \mu_k^{(i)}(m_i). \tag{7.13}$$

Exchange data. The posterior state estimates $\hat{\mathbf{x}}_{k|k}^{(i)}$ are exchanged with neighbors. Also, the local constraints needed for the global constraint evaluation are exchanged.

Time-update step.

- The transition matrix is updated based on the local and global constraints which determine the active mode set.

$$\text{if } c_k^{(i,m_i)} \wedge C_k^{(i,m_i)}$$

$$\mathbf{T}_k^{(i)}(\mathcal{M}^{(i)}, m_i) = \mathbf{T}_0^{(i)}(\mathcal{M}^{(i)}, m_i), \tag{7.14}$$

$$\text{else}$$

$$T_k^{(i)}(m_i, m_i) = 0, \tag{7.15}$$

$$\mathbf{T}_k^{(i)}(\mathcal{M}^{(i)} - \{m_i\}, m_i) = \mathbf{T}_0^{(i)}(\mathcal{M}^{(i)} - \{m_i\}, m_i) \times 10^{-3}, \tag{7.16}$$

$$\text{end} \tag{7.17}$$

$$\mathbf{T}_k^{(i)}(\mathcal{M}^{(i)}, m_i) = \frac{1}{\sum_{l_i \in \mathcal{M}^{(i)}} T_k^{(i)}(l_i, m_i)} \mathbf{T}_k^{(i)}(\mathcal{M}^{(i)}, m_i), \tag{7.18}$$

where $\mathbf{T}_k^{(i)}(\mathcal{M}^{(i)}, m_i)$ is the m_i-th column of matrix $\mathbf{T}_k^{(i)}$.

- The statistics of the modes are mixed according to the current transition matrix and mode probabilities

$$\pi_k^{(i)}(m_i | l_i) = \frac{T_k^{(i)}(l_i, m_i) \mu_k^{(i)}(l_i)}{\sum_{l_i \in \mathcal{M}^{(i)}} T_k^{(i)}(l_i, m_i) \mu_k^{(i)}(l_i)}, \tag{7.19}$$

$$\bar{\mathbf{x}}_{k|k}^{(i,m_i)} = \sum_{l_i \in \mathcal{M}^{(i)}} \hat{\mathbf{x}}_{k|k}^{(i,l_i)} \pi_k^{(i)}(m_i | l_i), \tag{7.20}$$

$$\bar{\mathbf{s}}_{p,k|k}^{(i,m_i)} = \sum_{l_i \in \mathcal{M}^{(i)}} \hat{\mathbf{s}}_{p,k|k}^{(i,l_i)} \pi_k^{(i)}(l_i | m_i), \tag{7.21}$$

$$\bar{\mathbf{P}}_{k|k}^{\Sigma,(i,m_i)} = \sum_{l_i \in \mathcal{M}^{(i)}} \pi_k^{(i)}(m_i | l_i) \Big[\mathbf{P}_{k|k}^{(i,l_i)} + \left(\hat{\mathbf{x}}_{k|k}^{(i,l_i)} - \bar{\mathbf{x}}_{k|k}^{(i,m_i)} \right) \left(\hat{\mathbf{x}}_{k|k}^{(i,l_i)} - \bar{\mathbf{x}}_{k|k}^{(i,m_i)} \right)^T \Big]. \tag{7.22}$$

- The statistics of the modes are propagated in time according to the models

$$\hat{\mathbf{A}}_{\mathrm{dc}}^{(i,m_i)} = \hat{\mathbf{A}}^{(i,m_i)} - \mathbf{S}_k^{(i,m_i)} \left(\alpha^{(i)}\mathbf{R}^{(i,m_i)}\right)^{-1} \mathbf{C}^{(i,m_i)}, \tag{7.23}$$

$$\mathbf{Q}_{k|k}^{\Sigma,(i,m_i)} = \mathbf{Q}_{k|k-1}^{\Sigma,(i,m_i)} - \mathbf{S}_k^{(i,m_i)} \left(\alpha^{(i)}\mathbf{R}^{(i,m_i)}\right)^{-1} \left(\mathbf{S}_k^{(i,m_i)}\right)^T, \tag{7.24}$$

$$\hat{\mathbf{x}}_{k+1|k}^{(i,m_i)} = \hat{\mathbf{A}}_{\mathrm{dc}}^{(i,m_i)} \bar{\mathbf{x}}_{k|k}^{(i,m_i)} + \hat{\mathbf{B}}^{(i,m_i)} \mathbf{u}_k^{(i)} + \sum_{j\in\mathcal{G}^{(i)}} \hat{\rho}_j^{(i)} \mathbf{G}_j^{(i,m_i)} \hat{\mathbf{x}}_{k|k}^{(j)} + \\ + \mathbf{S}_k^{(i,m_i)} \left(\alpha^{(i)}\mathbf{R}^{(i,m_i)}\right)^{-1} \mathbf{y}_k^{(i)}, \tag{7.25}$$

$$\hat{\mathbf{s}}_{p,k+1|k}^{(i,m_i)} = \hat{\mathbf{A}}_{\mathrm{dc}}^{(i,m_i)} \bar{\mathbf{s}}_{p,k|k}^{(i,m_i)} - \mathbf{b}_{p,k}^{(i,m_i)}, \tag{7.26}$$

$$\mathbf{P}_{k+1|k}^{\Sigma,(i,m_i)} = \hat{\mathbf{A}}_{\mathrm{dc}}^{(i,m_i)} \bar{\mathbf{P}}_{k|k}^{\Sigma,(i,m_i)} \left(\hat{\mathbf{A}}_{\mathrm{dc}}^{(i,m_i)}\right)^T + \mathbf{Q}_{k|k}^{\Sigma,(i,m_i)}. \tag{7.27}$$

- The state estimate is updated as the linear combination of the mode state estimates weighted by the mode probabilities

$$\hat{\mathbf{x}}_{k+1|k}^{(i)} = \sum_{m_i\in\mathcal{M}^{(i)}} \hat{\mathbf{x}}_{k+1|k}^{(i,m_i)} \mu_k^{(i)}(m_i). \tag{7.28}$$

The LIMM-XDKF describes the state and mode estimation. Nevertheless, the current mode decision needs to be made for FDD applications. The simplest decision-making algorithm is to select the mode with the highest mode probability

$$d_k^{(i)} = \max_{m_i} \mu_k^{(i)}(m_i). \tag{7.29}$$

This decision can also be used to reconfigure a model-based controller such as the MPC, for example, by switching to an alternative model and using a corresponding state estimate. The mode decision allows the MPC to use the model with the most accurate system representation, improving prediction accuracy and control performance.

7.2 Application to buildings

This section describes how the proposed algorithm can provide state estimates and mode decisions to the DMPC, which controls a building heating system. First, the building model is introduced. Then the model is used with the LIMM-XDKF, which provides the information to the local controller.

7.2.1 Building floor model

The building envelope includes the parts of the building where most of the thermal capacity is included (walls, floors, roofs). It creates a border between the outdoor and indoor environments. The indoor environment is further split into zones. Interconnected zone models create the model of a floor/building. The interactions between zones are depicted in Fig. 7.1. The zones interact by heat conduction (through walls) and heat convection (open corridors – dashed line).

The zone temperatures are controlled using decentralized DMPC controllers designed using linearized models of the zones. Each zone has its fan coil unit (FCU) with heat flow output manipulated by the controller and a temperature setpoint tracked by the controller.

The zones and their interactions are modeled using high-fidelity models designed in the Modelica environment. The model consists of the following components

- zone models with completely mixed air volume,
- floor and partition walls definitions
- fan coil unit with manipulated heat flow setpoint
- inputs for disturbances as inner heat gains
- fixed ventilation with the air of known temperature modeled as a source of constant pressure (supply duct) with a damper (VAV) and a sink of constant pressure with pressure resistance in the duct
- boundary walls are split between the zone models, and each zone contains approximately half of the wall
- external wall, where a concept of equivalent ambient air temperature is used to represent the cumulative impact of all heat sources.

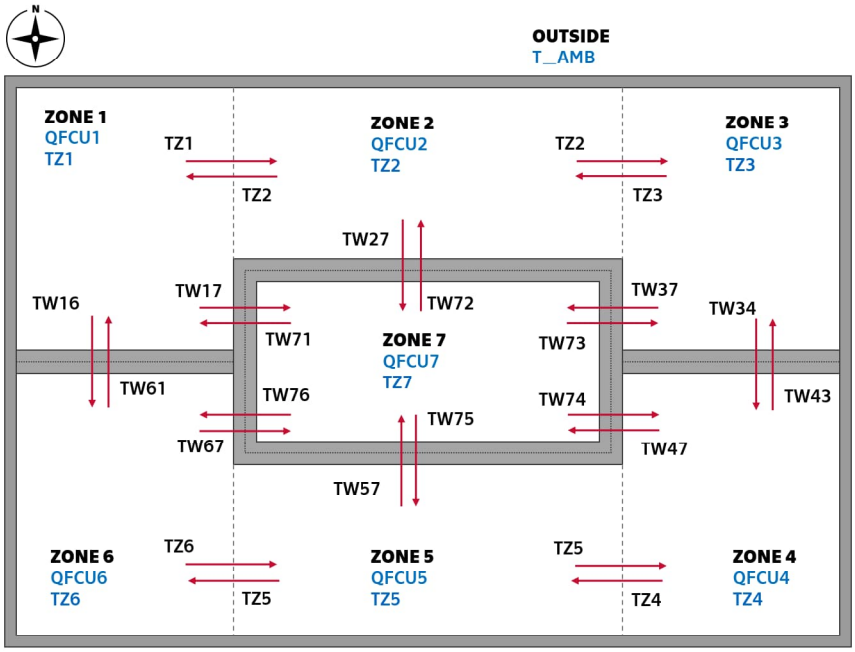

Figure 7.1. Building floor scheme with connections between zones.

In each zone, the air temperature (TZ) is measured. The inputs to the zone include a heat setpoint to the zone heat source (QFCU), a vector of air temperatures in adjacent zones, connected by open doors/openings (TZ), a vector of partial wall inner temperatures in adjacent zones (TW), and weather disturbances. The weather disturbances are calculated for each surface by the weather disturbance model, and they include a vector of the solar short-wave radiation coming through the windows (JWin), vectors of equivalent window temperatures (TEqWin) and wall temperatures (TEqWal). Both equivalent air temperatures are computed according to German VD 6007 standard [68], where solar irradiance with dry bulb ambient air temperature and longwave radiation can be incorporated into an equivalent air temperature. The solar radiation model evaluates the incidence solar radiation on every surface of the building based on the geographic location; direct and global solar radiation forecast; black body sky temperature (or cloudiness); azimuth and tilt of the surface, and date and time. Each window's total transmitted radiation is computed based on the computed incidence angle.

The model is described only verbally since the detailed description is extensive, and the building modeling is beyond the scope of the thesis.

Simplified model. The nonlinear zone models in Modelica are too complicated for control and FDD applications. They are also too specific, so it is difficult to reuse them in different projects.

Therefore, the zone models are simplified. First, each model is linearized at a fixed operating point and discretized using Euler's method with 60 seconds sampling period. The result is a set of 7 interconnected linear discrete-time models that approximately describe the building. The zone model has 14 states if the zone is located in a corner and 13 otherwise.

The state estimation using the approximate linearized model can be more accurate by using the uncertainty of model parameters. However, information about the uncertainty of parameters is lacking. Fortunately, the entire information about the parameter uncertainty statistics is not required for the LIMM-XDKF, but only the list of the parameters with expected values is needed. The list is created by perturbing each model parameter and observing if the model eigenvalues change significantly. This way, the list of parameters with a high impact on model dynamics is obtained. Their expected values are the values of the nominal model parameters before perturbation. For the LIMM-XDKF purpose, the list of the uncertain parameters is extended by the parameters that define the error of the neighbors' state estimates.

Next, the linearized zone models with uncertain parameters are reduced to the third order. The reduced model captures the main dynamics of the zone, which means it can also approximately describe zones similar to the one originally modeled in Modelica. The discrete-time zone models are reduced with the balanced truncation method using Matlab and MORLAB toolbox [69]. The parameterized model order reduction methods are discussed in [70].

Models of zone modes. The LIMM-XDKF for each zone uses predefined models for the normal behavior mode and additional modes that capture local and global faults. Abrupt heat loss in a zone is selected as an example of a local fault event. An example of such an event can be opening the window in winter. This mode is included in the mode set for all seven zones. The heat loss is modeled as an imbalance created by an unknown input to the zone heat balance equation. To be able to estimate the amplitude of the heat loss, the nominal model is augmented by an additional state and the difference equation. The heat loss state connects to the nominal model with the same gain as the FCU heat flow

$$\begin{bmatrix} \mathbf{r}_{k+1} \\ h_{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_r & b_{r,1} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r}_k \\ h_k \end{bmatrix} + \begin{bmatrix} b_{r,1} & \mathbf{B}'_{r,1} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{Q}_{\mathrm{FCU},k} \\ \mathbf{u}'_{1,k} \end{bmatrix} + \mathbf{w}_k, \tag{7.30}$$

where h_k is the heat loss state, $\mathbf{r}_k$ is the reduced order state vector of the nominal model, and $\mathbf{B}'_{r,1}$ is the reduced order input matrix without the column that corresponds to the FCU heat flow input $\dot{Q}_{\mathrm{FCU},k}$.

The global fault event is represented by the shadows restricting solar gains calculated by the forecast or unexpected clouds not included in the forecast. This mode is included in all controllers except the one for zone 7, which has no window. For demonstration purposes of this example, the mode representing this scenario is modeled by expecting half of the calculated solar gains

$$\mathbf{r}_{k+1} = \mathbf{A}_r \mathbf{r}_k + \begin{bmatrix} 0.5\, b_{r,2} & \mathbf{B}'_{r,2} \end{bmatrix} \begin{bmatrix} J_{\mathrm{Win},k} \\ \mathbf{u}'_{2,k} \end{bmatrix} + \mathbf{v}_{r,k}, \tag{7.31}$$

where $\mathbf{B}'_{r,2}$ is the reduced order input matrix without the column that corresponds to the solar gain input $J_{\mathrm{Win},k}$.

7.2.2 Zone state estimation and control

The performance of the LIMM-XDKF is tested in combination with the DMPC controller, which generates the local control action to follow the temperature setpoint in the zone. The local communication scheme between the LIMM-XDKF and controller is shown in Fig. 7.2. The LIMM-XDKF provides the state estimate information and mode decision to the DMPC. The DMPC algorithm is based on the optimal condition decomposition [71]. The particular design of the DMPC is beyond the scope of this thesis. For the purpose of this thesis, it is sufficient to describe it as a generator of the local heat flow signal QFCU. The local weather disturbances JWin, TEqWin, TEqWal are computed based on the weather disturbance model and weather forecast.

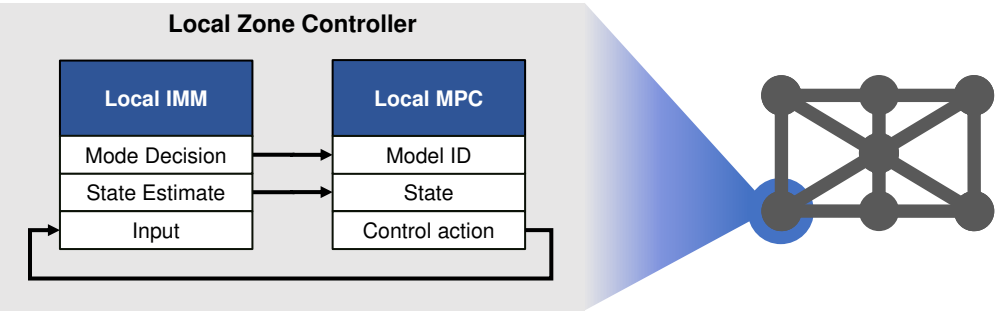

Figure 7.2. Communication between the local IMM-XDKF and the local MPC.

The LIMM-XDKF detects and diagnoses faults based on predefined models corresponding to individual modes. The nominal mode in the LIMM-XDKF is the local model with nominal behavior. The second mode is the fault mode, which has an additional estimated state representing the unknown disturbance of the heat balance equation. This mode is denoted as the heat loss or HL mode in the next text. The third mode is the fault mode with reduced solar gains on the input, which describes the scenario with a shadow in front of the zone window. This mode is denoted as the shadow or SH mode in the next text. The initial transition probabilities between modes are defined using transition matrices

$$\begin{aligned} \mathbf{T}_0^{(i)} &= \begin{bmatrix} 0.99 & 0.005 & 0.005 \\ 0.005 & 0.995 & 0 \\ 0 & 0.005 & 0.995 \end{bmatrix}, \text{ for } i \in \{1, 2, \ldots, 6\} \\ \mathbf{T}_0^{(7)} &= \begin{bmatrix} 0.995 & 0.005 \\ 0.005 & 0.995 \end{bmatrix}, \end{aligned} \tag{7.32}$$

where the order of modes is the nominal mode, the heat loss mode, and the shadow mode. Zone 7 does not have shadow mode since the mode is not feasible due to the zone location.

Notice that the shadow mode would have an identical likelihood with the nominal mode during zero solar gain in a zone. However, the shadow mode cannot occur if there is zero solar gain. This scenario is described by the local constraint that deactivates the shadow mode when the solar gain is lower than 4 kW.

Furthermore, the shadow mode activation is constrained by a global constraint. The building is virtually located in the northern hemisphere, with zone 2 pointing north.

Therefore, the shadow mode is mostly relevant in zones 4, 5, and 6. The global constraints are the conjugates of local constraints and neighbors' constraints

$$\begin{aligned} C_k^{(4,3)} &= c_k^{(4,3)} \wedge c_k^{(5,3)}, \\ C_k^{(5,3)} &= c_k^{(5,3)} \wedge c_k^{(4,3)} \wedge c_k^{(6,3)}, \\ C_k^{(6,3)} &= c_k^{(6,3)} \wedge c_k^{(5,3)}, \end{aligned} \tag{7.33}$$

where the local constraint is satisfied when the shadow mode likelihood is higher than the nominal mode likelihood

$$c_k^{(i,3)} = \Lambda_k^{(i,3)} > \Lambda_k^{(i,1)}. \tag{7.34}$$

The FDD capability of the LIMM-XDKF is demonstrated by simulating the heat loss and shadows. The zone temperatures and solar gains are shown in Fig. 7.3. It can be seen that the local controllers follow the constant temperature setpoints during simulations.

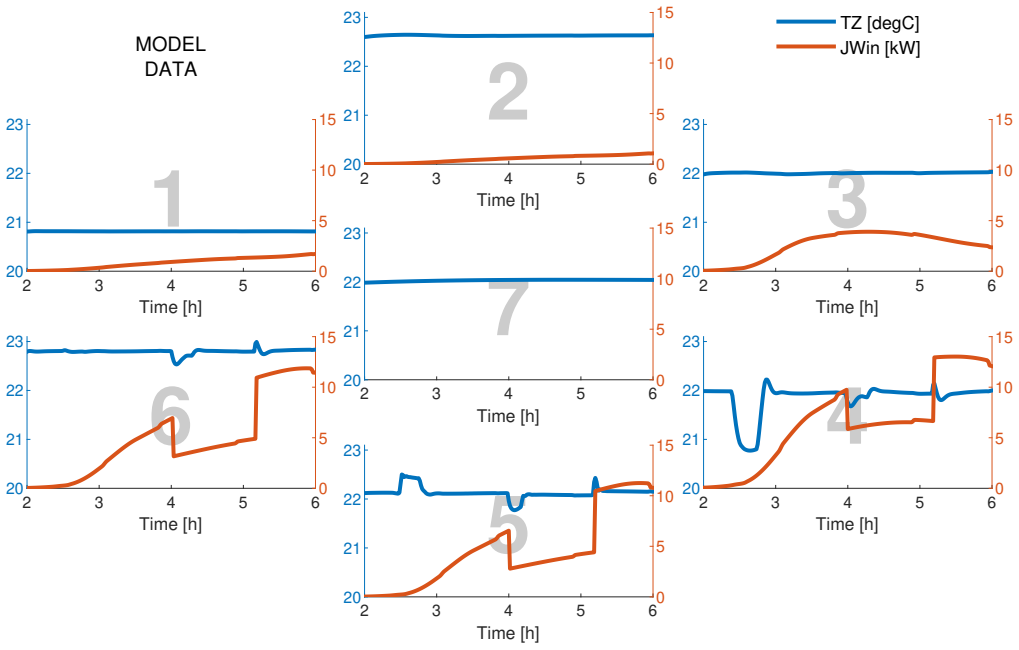

Figure 7.3. Zone temperatures and solar gains. The number in the plot backgrounds denotes the zone number. The zone temperature is denoted TZ and the solar gain is denoted as JWin. The axis scale color corresponds to the signal color. The x-axis shows the simulation time.

The heat loss in zone 4 demonstrates the local fault detection. The simulated scenario represents a real situation when a window is opened in winter. The simulated and detected faults in particular zones are depicted in Fig. 7.4. The heat loss is detected several minutes after the fault occurrence. Since the fault is modeled as unknown input, the algorithm can also estimate the amplitude of heat loss. The amplitude estimation is depicted in Fig. 7.5, where it can be seen that 5 kW heat loss is estimated with approximately 10 % (= 500 W) accuracy. Recovery back to the nominal behavior is again detected within several minutes, which is a very good result in building systems with relatively slow dynamics. The heat loss can also be seen in Fig. 7.3, where it causes a one-degree drop in the zone temperature. After the detection, the information about

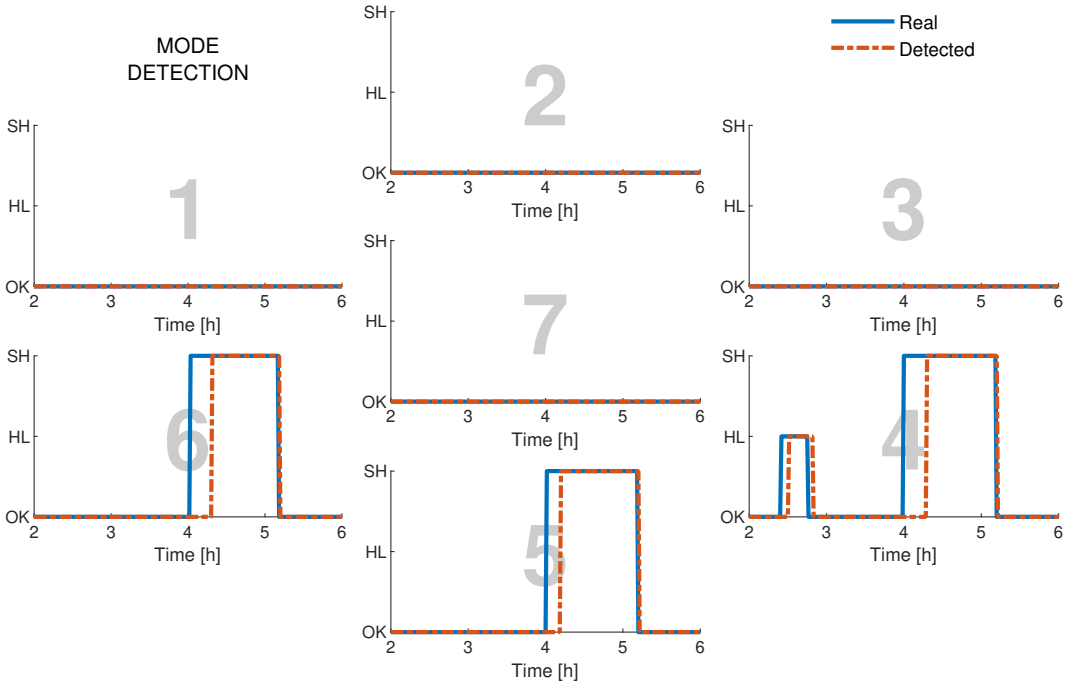

Figure 7.4. Mode detection. The number in the plot backgrounds denotes the zone number. The y-axis denotes the modes – nominal mode as OK, heat loss mode as HL, and shadow mode as SH. When the blue line is not visible at a given time, it is identical to the red line.

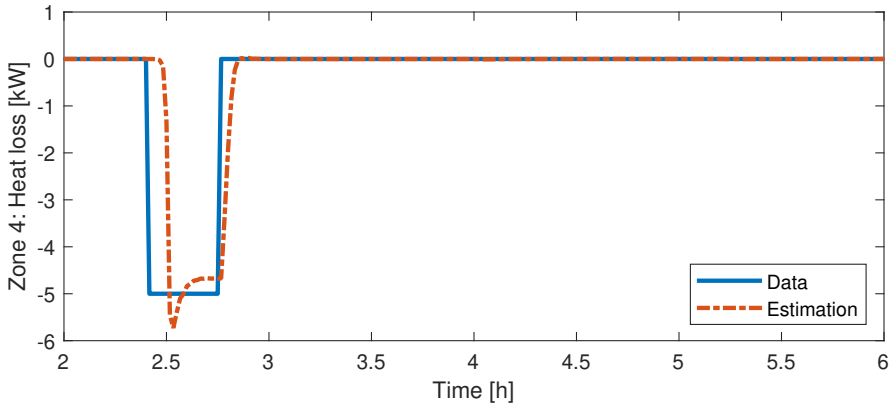

Figure 7.5. Heat loss estimation in zone 4. The figure shows the simulated heat loss (blue) and the state estimated by the LIMM-XDKF that corresponds to heat loss (red).

the fault occurrence is sent to the DMPC controller, which could take some corrective actions. For example, the zone setpoint or the zone thermal input is decreased to prevent wasting energy when the opened window is detected in winter. Such actions are not implemented in this example, which can be seen in Fig. 7.3, where the zone 5 temperature is increased to compensate for the heat loss in zone 4.

Global fault detection is demonstrated by reducing the real solar gains compared to the solar gains predicted by the weather forecast. The simulated scenario represents a real situation where unexpected clouds restrict solar radiation. The real solar radiation is shown in Fig. 7.3. The solar radiation starts to increase gradually, but at the simulation time equal to 4 hours, it is reduced for more than one hour. It is crucial to note that the LIMM-XDKF does not have information about this drop in solar radiation. First, notice that the shadow mode starts to be activated immediately after the solar gain

crosses the 4 kW threshold, which can be seen in Fig. 7.6. However, it is only activated for a short time since the global constraint is not satisfied. Similar likelihoods between the nominal and shadow mode cause fast activation and deactivation at low solar gains. The shadow mode is activated for longer when the real solar radiation drops. After that, the nominal mode probability drops, whereas the shadow mode becomes the most probable. Eventually, the global fault is detected locally in each zone where the fault occurred, which can be seen in Fig 7.4. The recovery to the nominal mode is fast after the fault is removed. The spikes of false positive shadow mode detection in the zones 4, 5, and 6 in Fig. 7.6 show that the FDD performance needs to be improved by adding a hysteresis for decision making.

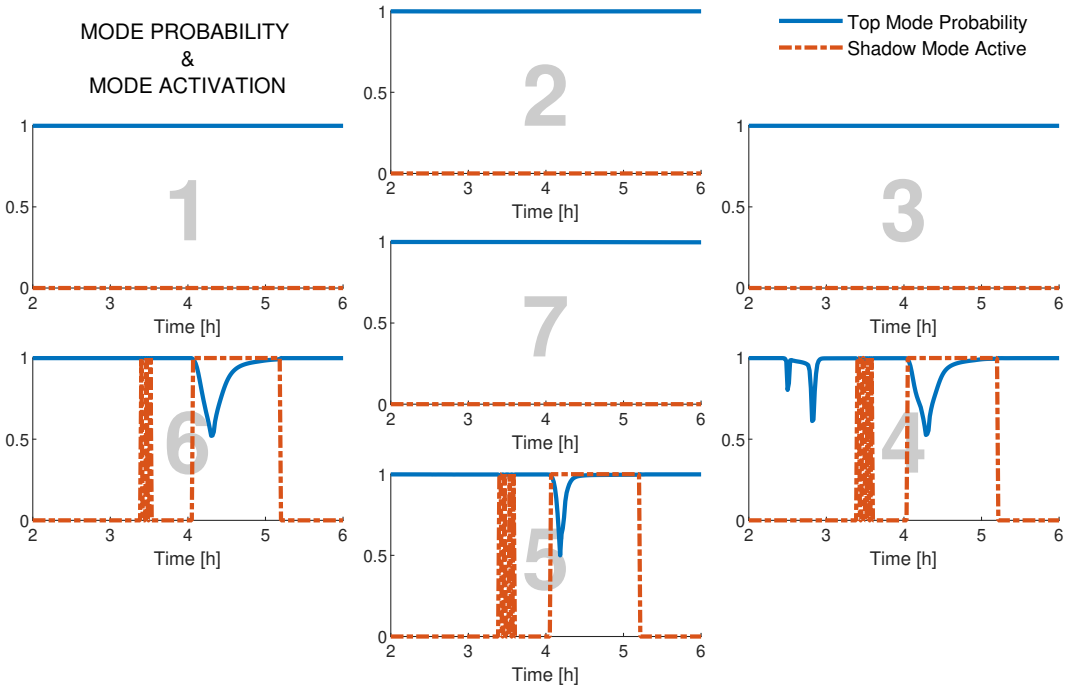

Figure 7.6. Mode probability and mode activation. The number in the plot backgrounds denotes the zone number. The figure shows the highest mode probability among active modes and shadow mode activation. The nominal mode and the heat loss mode are always active.

Chapter 8
Conclusion

This thesis brings new results to the desensitized filtering theory. The core result is the stochastic approach to sensitivity definition, which allows the derivation of the exact desensitized Kalman filter. Thanks to the convenient formulation of the exact desensitized Kalman filter, many interesting algorithms and results can be further derived. The main results mentioned in this thesis are summarized in the following sections.

8.1 Desensitized state estimation

The main contributions are in the field of desensitized filtering. The stochastic approach to reducing sensitivity leading to the exact desensitized Kalman filter (XDKF) is the core result that enables derivation of the following useful variants of the XDKF:

- Algorithm with zero gain sensitivity assumption resulting in reduced complexity has comparable performance to the XDKF, so it is useful for real applications.
- Algorithm with separate data-update and time-update steps is useful when both prior and posterior state estimates are required.
- Parametrized steady-state algorithm is ideal for applications with limited computing power, such as implementation in embedded devices.
- Algorithms with normalized objectives and adaptive weights improve the performance of the XDKF and make the weight tuning intuitive.

The formulation of the XDKF allowed us to analyze the stability and formulate the stability conditions related to weights. Also, the extended XDKF was derived for nonlinear systems.

8.2 Desensitized fault detection

A correct model is essential to model-based fault detection and diagnosis. Therefore, using simplified models can be difficult. This thesis showed that the interacting multiple model method with desensitized filtering is a promising fault detection and diagnosis method when using simplified models. Two algorithms were described. One is based on the original desensitized Kalman filter, and the other is based on the exact desensitized Kalman filter. Both algorithms proved the ability to detect and diagnose separate and simultaneously occurring faults. The proposed algorithms using simplified models with explicit uncertainty provide performance comparable to the conventional algorithms with accurate models.

8.3 Distributed state estimation and fault detection

In this thesis, it is shown how the desensitized approach can be valuable for distributed state estimation. The result is the local exact desensitized Kalman filter (LXDKF), a

distributed method that is scalable, reusable, accurate, and has a low communication burden. The LXDKF is suitable for large-scale networked systems that can be partitioned into interconnected subsystems, ensuring scalability. The uncertainty of neighbor estimates is considered in local estimates using the desensitized filtering approach without sharing it. Therefore, the state estimation accuracy is improved compared to the distributed methods without covariance sharing, whereas the network communication load remains low. Compared to the distributed methods with neighbor covariance sharing, the communication transfer is lower since the uncertainty is considered locally without sharing. The local state estimation is robust to parameter uncertainty, and its accuracy can be improved when there is a mismatch between real and model parameters. Consequently, simplified local models can be used, creating an opportunity to reuse the algorithm for similar systems. A distributed interacting multiple model filter for large-scale networks which use the approach of LXDKF is also introduced. This method, named LIMM-XDKF, is based on and has properties of the LXDKF. Another extension is the distributed model-based fault detection and diagnosis method based on the LIMM-XDKF. In case-study simulations, the method detected local or global faults while providing the information to the distributed model-predictive controller used for the heating control.

8.4 Future work

There is still much work to be done in the desensitized filtering theory. In the case of theoretical work, the most important area for further research is the stability analysis. Also, the benefits of stochastic sensitivity definition in more advanced state estimation methods need to be studied. For example, our preliminary results show that desensitized approach might be useful for the moving horizon estimation.

For real applications, numerically stable algorithms are required. A numerically robust formulation of the exact desensitized Kalman filter is presented but needs to be fine tuned. Last but not least, the algorithms should be applied to various problems because the true advantages and disadvantages arise only then.

Author's publications

Journal papers

[J1] Tabaček, Jaroslav, and Vladimír Havlena. (Authorship: 90%) Reduction of prediction error sensitivity to parameters in Kalman filter. *Journal of the Franklin Institute.* Elsevier BV, 2022, Vol. 359, No. 3, pp. 1303–1326. Available from DOI 10.1016/j.jfranklin.2021.12.019. Citations: 3 (Scopus), 2 (WoS)

[J2] Tabaček, Jaroslav, and Vladimír Havlena. (Authorship: 90%) Distributed state estimation and fault diagnosis using reduced sensitivity to neighbor estimates with application to building control. *Journal of the Franklin Institute.* Elsevier BV, 2022. Available from DOI 10.1016/j.jfranklin.2022.10.017.

Conference papers in WoS

[C1] Žáčeková, Eva, Matej Pčolka, Jaroslav Tabaček, et al. (Authorship: 7%) Identification and energy efficient control for a building: Getting inspired by MPC. In: *2015 American Control Conference (ACC).* IEEE, 2015, pp. 1671-1676, Available from DOI 10.1109/ACC.2015.7170973. Citations: 10 (Scopus), 8 (WoS)

[C2] Tabaček, Jaroslav, and Vladimír Havlena. (Authorship: 90%) Desensitized Filtering for Systems with Uncertain Parameters and Noise Correlation. In: *2018 26th Mediterranean Conference on Control and Automation (MED).* IEEE, 2018. Available from DOI 10.1109/MED.2018.8443049. Citations: 2 (Scopus)

[C3] Tabaček, Jaroslav, and Vladimír Havlena. (Authorship: 90%) Schmidt-Kalman Filters for Systems with Uncertain Parameters and Asynchronous Sampling. In: *2018 18th International Conference on Control, Automation and Systems (ICCAS).* Daegwallyeong: IEEE, 2018. pp. 357-362. Citations: 2 (Scopus)

[C4] Tabaček, Jaroslav, and Vladimír Havlena. (Authorship: 90%) Desensitized Extended Kalman Filter with Stochastic Approach to Sensitivity Reduction and Adaptive Weights. In: *2022 25th International Conference on Information Fusion (FUSION).* IEEE, 2022. Available from DOI 10.23919/fusion49751.2022.9841381.

Other publications

[O1] Pachner, Daniel, Jaroslav Tabaček, and Vladimír Havlena. (Authorship: 30%) Control-oriented models for turbocharged engines In: *Turbochargers and turbocharging: Advancements, applications and research.* Hauppauge NY: Nova Science Publisher, Inc., 2017. p. 435-452. ISBN 978-1-5361-2239-8.

[O2] Tabaček, Jaroslav. *SESUP: Testing tool for state estimation algorithms for systems with uncertain parameters* [Mendeley Data, V2]. Available from DOI 10.17632/69529JS9VR.2.

[O3] Tabaček, Jaroslav, and Vladimír Havlena. (Authorship: 90%) Fault Detection for Buildings Using Uncertain Parameters and Interacting Multiple-Model Method. In: Elena Zattoni, Silvio Simani, and Giuseppe Conte, eds. *15th European Workshop on Advanced Control and Diagnosis (ACD 2019).* Springer International Publishing, 2022. Lecture Notes in Control and Information Sciences - Proceedings. Available from DOI 10.1007/978-3-030-85318-1_77.

References

[1] Kalman, Rudolph Emil. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME–Journal of Basic Engineering*. 1960, Vol. 82, No. Series D, pp. 35–45.

[2] Schmidt, S. F.. The Kalman filter - Its recognition and development for aerospace applications. *Journal of Guidance, Control, and Dynamics*. American Institute of Aeronautics and Astronautics (AIAA), jan, 1981, Vol. 4, No. 1, pp. 4–7. Available from DOI 10.2514/3.19713.

[3] Jazwinski, Andrew H.. *Stochastic Processes and Filtering Theory*. Burlington: Academic Press, 1970. ISBN 9780123815507;0123815509.

[4] Schmidt, Stanley F.. Applications of State Space Methods to Navigation Problems. *in C. T. Leondes, Editor, Advanced Control Systems*. 1966, Vol. 3, pp. 293–340.

[5] Woodbury, Drew, and John Junkins. On the Consider Kalman Filter. In: *AIAA Guidance, Navigation, and Control Conference*. 2010.

[6] Woodbury, Drew Patton. *Accounting for parameter uncertainty in reduced-order static and dynamic systems*. Texas A&M University, 2011. Ph.D. Thesis.

[7] Chen, Jie, Ron J. Patton, and Hong-Yue Zhang. Design of unknown input observers and robust fault detection filters. *International Journal of Control*. Jan., 1996, Vol. 63, No. 1, pp. 85–105. Available from DOI 10.1080/00207179608921833.

[8] Chen, Jie, R. J. Patton, and Jie Chen. *Robust Model-Based Fault Diagnosis for Dynamic Systems*. Springer US, 1999. ISBN 0792384113.

[9] Petersen, J.R., and D.C. McFarlane. Robust state estimation for uncertain systems. In: *Proceedings of the 30th IEEE Conference on Decision and Control*. IEEE, 1991. Available from DOI 10.1109/cdc.1991.261827.

[10] Xie, Lihua, Carlos E. de Souza, and Minyue Fu. H_∞ estimation for discrete-time linear uncertain systems. *International Journal of Robust and Nonlinear Control*. Wiley, apr, 1991, Vol. 1, No. 2, pp. 111–123. Available from DOI 10.1002/rnc.4590010206.

[11] Shaked, U., and Y. Theodor. H_∞-optimal estimation: a tutorial. In: *Proceedings of the 31st IEEE Conference on Decision and Control*. IEEE, 1992. Available from DOI 10.1109/cdc.1992.371384.

[12] Shaked, U., Lihua Xie, and Yeng Chai Soh. New approaches to robust minimum variance filter design. *IEEE Transactions on Signal Processing*. Institute of Electrical and Electronics Engineers (IEEE), 2001, Vol. 49, No. 11, pp. 2620–2629. Available from DOI 10.1109/78.960408.

[13] Geromel, J. C., M. C. de Oliveira, and J. Bernussou. Robust Filtering of Discrete-Time Linear Systems with Parameter Dependent Lyapunov Functions. *SIAM Journal on Control and Optimization*. Society for Industrial & Applied

Mathematics (SIAM), jan, 2002, Vol. 41, No. 3, pp. 700–711. Available from DOI 10.1137/s0363012999366308.

[14] Xie, Lihua, Lilei Lu, David Zhang, and Huanshui Zhang. Improved robust H2 and H∞ filtering for uncertain discrete-time systems. *Automatica*. Elsevier BV, may, 2004, Vol. 40, No. 5, pp. 873–880. Available from DOI 10.1016/j.automatica.2004.01.003.

[15] Souto, Rodrigo Fontes, and Joao Yoshiyuki Ishihara. Robust Kalman filter for discrete-time systems with correlated noises. In: *2008 16th Mediterranean Conference on Control and Automation*. IEEE, 2008. Available from DOI 10.1109/med.2008.4602037.

[16] Lewis, Frank L., Lihua Xie, and Dan Popa. *Optimal and robust estimation: with an introduction to stochastic control theory*. 2nd ed. Boca Raton: CRC, 2008. ISBN 0849390087;9780849390081.

[17] Karlgaard, Christopher D., and Haijun Shen. Desensitised Kalman filtering. *IET Radar, Sonar & Navigation*. Institution of Engineering and Technology (IET), Jan., 2013, Vol. 7, No. 1, pp. 2–9. Available from DOI 10.1049/iet-rsn.2012.0075.

[18] Seywald, Hans, and Renjith R. Kumar. Desensitized Optimal Trajectories. In: *Proceedings of the Sixth Annual AAS/AIAA Space Flight Mechanics conference*. Austin, TX, 1996. pp. 96-107.

[19] Seywald, Hans, and Kevin Seywald. Desensitized Optimal Control. In: *AIAA Scitech 2019 Forum*. American Institute of Aeronautics and Astronautics, 2019. Available from DOI 10.2514/6.2019-0651.

[20] Karlgaard, Christopher D.. *Desensitized Optimal Filtering and Sensor Fusion Toolkit*. Available from `https://ntrs.nasa.gov/citations/20160005425`.

[21] Lou, Taishan. Desensitized Kalman Filtering with Analytical Gain. *CoRR*. 2015, Vol. abs/1504.04916. Available from `http://arxiv.org/abs/1504.04916`.

[22] Chen, Nanhua, Liangyu Zhao, Tai-shan Lou, and Chuanjun Li. Adaptive fast desensitized ensemble Kalman filter for uncertain systems. *Signal Processing*. Elsevier BV, sep, 2022, pp. 108767. Available from DOI 10.1016/j.sigpro.2022.108767.

[23] Li, Shuang, Xiuqiang Jiang, and Yufei Liu. High-precision Mars Entry Integrated Navigation Under Large Uncertainties. *Journal of Navigation*. Cambridge University Press (CUP), nov, 2013, Vol. 67, No. 2, pp. 327–342. Available from DOI 10.1017/s0373463313000738.

[24] Lou, Taishan, and Liangyu Zhao. Robust Mars atmospheric entry integrated navigation based on parameter sensitivity. *Acta Astronautica*. Elsevier BV, feb, 2016, Vol. 119, pp. 60–70. Available from DOI 10.1016/j.actaastro.2015.11.006.

[25] Wang, Liansheng, and Yuanqing Xia. Mars Entry Navigation With Uncertain Parameters Based on Desensitized Extended Kalman Filter. *IEEE Transactions on Industrial Informatics*. Institute of Electrical and Electronics Engineers (IEEE), oct, 2015, Vol. 11, No. 5, pp. 998–1005. Available from DOI 10.1109/tii.2015.2463763.

[26] Shen, Haijun, and Christopher D. Karlgaard. Desensitized Unscented Kalman Filter About Uncertain Model Parameters. In: *2012 ION International Technical Meeting*. Newport Beach, CA, 2012.

[27] Karlgaard, Christopher D., and Haijun Shen. Robust state estimation using desensitized Divided Difference Filter. *ISA Transactions*. Elsevier BV, sep, 2013, Vol. 52, No. 5, pp. 629–637. Available from DOI 10.1016/j.isatra.2013.04.009.

[28] Lou, Tai-Shan, Lei Wang, Housheng Su, Mao-Wen Nie, Ning Yang, and Yanfeng Wang. Desensitized cubature Kalman filter with uncertain parameters. *Journal of the Franklin Institute*. Elsevier BV, Dec., 2017, Vol. 354, No. 18, pp. 8358–8373. Available from DOI 10.1016/j.jfranklin.2017.09.004.

[29] Yang, Xiao-Liang, Guo-Rong Liu, Nan-Hua Chen, and Tai-Shan Lou. Desensitized Ensemble Kalman Filtering for Induction Motor Estimation. *IEEE Access*. Institute of Electrical and Electronics Engineers (IEEE), 2019, Vol. 7, pp. 78029–78036. Available from DOI 10.1109/access.2019.2921971.

[30] Patton, Ron J, Paul M Frank, and Robert N Clark. *Issues of Fault Diagnosis for Dynamic Systems*. Springer Science & Business Media, 2000.

[31] Isermann, R., and P. Ballé. Trends in the application of model-based fault detection and diagnosis of technical processes. *Control Engineering Practice*. Elsevier BV, may, 1997, Vol. 5, No. 5, pp. 709–719. Available from DOI 10.1016/s0967-0661(97)00053-1.

[32] Venkatasubramanian, Venkat, Raghunathan Rengaswamy, Surya N. Kavuri, and Kewen Yin. A review of process fault detection and diagnosis: Part III: Process history based methods. *Computers & Chemical Engineering*. Elsevier BV, mar, 2003, Vol. 27, No. 3, pp. 327–346. Available from DOI 10.1016/s0098-1354(02)00162-x.

[33] Venkatasubramanian, Venkat, Raghunathan Rengaswamy, and Surya N Kavuri. A review of process fault detection and diagnosis: Part II: Qualitative models and search strategies. *Computers & Chemical Engineering*. Elsevier BV, mar, 2003, Vol. 27, No. 3, pp. 313–326. Available from DOI 10.1016/s0098-1354(02)00161-8.

[34] Venkatasubramanian, Venkat, Raghunathan Rengaswamy, Kewen Yin, and Surya N. Kavuri. A review of process fault detection and diagnosis: Part I: Quantitative model-based methods. *Computers & Chemical Engineering*. Elsevier BV, mar, 2003, Vol. 27, No. 3, pp. 293–311. Available from DOI 10.1016/s0098-1354(02)00160-6.

[35] Gertler, Janos. *Fault Detection and Diagnosis in Engineering Systems*. MARCEL DEKKER INC, 1998. ISBN 0824794273.

[36] Isermann, Rolf. Model-based fault-detection and diagnosis – status and applications. *Annual Reviews in Control*. Elsevier BV, jan, 2005, Vol. 29, No. 1, pp. 71–85. Available from DOI 10.1016/j.arcontrol.2004.12.002.

[37] Hwang, Inseok, Sungwan Kim, Youdan Kim, and Chze Eng Seah. A Survey of Fault Detection, Isolation, and Reconfiguration Methods. *IEEE Transactions on Control Systems Technology*. Institute of Electrical and Electronics Engineers (IEEE), may, 2010, Vol. 18, No. 3, pp. 636–653. Available from DOI 10.1109/tcst.2009.2026285.

[38] Costa, O. L. V., M. D. Fragoso, and R. P. Marques. *Discrete-Time Markov Jump Linear Systems*. Springer-Verlag GmbH, 2006. ISBN 9781846280825.

[39] Maybeck, Peter S.. *Stochastic Models, Estimation, and Control*. Academic Press, 1982. ISBN 012480702X.

[40] Blom, H.A.P., and Y. Bar-Shalom. The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE Transactions on Automatic Control*. Institute of Electrical and Electronics Engineers (IEEE), 1988, Vol. 33, No. 8, pp. 780–783. Available from DOI 10.1109/9.1299.

[41] Mazor, E., A. Averbuch, Y. Bar-Shalom, and J. Dayan. Interacting multiple model methods in target tracking: a survey. *IEEE Transactions on Aerospace and Electronic Systems*. Institute of Electrical and Electronics Engineers (IEEE), 1998, Vol. 34, No. 1, pp. 103–123. Available from DOI 10.1109/7.640267.

[42] Efe, Murat, and Derek P. Atherton. The IMM Approach to the Fault Detection Problem. *IFAC Proceedings Volumes*. Elsevier BV, jul, 1997, Vol. 30, No. 11, pp. 603–608. Available from DOI 10.1016/s1474-6670(17)42911-9.

[43] Basseville, Michèle, and Igor V. Nikiforov. *Detection of Abrupt Changes: Theory and Application*. 1. ed. Englewood Cliffs: Prentice Hall, 1993. ISBN 9780131267800;0131267809.

[44] Olfati-Saber, Reza. Kalman-Consensus Filter : Optimality, stability, and performance. In: *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*. IEEE, 2009. Available from DOI 10.1109/cdc.2009.5399678.

[45] Cattivelli, Federico S., and Ali H. Sayed. Diffusion Strategies for Distributed Kalman Filtering and Smoothing. *IEEE Transactions on Automatic Control*. Institute of Electrical and Electronics Engineers (IEEE), sep, 2010, Vol. 55, No. 9, pp. 2069–2084. Available from DOI 10.1109/tac.2010.2042987.

[46] Marelli, Damián, Mohsen Zamani, Minyue Fu, and Brett Ninness. Distributed Kalman filter in a network of linear systems. *Systems & Control Letters*. Elsevier BV, jun, 2018, Vol. 116, pp. 71–77. Available from DOI 10.1016/j.sysconle.2018.04.005.

[47] Roshany-Yamchi, Samira, Marcin Cychowski, Rudy R. Negenborn, Bart De Schutter, Kieran Delaney, and Joe Connell. Kalman Filter-Based Distributed Predictive Control of Large-Scale Multi-Rate Systems: Application to Power Networks. *IEEE Transactions on Control Systems Technology*. Institute of Electrical and Electronics Engineers (IEEE), jan, 2013, Vol. 21, No. 1, pp. 27–39. Available from DOI 10.1109/tcst.2011.2172444.

[48] Farina, Marcello, and Ruggero Carli. Partition-Based Distributed Kalman Filter With Plug and Play Features. *IEEE Transactions on Control of Network Systems*. Institute of Electrical and Electronics Engineers (IEEE), mar, 2018, Vol. 5, No. 1, pp. 560–570. Available from DOI 10.1109/tcns.2016.2633786.

[49] Boem, Francesca, Ruggero Carli, Marcello Farina, Giancarlo Ferrari-Trecate, and Thomas Parisini. Scalable monitoring of interconnected stochastic systems. In: *2016 IEEE 55th Conference on Decision and Control (CDC)*. IEEE, 2016. Available from DOI 10.1109/cdc.2016.7798443.

[50] Boem, Francesca, Ruggero Carli, Marcello Farina, Giancarlo Ferrari-Trecate, and Thomas Parisini. Distributed Fault Detection for Interconnected Large-Scale Systems: A Scalable Plug & Play Approach. *IEEE Transactions on Control of Network Systems*. Institute of Electrical and Electronics Engineers (IEEE), jun, 2019, Vol. 6, No. 2, pp. 800–811. Available from DOI 10.1109/tcns.2018.2878500.

[51] Boem, Francesca, Stefano Riverso, Giancarlo Ferrari-Trecate, and Thomas Parisini. Plug-and-Play Fault Detection and Isolation for Large-Scale Nonlinear Systems With Stochastic Uncertainties. *IEEE Transactions on Automatic Control*. Institute of Electrical and Electronics Engineers (IEEE), jan, 2019, Vol. 64, No. 1, pp. 4–19. Available from DOI 10.1109/tac.2018.2811469.

[52] Ding, Zhen, and Lang Hong. A distributed IMM fusion algorithm for multi-platform tracking. *Signal Processing*. Elsevier BV, jan, 1998, Vol. 64, No. 2, pp. 167–176. Available from DOI 10.1016/s0165-1684(97)98184-6.

[53] Li, Wenling, Yingmin Jia, Junping Du, and Jun Zhang. Distributed consensus filtering for jump Markov linear systems. *IET Control Theory & Applications*. Institution of Engineering and Technology (IET), aug, 2013, Vol. 7, No. 12, pp. 1659–1664. Available from DOI 10.1049/iet-cta.2012.0742.

[54] Li, Wenling, and Yingmin Jia. Distributed Estimation for Markov Jump Systems via Diffusion Strategies. *IEEE Transactions on Aerospace and Electronic Systems*. Institute of Electrical and Electronics Engineers (IEEE), feb, 2017, Vol. 53, No. 1, pp. 448–460. Available from DOI 10.1109/taes.2017.2650801.

[55] Horn, Roger A., and Charles R. Johnson. *Matrix Analysis*. 2nd ed. Cambridge: Cambridge University Press, 2013. ISBN 9780521548236.

[56] Simon, Dan. *Optimal state estimation: Kalman, H infinity, and Nonlinear Approaches*. Hoboken: Wiley, 2006.

[57] Petersen, K. B., and M. S. Pedersen. *The Matrix Cookbook*. Available from `https://www.math.uwaterloo.ca/~hwolkowi/matrixcookbook.pdf`. Version 20121115.

[58] Chui, Charles K., and Guanrong Chen. *Kalman Filtering: with Real-Time Applications*. 3 ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999. Springer Series in Information Sciences Ser.. ISBN 9783662038598.

[59] Anderson, B. D. O., and J. B. Moore. Detectability and Stabilizability of Time-Varying Discrete-Time Linear Systems. *SIAM Journal on Control and Optimization*. Society for Industrial & Applied Mathematics (SIAM), jan, 1981, Vol. 19, No. 1, pp. 20–32. Available from DOI 10.1137/0319002.

[60] Goodwin, Graham C., and Kwai S. Sin. *Adaptive Filtering Prediction and Control*. 1st ed. Englewood Cliffs: Prentice-Hall, 1984. ISBN 9780130040695;013004069X.

[61] Bitmead, Robert R., Michel R. Gevers, Ian R. Petersen, and R.John Kaye. Monotonicity and stabilizability- properties of solutions of the Riccati difference equation: Propositions, lemmas, theorems, fallacious conjectures and counterexamples. *Systems & Control Letters*. Elsevier BV, apr, 1985, Vol. 5, No. 5, pp. 309–315. Available from DOI 10.1016/0167-6911(85)90027-1.

[62] Kim, I. Y., and O. L. de Weck. Adaptive weighted sum method for multiobjective optimization: a new method for Pareto front generation. *Structural and Multidisciplinary Optimization*. Feb., 2006, Vol. 31, No. 2, pp. 105–116. ISSN 1615-1488. Available from DOI 10.1007/s00158-005-0557-6.

[63] Petersen, Ian. *Robust Kalman filtering for signals and systems with large uncertainties*. New York: Springer Science+Business Media, LLC, 1999. ISBN 9781461215943.

[64] Sayed, A.H.. A framework for state-space estimation with uncertain models. *IEEE Transactions on Automatic Control*. Institute of Electrical and Electronics Engineers (IEEE), jul, 2001, Vol. 46, No. 7, pp. 998–1013. Available from DOI 10.1109/9.935054.

[65] Fernandes, Marcos R., Joao B. R. do Val, and Rafael F. Souto. Robust Estimation and Filtering for Poorly Known Models. *IEEE Control Systems Letters*.

Institute of Electrical and Electronics Engineers (IEEE), apr, 2020, Vol. 4, No. 2, pp. 474–479. Available from DOI 10.1109/lcsys.2019.2951611.

[66] Granstrom, Karl, Peter Willett, and Yaakov Bar-Shalom. Systematic approach to IMM mixing for unequal dimension states. *IEEE Transactions on Aerospace and Electronic Systems*. Institute of Electrical and Electronics Engineers (IEEE), oct, 2015, Vol. 51, No. 4, pp. 2975–2986. Available from DOI 10.1109/taes.2015.150015.

[67] Li, Xiao-Rong, and Y. Bar-Shalom. Multiple-model estimation with variable structure. *IEEE Transactions on Automatic Control*. Institute of Electrical and Electronics Engineers (IEEE), apr, 1996, Vol. 41, No. 4, pp. 478–493. Available from DOI 10.1109/9.489270.

[68] Gebäudeausrüstung, VDI-Fachbereich Technische. *VDI 6007 Part 3: Calculation of transient thermal response of rooms and buildings - Modelling of rooms*.

[69] Benner, Peter, and Steffen W. R. Werner. *MORLAB - Model Order Reduction LABoratory*. Available from DOI 10.5281/ZENODO.3332716.

[70] Benner, Peter, Serkan Gugercin, and Karen Willcox. A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems. *SIAM Review*. Society for Industrial & Applied Mathematics (SIAM), jan, 2015, Vol. 57, No. 4, pp. 483–531. Available from DOI 10.1137/130932715.

[71] Conejo, Antonio J., Enrique Castillo, Roberto Minguez, and Raquel Garcia-Bertrand. *Decomposition Techniques in Mathematical Programming*. Springer, 2006. ISBN 978-3-540-27686-9. Available from DOI 10.1007/3-540-27686-6.

[72] Grewal, Mohinder S., and Angus P. Andrews. *Kalman filtering: theory and practice using MATLAB*. 3rd ed. Hoboken: Wiley, 2008. ISBN 0470173661;9780470173664.

[73] Gill, Philip Edward, Walter Murray, and Margaret H. Wright. *Practical Optimization*. Emerald Publishing Limited, 1982. ISBN 0122839528.

[74] Peterka, Václav. Control of uncertain processes: applied theory and algorithms. *Kybernetika*. Institute of Information Theory and Automation AS CR, 1986, Vol. 22, No. Suppl, pp. (1)-101. Available from `http://eudml.org/doc/28730`.

Appendix A
Numerically robust algorithm

In theory, the Kalman filter algorithm includes matrix operations that guarantee the positive semi-definiteness of a covariance matrix. However, in real applications, a covariance matrix can lose positive semi-definiteness due to limited floating point precision, which can cause numerical instability of the algorithm. Decomposing a covariance matrix in factors can diminish these issues. Modified algorithms of the Kalman filter are used to update covariance factors in a way that does not corrupt the positive semi-definiteness of the covariance matrix.

The most common methods are based on Cholesky decomposition [72] where a covariance is decomposed as $\mathbf{P} = \mathbf{L}\mathbf{L}^T$ where $\mathbf{L}$ is a lower triangular matrix. The LDL decomposition is a variant of the Cholesky decomposition where a covariance matrix is decomposed in $\mathbf{L}$ and $\mathbf{d}$ factors

$$\mathbf{P} = \mathbf{L}\mathbf{D}\mathbf{L}^T \equiv |\mathbf{L}; \mathbf{d}| \tag{8.1}$$

where $\mathbf{D}$ is a diagonal matrix with vector $\mathbf{d}$ on the diagonal, $\mathbf{L}$ is a lower triangular matrix with ones on the main diagonal, and $|\mathbf{L}; \mathbf{d}|$ is the notation for the matrix decomposition.

In this chapter, the XDKF-Z with separated steps described in Algorithm 3.8 is interpreted in a factorized form. Second moment updates in the XDKF-Z are reformulated to updates of the $\mathbf{L}$ and $\mathbf{d}$ factors of the second moments.

A.1 Uncertainty-update step

The uncertainty-update step in Algorithm 3.8 is defined as

$$\mathbf{b}_{p,k} = \mathbf{A}_p \hat{\mathbf{x}}_{k|k-1} + \mathbf{B}_p \mathbf{u}_k, \tag{8.2}$$

$$\mathbf{S}_{k|k-1} = -\sum_p \gamma_p \mathbf{b}_{p,k} \hat{\mathbf{s}}^T_{p,k|k-1} \mathbf{C}^T, \tag{8.3}$$

$$\mathbf{Q}^{\Sigma}_{k|k-1} = \alpha\mathbf{Q} + \sum_p \gamma_p \left[\mathbf{b}_{p,k}\mathbf{b}^T_{p,k} - \mathbf{b}_{p,k}\hat{\mathbf{s}}^T_{p,k|k-1}\hat{\mathbf{A}}^T - \hat{\mathbf{A}}\hat{\mathbf{s}}_{p,k|k-1}\mathbf{b}^T_{p,k} \right]. \tag{8.4}$$

In this step, the update of the cumulated process noise moment must be replaced by the update of factors. First, the update is reformulated as

$$\begin{aligned} \mathbf{Q}^{\Sigma}_{k|k-1} &= \alpha\mathbf{Q} \\ &+ \sum_p \gamma_p \left(\hat{\mathbf{A}}\hat{\mathbf{s}}_{p,k|k-1} - \mathbf{b}_{p,k} \right) \left(\hat{\mathbf{A}}\hat{\mathbf{s}}_{p,k|k-1} - \mathbf{b}_{p,k} \right)^T \\ &- \sum_p \gamma_p \left(\hat{\mathbf{A}}\hat{\mathbf{s}}_{p,k|k-1} \right) \left(\hat{\mathbf{A}}\hat{\mathbf{s}}_{p,k|k-1} \right)^T. \end{aligned} \tag{8.5}$$

The process noise covariance $\alpha\mathbf{Q}$ can be decomposed as $\left|\mathbf{L}_\mathbf{Q}; \alpha\mathbf{d}_\mathbf{Q}\right|$. Then the update consists of updating the process noise covariance factors with rank-1 updates and rank-1 downdates sequentially

$$\begin{aligned}\left|\mathbf{L}^{\Sigma}_{k|k-1}; \mathbf{d}^{\Sigma}_{k|k-1}\right| &= \left|\mathbf{L}_\mathbf{Q}; \alpha\mathbf{d}_\mathbf{Q}\right| \\ &+ \sum_p \gamma_p \left(\hat{\mathbf{A}}\hat{\mathbf{s}}_{p,k|k-1} - \mathbf{b}_{p,k}\right)\left(\hat{\mathbf{A}}\hat{\mathbf{s}}_{p,k|k-1} - \mathbf{b}_{p,k}\right)^T \\ &- \sum_p \gamma_p \left(\hat{\mathbf{A}}\hat{\mathbf{s}}_{p,k|k-1}\right)\left(\hat{\mathbf{A}}\hat{\mathbf{s}}_{p,k|k-1}\right)^T. \end{aligned} \tag{8.6}$$

The rank-1 update and downdate algorithms can be found in [73]. The downdates are subtractions of dyadic products from factors. Adding a dyadic product to the positive semi-definite matrix will always result in a positive semi-definite matrix. However, the subtraction can cause numerical instability in the algorithm. Therefore it is important to limit the downdating if the operation would result in a matrix which is not positive definite. See [73] for more details on downdates.

A.2 Data-update step

The prior cumulative second moment is represented by the decomposition

$$\mathbf{P}^{\Sigma}_{k|k-1} = \left|\mathbf{L}^{\Sigma}_{k|k-1}; \mathbf{d}^{\Sigma}_{k|k-1}\right|, \tag{8.7}$$

and the measurement noise covariance is decomposed as

$$\mathbf{R} = \left|\mathbf{L}_\mathbf{R}; \mathbf{d}_\mathbf{R}\right|. \tag{8.8}$$

Then the main computation of the update is done by triangulating the joint probability covariance matrix

$$\begin{bmatrix}\mathbf{P}^{yy} & \mathbf{P}^{yx} \\ \mathbf{P}^{xy} & \mathbf{P}^{xx}\end{bmatrix} \equiv \left|\begin{bmatrix}\mathbf{L}_\mathbf{R} & \mathbf{C}\mathbf{L}^{\Sigma}_{k|k-1} \\ \mathbf{0} & \mathbf{L}^{\Sigma}_{k|k-1}\end{bmatrix}; \begin{bmatrix}\mathbf{d}_\mathbf{R} \\ \mathbf{d}^{\Sigma}_{k|k-1}\end{bmatrix}\right| \longrightarrow \left|\begin{bmatrix}\mathbf{L}_y & \mathbf{0} \\ \mathbf{K}_y & \mathbf{L}^{\Sigma}_{k|k}\end{bmatrix}; \begin{bmatrix}\mathbf{d}_y \\ \mathbf{d}^{\Sigma}_{k|k}\end{bmatrix}\right|, \tag{8.9}$$

where the arrow denotes processing by a triangularization algorithm. An algorithm for triangularization called dyadic reduction was pioneered by Karel Šmuk and published in [74]. Interestingly, the elements of the factors give us the updated factors and gain

$$\mathbf{P}^{\Sigma}_{k|k} = \left|\mathbf{L}^{\Sigma}_{k|k}; \mathbf{d}^{\Sigma}_{k|k}\right|, \tag{8.10}$$

$$\mathbf{K}_k = \mathbf{K}_y \mathbf{L}_y^{-1}. \tag{8.11}$$

Then the gain is used to update the state and the sensitivity

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\left(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1}\right), \tag{8.12}$$

$$\hat{\mathbf{s}}_{p,k|k} = \hat{\mathbf{s}}_{p,k|k-1} - \mathbf{K}_k\mathbf{C}\hat{\mathbf{s}}_{p,k|k-1}. \tag{8.13}$$

A.3 Time-update step

The decorrelated system matrix computation and the propagation of the state and sensitivity remain the same as in Algorithm 3.8

$$\begin{aligned}\hat{\mathbf{A}}_{\mathrm{dc}} &= \hat{\mathbf{A}} - \mathbf{S}_k \left(\alpha \mathbf{R}\right)^{-1} \mathbf{C},\\ \hat{\mathbf{x}}_{k+1|k} &= \hat{\mathbf{A}}_{\mathrm{dc}} \hat{\mathbf{x}}_{k|k} + \hat{\mathbf{B}} \mathbf{u}_k + \mathbf{S}_k \left(\alpha \mathbf{R}\right)^{-1} \mathbf{y}_k,\\ \hat{\mathbf{s}}_{p,k+1|k} &= \hat{\mathbf{A}}_{\mathrm{dc}} \hat{\mathbf{s}}_{p,k|k} - \mathbf{b}_{p,k}.\end{aligned}$$

The second moment update requires more computation. First, the matrix to be subtracted must be triangulated

$$\mathbf{S}_k \left(\alpha \mathbf{R}\right)^{-1} \left(\mathbf{S}_k\right)^T \equiv \left| \mathbf{S}_k \mathbf{L}_{\mathbf{R}}^{-1}; \frac{\alpha}{\mathbf{d}_{\mathbf{R}}} \right| \longrightarrow \left| \mathbf{L}_{\mathbf{SR}}; \mathbf{d}_{\mathbf{SR}} \right|. \tag{8.14}$$

If the system dimension is lower than the output dimension, the triangularization in (8.14) is impossible since the matrix has more rows than columns. However, the **L**-factor can be augmented with a zero matrix to obtain a square shape matrix, and the **d**-factor can be augmented by ones. Then triangularization is possible

$$\left| \left[\mathbf{S}_k \mathbf{L}_{\mathbf{R}}^{-1} \quad \mathbf{0}^{N_x \times (N_x - N_y)} \right]; \begin{bmatrix} \frac{\alpha}{\mathbf{d}_{\mathbf{R}}} \\ \mathbf{1}^{N_x - N_y} \end{bmatrix} \right| \longrightarrow \left| \mathbf{L}_{\mathbf{SR}}; \mathbf{d}_{\mathbf{SR}} \right|. \tag{8.15}$$

After that, the process noise covariance can be downdated as

$$\left| \left[\mathbf{L}_{k|k-1}^{\Sigma} \quad \mathbf{L}_{\mathbf{SR}} \right]; \begin{bmatrix} \mathbf{d}_{k|k-1}^{\Sigma} \\ -\mathbf{d}_{\mathbf{SR}} \end{bmatrix} \right| \longrightarrow \left| \mathbf{L}_{\mathbf{Q}}; \mathbf{d}_{\mathbf{Q}} \right|. \tag{8.16}$$

The downdate can again result in losing the positive semi-definiteness of the covariance. In such a case, the downdate must be skipped. Finally, the second moment factors can be updated

$$\left| \left[\mathbf{L}_{\mathbf{Q}} \quad \hat{\mathbf{A}}_{\mathrm{dc}} \mathbf{L}_{k|k}^{\Sigma} \right]; \begin{bmatrix} \mathbf{d}_{\mathbf{Q}} \\ \mathbf{d}_{k|k}^{\Sigma} \end{bmatrix} \right| \longrightarrow \left| \mathbf{L}_{k+1|k}^{\Sigma}; \mathbf{d}_{k+1|k}^{\Sigma} \right|. \tag{8.17}$$

This concludes the algorithm. Note that in this formulation, the second moments are not stored in memory. Instead, the factors are updated and kept in memory. The second moment can be computed from the factors at any time. The efficiency of this formulation can be perfected, and it is the subject of further research.