LEUPHANA
UNIVERSITÄT LÜNEBURG

Advances in applied nonlinear control and observation methods powered by optimal filtering and disturbance compensation

Von der Fakultät Wirtschaftswissenschaften der
Leuphana Universität Lüneburg zur Erlangung des Grades
Doktor der Ingenieurwissenschaften
– Dr.-Ing. –
genehmigte Dissertation von Benedikt Haus
geboren am 29.07.1990 in Flensburg

Eingereicht am: 28.08.2020

Mündliche Verteidigung (Disputation) am: 17.12.2020

Erstbetreuer und Erstgutachter:	Prof. Dr.-Ing. Paolo Mercorelli
Zweitgutachter:	Prof. Dr.-Ing. habil. Benjamin Klusemann
Drittgutachter:	Prof. Dr.-Ing. Nils Werner (Hochschule Flensburg)

Erschienen unter dem Titel:
Advances in applied nonlinear control and observation methods
powered by optimal filtering and disturbance compensation

Druckjahr: 2021

Vorwort

Zuerst möchte ich mich bei meinen Eltern bedanken, für die jahrelange Unterstützung mit Rat und Tat.

Auch meinem Betreuer und Erstgutachter Prof. Dr.-Ing. Paolo Mercorelli (*Leuphana Universität Lüneburg*) bin ich zu besonderem Dank verpflichtet: für die außerordentliche Betreuung während meiner gesamten Hochschulausbildung, für Freundschaft und Zusammenarbeit, für Konferenzbesuche und zahlreiche gemeinsame Publikationen – und für mehr Freiheit, als ich es mir wünschen könnte.

Das selbe gilt für Prof. Dr.-Ing. Nils Werner (*Hochschule Flensburg*), der selbst noch Doktorand war, als wir unsere (nach wie vor bestehende) Kooperation an der *Ostfalia Hochschule für angewandte Wissenschaften* in Wolfsburg begonnen haben. Nils Werner und seinem Laboringenieur, Dipl.-Ing. Harald Wiegers, bin ich außerdem sehr dankbar für die spannenden praktischen Projekte, an denen wir während meiner Zeit in Flensburg zusammengearbeitet haben – insbesondere am noch laufenden Projekt „Hybridmotorrad“, dessen elektrischer Antrieb die Hauptinspiration für Kapitel 6 war. Auch die Kapitel 3 und 4 sind im Nachgang eines zusammen mit Nils Werner durchgeführten Forschungsprojekts entstanden, das zum Ziel hatte, einen neuartigen Aktor zu erforschen, vorgesehen für vollvariable Ventilsteuerung und somit für einen effizienteren Betrieb von Verbrennungsmotoren. Sie vertiefen einige der dort aufgetretenen praktischen Probleme methodisch.

Mein Dank gilt Dr. Brit-Maren Block (*Leuphana*) für die hervorragende Betreuung und Förderung schon seit meinem Bachelorstudium und Prof. Dr.-Ing. Harald Aschemann (*Universität Rostock*) für die lehrreiche, produktive und angenehme Zusammenarbeit in einigen Publikationen und Projekten.

Prof. Dr.-Ing. habil. Benjamin Klusemann (*Leuphana* und *Helmholtz-Zentrum für Material- und Küstenforschung Geesthacht*) danke ich für die Bereitschaft, als Zweitgutachter zu fungieren und für die vielen hilfreichen Hinweise.

Benedikt Haus *Lüneburg, den 27. August 2020*

Deutsche Zusammenfassung

In dieser Dissertation werden fortgeschrittene nichtlineare Regelungsstrategien und Minimum-Varianz-Schätzalgorithmen kombiniert, um die Schätz- und/oder Regelgüte bei Regelungs- und Fehlerzustandserkennungsaufgaben für jene physikalischen Systemtypen aus den Bereichen Elektromobilität und konventioneller Antriebstechnik zu verbessern, die Potential für Verbesserungen bezüglich Nachhaltigkeit oder Leistung aufweisen. Die anwendungsspezifischen Neuerungen im Hinblick auf nichtlineare Kalman-Filter-Methoden sind:

- Verbesserte Ladestandsschätzung für Lithium-Ionen-Akkuzellen, basierend auf einem neuartigen, selbst-adaptiven EKF, das einen polynomialen Kurven-Fit hoher Ordnung als Zerlegung der unsicherheitsbehafteten, nichtlinearen Ausgangsgleichung verwendet (mit absichtlicherweise redundanten Basen für die Zerlegung und mit einer reduzierten Anzahl von Parametern), wobei die Polynomkoeffizienten durch das EKF selbst angepasst werden.

- Offset-freie Online-Schätzung der Zeitverzögerung (Totzeit) zwischen zwei periodischen Signalen von etwa gleicher Form, die ein ausgeprägtes (zueinander unkorreliertes) Messrauschen aufweisen, basierend auf einer Approximation fraktionaler Ordnung der transzendenten Übertragungsfunktion der Totzeit, die als Modell in einem neuartigen EKF verwendet wird.

- Verwendung von zwei (E)KFs (eines für das lineare Subsystem und eines für das nichtlineare Subsystem eines neuartigen, mehrstufigen piezohydraulischen Aktors) in einer kaskadierten Schleifenstruktur, um die Rechenlast der Schätzung zu reduzieren, wobei die Schleifenstruktur durch geschickte Wahl geeigneter „Schnittstellen“ zwischen den beiden Beobachtern erst ermöglicht wird (u.a. durch die Verwendung einer gemeinsamen Systemmodellgleichung in beiden Beobachtern).

Auch die Innovationen in Bezug auf nichtlineare Regelungsmethoden basieren auf Beobachtern:

- Gleitzustands-Drehzahlregelung eines Gleichstrommotors, der nichtlinearer Reibung und unbekannten Lastmomenten ausgesetzt ist, verbessert durch ein kontinuierliches, zustandsrückführungsbasiertes „äquivalentes Stellgesetz“ und mit einem neuen intelligenten Adaptierungsschema für die Amplitude des diskontinuierlichen Schaltanteils (für reduziertes *chattering* und somit weniger Energieverbrauch/Stellgliedverschleiß), das wiederum ermöglicht wird durch eine linear-modellprädiktive *Meta-Regelung des bereits geregeltem Systems* mit Taylor-linearisiertem Systemmodell, wobei eine beobachterbasierte Störgrößenkompensation (durch ein KF mit einem Doppelintegrator-Störmodell) für *model-matching*-Zwecke erforderlich ist.

- Direkte Drehzahlregelung von permanentmagnet-erregten dreiphasigen Synchronmotoren, die eine hohe spezifische Leistung (bezogen auf das Volumen des Bauraums) haben, basierend auf einer neuartigen Gleitzustands-Regelung in einem mit dem Läufer rotierenden d, q-Koordinatensystem und mit einem neuen „äquivalenten Stellgesetz“ für beide Systemeingänge, und mit einer sekundären Gleitoberfläche, um die beiden Ströme (im d, q-System sind es nur zwei für das dreiphasige System) auf der Trajektorie mit minimalen Strömen für das erforderliche Drehmoment zu halten, wobei diese komplexe Regelungsstrategie ohne Messung des Rotorwinkels arbeitet (dank eines ebenfalls neuartigen EKF, das alle Zustände im statorfesten α, β-Koordinatensystem sowie das Stör-/Lastmoment und dessen Ableitung schätzt).

In allen Fällen wurden Verbesserungen (im Vergleich zu den in der Literatur beschriebenen Methoden) in Bezug auf die Regel- und Schätzgüte erzielt und mittels Simulationsstudien oder Experimenten belegt.

English abstract

In this dissertation, advanced nonlinear control strategies and nonlinear minimum-variance observation are combined, in order to improve the estimation and/or tracking quality within control and fault detection tasks, for several types of systems from the fields of electromobility and conventional drivetrain technology that have some potential for sustainability or performance improvements. The application-specific innovations in terms of nonlinear Kalman filter methods are:

- Improved state of charge estimation for Lithium-ion battery cells, powered by a novel self-adaptive EKF that uses a high-order polynomial curve fit as a decomposition of the uncertain nonlinear output equation with intentionally redundant bases, and with a reduced number of polynomial parameters that are adapted online by the EKF itself.
- Online estimation of the time delay between two periodic signals of roughly the same shape that have pronounced uncorrelated noise, based on a fractional-order approximation of the transcendent transfer function of the time delay which is used as a model in a novel kind of EKF.
- Using two (E)KFs (one for the linear subsystem and one for the nonlinear subsystem of a new kind of multi-stage piezo-hydraulic actuator) in a cascaded loop structure in order to reduce the computation load of the estimation, by appropriate "interfacing" between the two observers (using one shared system model equation, among other aspects).

The innovations in terms of nonlinear control methods are powered by observation, as well:

- Sliding mode velocity control of a DC drive that is subject to nonlinear friction and unknown load torques, enhanced by an equivalent control law, and with a new intelligent switching gain adaptation scheme (for reduced control chattering and, thus, less energy consumption and actuator wear), which is powered by Taylor-linearized model predictive control, which in turn requires observer-based disturbance compensation (by a KF with a double-integrator disturbance model) for model-matching purposes in order to function correctly.
- Direct speed control of permanent-magnet three-phase synchronous motors that have a high power-to-volume ratio, based on sliding mode control in a rotating d, q coordinate system, with a new equivalent control method that exploits both system inputs and with a secondary sliding surface to ensure compliance with the current-trajectory of maximum efficiency for the required torque, and which works without measurement of the rotor angle (thanks to a new kind of EKF that estimates all states in the stationary α, β coordinate system, as well as the disturbance/load torque and its derivative).

In all instances, improvements (compared to methods existing in the literature) in terms of control and estimation performance have been achieved and confirmed using simulation studies or real experiments.

Contents

Acronyms

ADC	Analog-to-digital converter
BMS	Battery management system
CC-CV	Constant current charging followed by constant voltage top-off
DC	Direct current
ECM	Equivalent circuit model
EKF	Extended Kalman filter
Eq.	Equation
FC	Fractional calculus
Fig.	Figure
FPGA	Field-programmable gate array
HPPC	Hybrid pulse power characterization
IC	Initial condition
I/O	Input/output
KF	Kalman filter
LIB	Lithium-ion battery
LQR	Linear-quadratic regulator
LQI	Linear-quadratic-integral regulator
LTI	Linear time-invariant
LUT	Lookup table
MIMO	Multiple-input and multiple-output
MPC	Model predictive control
MTPA	Maximum torque per Ampere
OCV	Open circuit voltage
PID	Proportional-integral-derivative control
PMSM	Permanent-magnet synchronous motor
RK3	Explicit third-order Runge-Kutta method
RPM	Revolutions per minute
SISO	Single-input and single-output
SMC	Sliding mode control
SOC	State of charge
SOH	State of health

Preliminary remarks

0.1 Thesis-related publications

Parts of the content of this dissertation have already been disseminated to the scientific community by the author, as peer-reviewed journal and book chapter publications:

- *Polynomial augmented extended Kalman filter to estimate the state of charge of Lithium-ion batteries* [1], published in the *IEEE Transactions on Vehicular Technology*.
- *An extended Kalman filter for time delays inspired by a fractional order model* [2], published within the *Lecture Notes in Electrical Engineering* book series by Springer.
- *Tracking control of a piezo-hydraulic actuator using input-output linearization and a cascaded extended Kalman filter structure* [3], published in the *Journal of the Franklin Institute*.
- *Gain adaptation in sliding mode control using model predictive control and disturbance compensation with application to actuators* [4], published in *Information* (MDPI).

Instead of citing these publications within the chapters, the assignment of the papers to the chapters is given in Table 0.1, which also indicates the author's contribution to each paper in detail, marked by his initials (B.H.). While these publications were written together with Prof. Dr.-Ing. Paolo Mercorelli

Table 0.1: Authors' contributions to publications [1–4]

Chapter	2	3	4	5
Content from publication	[1]	[2]	[3]	[4]
Supervision	P.M.	P.M.	P.M.	P.M. and H.A.
Conceptualization	B.H. and P.M.	B.H. and P.M.	B.H. and P.M.	B.H.
Methodology	B.H. and P.M.	B.H. and P.M.	P.M. and H.A.	B.H., H.A. and P.M.
Formal analysis	B.H. and P.M.	P.M.	P.M. and H.A.	H.A.
Software development	B.H.	B.H.	B.H.	B.H.
Visualization	B.H.	B.H.	B.H.	B.H.
Writing	B.H. and P.M.	B.H. and P.M.	B.H., P.M. and H.A.	B.H., P.M. and H.A.

(P.M., doctorate supervisor), and two also with Prof. Dr.-Ing. Harald Aschemann (H.A., University of Rostock, Germany), the author was the main contributor for all of them, and also the first and corresponding author. Another two peer-reviewed journal papers were published in collaboration with two former Leuphana PhD students. In terms of methods, they fit the topic of the dissertation in a broader sense. However, they treat physical systems from fundamentally different fields and are, therefore, not part of this document.

- *Extended Kalman Filter for Temperature Estimation and Control of Peltier Cells in a Novel Industrial Milling Process*, written together with Alexandra Mironova [5] and published in the *IEEE Transactions on Industry Applications*.

- *An Extended Kalman Filter as an Observer in a Control Structure for Health Monitoring of a Metal-Polymer Hybrid Soft Actuator*, written together with Manuel Schimmack [6] and published in the *IEEE/ASME Transactions on Mechatronics*.

0.2 Structure of the dissertation

The structure of this dissertation is illustrated in Fig. 0.1, where content descriptions of the five chapters and information about composition and classification can be found. To aid cursory comprehension at a glance, each chapter begins with a short summary that also serves as a contextual transition, followed by content taken from the corresponding papers.

While the chapters treat different applications (categorizable into combustion engine and electromobility systems) and can be read independently, they share many characteristics in terms of the methods that are employed, refined, or derived. The chapters with example applications from the field of electromobility are particularly innovative in terms of the *methodology*, while those chapters using example applications from the field of combustion engine technology additionally deal with innovative *practical developments*.

Obviously, all chapters exploit state observers as a common theme, and some also disturbance observers (chapters 5-6), in order to achieve different nonlinear control (chapters 4-6) or estimation goals (chapters 2-3).

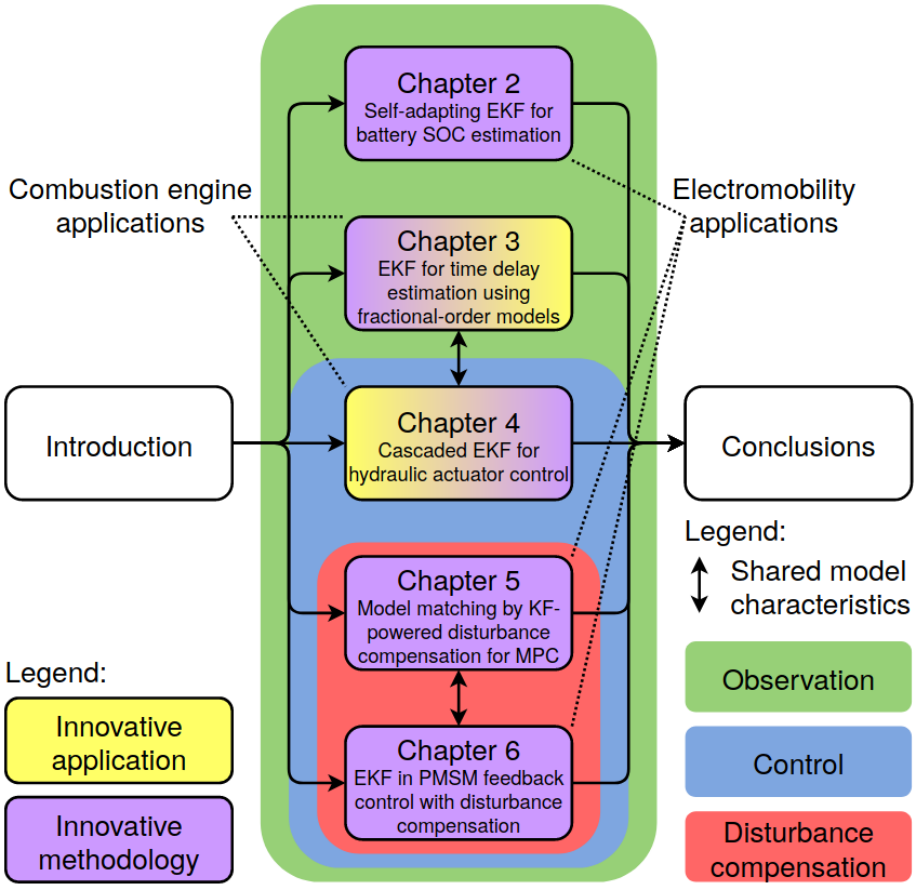


Figure 0.1: Structure of the dissertation with common features and distinctions.

1 Introduction

This dissertation deals with advanced nonlinear control methods and state/disturbance observation algorithms using minimum variance estimation and disturbance compensation. On the one hand, the focus is on using these two approaches to design more intelligent, adaptive control architectures that are specifically derived for selected, important physical systems. On the other hand, this work considers observers for other uses, apart from control purposes in the classic sense, in several scenarios. Observers are an integral component of supervision and fault detection systems and are, therefore, often used when there are corresponding non-functional requirements, for example to ensure operational safety (like monitoring each cell in batteries within electric vehicles to avoid thermal runaway).

In terms of application examples treated in this work, the focus is not only on electromobility (that is, electrical drive technology and electrochemical energy storage), but also on conventional engine technology, since internal combustion engines are a justifiably phased-out, yet still necessary, transitional technology that holds some untapped efficiency gains, even today. These topics can be considered as important fields for a more sustainable near-term future of the engineering sciences, and specifically their applications, both in mobility of people or cargo and in industry.

Estimation/observation of states[1] is an important field in modern control engineering since the space-flight pioneering era in the 1960s, because, in addition to noise suppression of measured output variables, it also makes it possible to estimate quantities that are not measurable (or are not desirable to be measured) by exploiting existing knowledge about the system (that is, a model). Depending on the approach and the system, it is possible to obtain estimates that are at least as accurate as the measured value (if the state variable is measurable at all), but usually even more accurate. Today, one of the most ubiquitous occurrences of this principle is the navigation of a car using GPS or similar satellite-based localization services. While the position measurement is always done via satellite signal, it is usually not accurate enough – at least for non-military users – for advanced car navigation features, like lane detection. However, all cars come with relatively accurate speed sensors (or even directional awareness, for steer-by-wire systems), which can be incorporated as additional measurements into an observer for improved localization accuracy, compared to pure GPS localization. (This can also be taken one step further, using acceleration sensors.) This approach is also called *sensor fusion*, which became a rather overloaded buzz word in the scientific engineering community in recent years.

Traditionally, robustness against parameter variation or disturbances and tracking performance[2] have been conflicting goals in the field of applied control. Practitioners often resort to proportional-integral-derivative (PID) controllers as they are readily available in various dedicated hardware and software modules and require little tuning effort if one of the many existing tuning heuristics is applied, or if an accurate model is known for controller tuning within simulation studies, or if the practitioner is experienced with the type of system to be controlled. Also, PID often performs well enough in the industry, using few (or even single) state feedbacks, and can be tuned to guarantee safe operating conditions at all times – at the cost of a not particularly dynamic response. One notable exception to this general trend is the chemical/pharmaceutical industry, where model predictive control (MPC, from the field of optimal control) has already been used for several decades, enabled by the (usually) very slow processes and, hence, very large sampling times in which optimal or suboptimal setpoints and commands can be computed by means of online optimization. Also, this industry often has huge savings

[1] *state variables* (abbreviated as *states* in this work) of a physical system are usually all energy-storing variables, such as speed (kinetic energy), height or spring displacement (potential energy), electrical current (inductance) or voltage (capacity).

[2] the ability to quickly and accurately follow/track a prescribed, continuously or discontinuously changing trajectory signal.

potentials, in case of large quantities or high value of the educts and products.

Advanced control methods from the (relatively new) fields of *optimal control* and *robust control* have the capability to dissolve, or at least improve, this trade-off between robustness and tracking performance that is present with PID control. The base concept is to incorporate knowledge about the system, as a mathematical model, during the constructive derivation of the control laws. Since this has to be done case by case for each type of system and each set of system parameters, and should – ideally – also be mathematically proven to be asymptotically stable for all realistic working conditions, these advanced approaches are still not very common in actual, real-world applications. In fact, stability proofs become much harder or even impossible if state observers are present in the control loop. However, as mentioned before, state observers are necessary in the numerous cases where some states are not measurable, but are required by the chosen advanced control strategy.

Many control engineers work exclusively on a theoretical basis, often not even considering specific real-world systems but, instead, rather abstracted classes of systems, evaluating their control algorithms in simulations using synthetic numerical examples. This dissertation, on the other hand, aims to fall right into the notorious trench between theory and practice in control engineering (but perhaps a little closer to the practice side). While the (realistic) example systems treated in this work may not offer much money-saving potential like the chemical industry, significant control performance improvements (compared to classic PID control) have been achieved. Rigorous mathematical convergence proofs or stability analyses involving the state/disturbance observer in the loop are *not* given anywhere in this work[3], due to the complications mentioned above, but the derived algorithms are always implemented and thoroughly tested either in simulation studies using Matlab/Simulink, or even using dedicated hardware test-benches, for evaluation and validation.

1.1 Mathematical fundamentals

State estimation/observation algorithms are called observers; they always consist of a combination of *a-priori* existing model[4] knowledge about the system and *a-posteriori* available measured values.

Notable historical milestones of this estimation principle is the Luenberger observer, named after the US-American electrical engineer David Gilbert Luenberger, and the Kalman filter (KF). The Luenberger observer, arranged in parallel to the real system in state-space form (see Fig. 1.1), simulatively calculates (predicts) the system outputs using an identified system model. The difference of these predicted outputs and the measured outputs (= prediction error) is then weighted with a constant gain matrix and fed back into the system model, in order to determine the states. In this sense, one can speak of a *regulated/controlled algorithm*, in analogy to the classical P-controller with additional feedforward control. The gain matrix, typically denoted by $\mathbf{L}$, can be tuned using pole placement methods, which poses another similarity with control.

The KF, named after the Hungarian mathematician Rudolf Emil Kálmán (who passed away only recently, in 2016), has a structure very similar to the Luenberger observer (even though its scope and derivation is entirely different), and automatically adapts the gain matrix with which the estimation error is fed back. The discrete-time form (being the most practice-relevant, due to the omnipresence of digital control systems) of the KF for linear time-invariant (LTI) systems can be seen in Fig. 1.2. Note that the discrete-time Luenberger observer has an identical structure; the only difference lies in the adaptation of the feedback gain matrix that is done by the KF. This matrix is calculated based on the solution of a (in discrete-time: algebraic) Riccati differential equation, involving the dynamics of the covariances, in such a way that the covariance (more precisely: the trace of the covariance matrix), which describes the system uncertainty, is minimized. Therefore, the KF is also called the minimum variance estimator. It can be extended to the nonlinear case, see Fig. 1.3 where the extended Kalman filter [7] (EKF) is shown within a control loop. The EKF is not the only nonlinear extension of the KF,

[3] in case of sliding mode control, as in chapters 5 and 6, the stability proof is inherently included in the derivations.

[4] *model* in this context means the *system dynamics* formulated in state space, resulting in a system of first-order differential equations. This is why differential equations for all states are best suited for the observer-oriented modelling.

but the one that is most commonly found in the literature. The (E)KF is one of the most widely-used observers today, due to its optimality (under certain mathematical preconditions) and relatively simple, iterative implementation, and is the common denominator of all chapters of this dissertation.

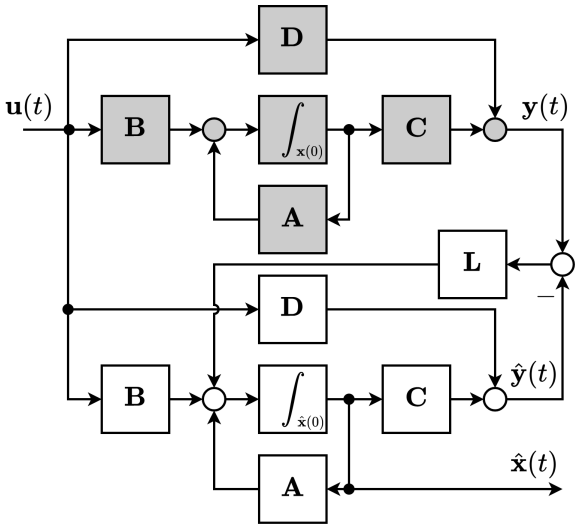


Figure 1.1: Block diagram of the continuous-time Luenberger observer. The gray blocks represent the real system, modelled as LTI, with known parameters.

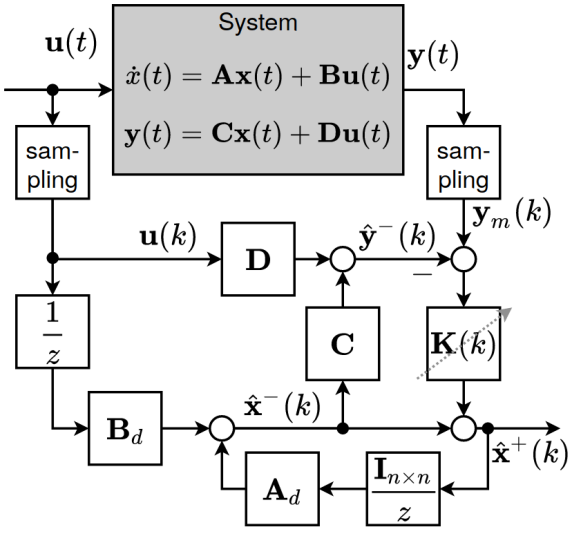


Figure 1.2: Block diagram of the discrete-time KF algorithm for LTI systems, where $\mathbf{K}(k)$ is adapted accordingly. The $1/z$ blocks represent the unit delay for one step size T_s (with corresponding ICs). $\mathbf{I}_{n\times n}$ is the identity matrix where n is the system order (state vector size). Matrices $\mathbf{A}_d = \mathbf{I}_{n\times n} + \mathbf{A}T_s$ and $\mathbf{B}_d = \mathbf{B}T_s$ in case of explicit Euler discretization with step size T_s. For constant feedback gain $\mathbf{K}(k) = \mathbf{L}_d$, this is equivalent to the discretized version of the continuous-time Luenberger observer with gain $\mathbf{L}$, which can also be discretized using explicit Euler as $\mathbf{L}_d = \mathbf{L}T_s$.

1.1.1 Observability and choice of measured outputs

An important property of systems is their *observability*, which formally is a sufficient condition for the existence of an observer. A system is called observable if its initial state can be reconstructed using the system model and progressions of the inputs and outputs over a finite time interval [8]. Practically,

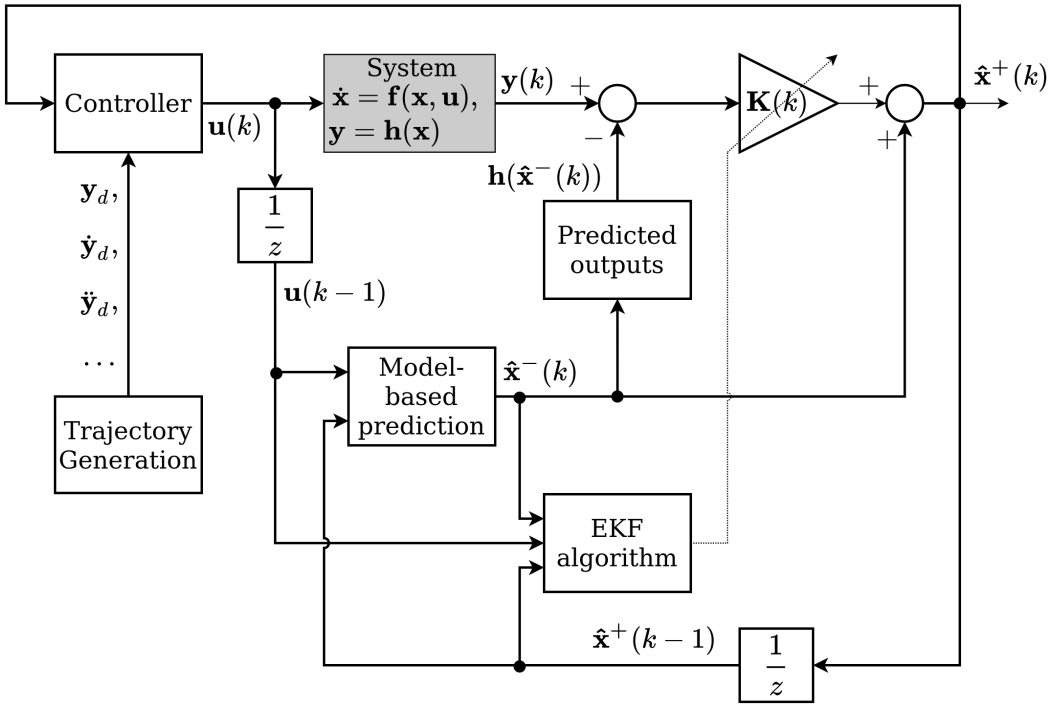


Figure 1.3: Schematic of the discrete-time extended Kalman filter algorithm within a control loop.

observability allows to reconstruct the states of the system (at any past point in time), using information about past and current values of the inputs/outputs and the system model. See Table 1.1 for an overview of rank conditions for observability of the most important classes of systems. Note that these rank

Table 1.1: Observability rank conditions (simplified) for different classes of systems

Linear time-invariant [8]	**Analytically varying [8]**	**Nonlinear [9]**
$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$ $\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$	$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t)$ $\mathbf{y}(t) = \mathbf{C}(t)\mathbf{x}(t)$	$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, \mathbf{u})$ $\mathbf{y}(t) = \mathbf{h}(\mathbf{x})$
$\mathbf{O} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \dots \\ \mathbf{CA}^{n-1} \end{bmatrix}$	$\mathbf{O}(t) = \begin{bmatrix} \mathbf{O}_0 \\ \mathbf{O}_1 \\ \mathbf{O}_2 \\ \dots \\ \mathbf{O}_{n-1} \end{bmatrix}$ $\mathbf{O}_0(t) = \mathbf{C}(t)$ $\mathbf{O}_{i+1}(t) = \mathbf{O}_i(t)\mathbf{A}(t) + \frac{\mathrm{d}\mathbf{O}_i(t)}{\mathrm{d}t}$	$\mathbf{O}(\mathbf{x}, \mathbf{u}, t) = \begin{bmatrix} \mathbf{O}_0 \\ \mathbf{O}_1 \\ \mathbf{O}_2 \\ \dots \\ \mathbf{O}_{n-1} \end{bmatrix}$ $= \begin{bmatrix} \frac{\partial \mathbf{h}}{\partial \mathbf{x}^T} \\ \frac{\partial \mathcal{L}_\mathbf{f} \mathbf{h}}{\partial \mathbf{x}^T} \\ \frac{\partial \mathcal{L}_\mathbf{f}^2 \mathbf{h}}{\partial \mathbf{x}^T} \\ \dots \\ \frac{\partial \mathcal{L}_\mathbf{f}^{n-2} \mathbf{h}}{\partial \mathbf{x}^T} \end{bmatrix} = \begin{bmatrix} \frac{\partial \mathbf{h}}{\partial \mathbf{x}^T} \\ \frac{\partial \mathbf{O}_0 \mathbf{f}}{\partial \mathbf{x}^T} \\ \frac{\partial \mathbf{O}_1 \mathbf{f}}{\partial \mathbf{x}^T} \\ \dots \\ \frac{\partial \mathbf{O}_{n-2} \mathbf{f}}{\partial \mathbf{x}^T} \end{bmatrix}$
$\mathrm{rank}(\mathbf{O}) = n$	$\mathrm{rank}(\mathbf{O}(t)) = n$	$\mathrm{rank}(\mathbf{O}(\mathbf{x}, \mathbf{u}, t)) = n$

conditions only provide a *qualitative* yes/no predication, but no quantitative measure (index/metric) of the observability. Numerous *quantitative* observability measures have been reported in the literature, for example the one by Lückel and Müller [10] which is based on eigenvectors and eigenvalues (hence, usable only for linear systems), the evaluation of the observability Gramian matrix [11] (for linear and nonlinear systems, thanks to local linearization), or the one recently proposed by Letellier et al., [12] (for nonlinear systems). However, all of the approaches mentioned above suffer a lack of comparability across systems, as they are influenced by the scale of the system's states, parameters, and/or dynamics. However, particularly the measure by Letellier et al. has some convenient uses during the design stage of feedback controllers for a specific system, as it allows to determine which system outputs should be measured (and also what the influence of different sensors is going to be, provided the sensor model is known) and which outputs are less important in order to maintain "enough" observability for the specific control goals. Another interesting (but, in this context, less important) aspect of the observability property is its kinship with the controllability property of a system. In fact, the two are often described as *dual* properties [8]. For systems that naturally have state-regions of "low observability"[5], as for example the one treated in chapter 2 (where a decomposition with intentionally redundant bases is employed as a fix), there are some work-arounds to keep the observer from "falling asleep" (also called *dropping-off phenomenon*, where new measurement information no longer corrects the estimates). A new streaming in the literature is *observability-oriented control* [13] that considers such regions of low observability and aims to avoid them, based on some kind of observability metric, with or without specified thresholds.

1.1.2 Observer-oriented modelling

Apart from model accuracy requirements, verification/validation, and maintaining the system's observability also in its model, there are some further aspects that should be considered when deriving models (that is, formulating differential equations describing each state's dynamics) for the purpose of observer design. An important aspect is the *size* of the model, i.e. the number of states whose *interaction* should be modelled and observed. This number, often denoted by n, is a good indicator of the computational load that is generated by evaluating the model numerically (for prediction). For linear and linearized systems of order n, state transition matrices[6] are $n \times n$-dimensional, so in case of matrix multiplications, the (theoretical worst case) computation time is proportional to the square of the number of states. For physical systems with many energy-storing state variables, like that in chapter 4, it can be sensible to split the system into several subsystems. This is especially reasonable if one subsystem is much faster than the other ones. In chapter 4, for example, the oil pressure dynamics are treated in a dedicated subsystem since they are, a.) much faster than the mechanical system that consists of springs and dampers, and b.) nonlinear, unlike the rest of the system, thus requiring a different type of observer. Splitting the model in two parts requires for *interfacing*. In this sense, states that have impacts on other states' dynamics within more than one subsystem must be considered as new inputs/outputs of these subsystems, in order to maintain the correctness of the overall model. It is sensible to keep the number of states as low as possible in a model that is to be used within observers, not only for the mentioned computational efficiency considerations, but also for *estimation robustness*. This aspect surfaces in chapter 3, where the time delay is (in a first step) approximated using a linear low pass filter. Theoretically, the transfer function of the time delay requires an infinite model order in implementation (e.g. the Padé approximation). Using a first-order approximation for the model, as was done here, unsurprisingly results in significant estimation biases. However, using a second-order approximation instead already resulted in stability and convergence issues of the observer. Hence, the first-order approximation was used. The problem of the remaining bias is solved in chapter 3 by introducing an augmented adaptation parameter, inspired by the mathematical field of non-integer-order calculus.

For the decision about the model size and potential split-up into subsystems, another aspect should be regarded, which is the inclusion of model parameters or disturbances as *augmented states*. Often, system

[5] A better term would be "weak observability", but that has already been overloaded with different conflicting mathematical definitions during the last decades.

[6] also called *system matrices*; only meaningful for linear and linearized systems

behavior will change during its lifetime, causing a model mismatch between observer and real system that will eventually cause the observer to stop converging. Since this violates the assumptions made for the observer derivation, in particular the zero-mean assumption of the process noise, modelling the causative parameters as *variables* (instead of constants that are assumed to be known) is often necessary. The model for the parameter's dynamics can be an open integrator, i.e. $\dot{p} = 0$ in case no further dynamics information is available (done in chapters 2 and 3), or an actual model (e.g. the temperature dependence of an electrical resistor, done in [5, 14]). This is then included as an additional differential equation in the overall state-space model; the new state is called an augmented state. For unknown lumped disturbance signals[7] the simplest model is the integrator disturbance model, identical to the open-integrator model mentioned above. Another common choice is the double integrator disturbance model, especially if the disturbance's time derivative is also required, as in chapters 6 and 5. Including parameters as augmented states together with the physical state variables is particularly reasonable if the parameters change about as fast as the physical states, as in chapter 2. If, however, there are many parameters to be estimated or if the timescales of change differ a lot between physical states and augmented parameters, it can be advantageous to build a secondary, dedicated observer exclusively for the parameters, which simplifies the allocation of the measured information into either physical state estimates or parameter estimates. It also allows for further gains in terms of computational efficiency if the state vector size of the primary observer can be shrunk enough, due to the influence of n^2 described above.

1.1.3 The Kalman filter algorithm

In this subsection, the algorithm equations of the minimum variance estimator – the Kalman filter – are given. For nonlinear systems, the EKF can be used, which is structurally very similar to the KF since it is based on Taylor linearization of the nonlinear system and uses a Jacobian matrix instead of a system state transition matrix. Due to the similarity, only the linear case is presented in this subsection. Multiple different mathematical derivations based on analytical minimization of the estimation error covariance matrix are available in the literature (e.g. [15]) but will not be detailed in this work. Instead, a much more intuitive derivation from [16, pp. 291-292], inspired by Kálmán's original 1960 paper [17, Theorem 2], is outlined in the following. The discrete-time KF algorithm consists of two steps (see Fig. 1.4)

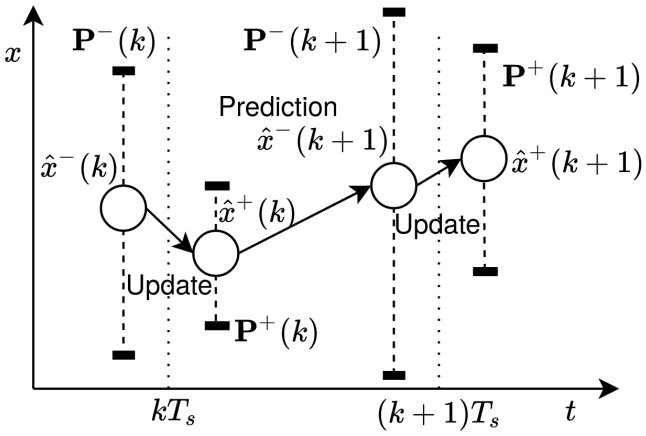


Figure 1.4: Evolution of a single state and its uncertainty (represented by error bars, related to the covariance matrix $\mathbf{P}$) during prediction and correction steps of the KF algorithm.

that are executed in iteration: a prediction of the states one time step into the future, based on the time-discretized model equations of the states' dynamics, and a correction step that is based on the output equation and allows for inclusion of measured data for one or multiple outputs.

[7] sometimes called *unknown inputs*

Presuppositions

The model for the prediction step of the linear KF must be available in state-space form, with state vector $\mathbf{x}(t)$, input vector $\mathbf{u}(t)$, outputs $\mathbf{y}(t)$, and a system of first-order differential equations $\dot{\mathbf{x}}(t)$

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \qquad \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t). \tag{1.1}$$

Note that the system can also be time-variant, i.e. matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ are generally time-varying. For the sake of simplicity of notation, they are assumed to be constant (LTI system). Also, matrix $\mathbf{D}$ (mapping the feedthrough of the inputs into the outputs, if any) is ignored in the following, because its presence does not change the derivation much. Thus, the system to be considered can be stated as

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \qquad \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t). \tag{1.2}$$

The discrete-time state-space model can be obtained, for example, using explicit Euler discretization with step width T_s, after which the random variables $\mathbf{w}(k)$ and $\mathbf{v}(k)$ are introduced to represent white, uncorrelated, zero-mean process and measurement noise, respectively, with normal probability distributions [16].

$$\mathbf{x}(k+1) = \mathbf{A}_d\mathbf{x}(k) + \mathbf{B}_d\mathbf{u}(k) + \mathbf{w}(k), \tag{1.3}$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{v}(k), \tag{1.4}$$

$$\mathbf{A}_d = \mathbf{I}_{n\times n} + T_s\mathbf{A}, \qquad \mathbf{B}_d = T_s\mathbf{B}. \tag{1.5}$$

The notation $\mathbf{x}(k)$ is an abbreviation of $\mathbf{x}(t) = \mathbf{x}(T_s k)$ and is commonly used for sampled, discrete-time signals. The covariances (which can also be time-variant) of the process and measurement noise are

$$\mathbf{Q}(k) = \mathrm{E}\left[\mathbf{w}(k)\mathbf{w}(k)^T\right], \qquad \mathbf{R}(k) = \mathrm{E}\left[\mathbf{v}(k)\mathbf{v}(k)^T\right], \tag{1.6}$$

and are assumed to be known during the derivation of the algorithm. (In practice, they are usually treated as tuning parameters in the observer design stage.)

Prediction step

For the prediction of the states one time step into the future, the a-priori estimates are calculated based on the discretized model as

$$\hat{\mathbf{x}}^-(k) = \mathbf{A}_d\hat{\mathbf{x}}^+(k-1) + \mathbf{B}_d\mathbf{u}(k-1), \tag{1.7}$$

which is possible thanks to the assumption $\mathrm{E}[\mathbf{w}(k)] = \mathbf{0}$. For the first step, initial values $\hat{\mathbf{x}}^+_{KF}(0)$ can be either specified by the user or simply set to zero.

Since the algorithm is based on covariance minimization, the covariance matrix of the prediction error (called a-priori covariance matrix) must be calculated:

$$\mathbf{P}^-(k) = \mathrm{E}\left[\left(\mathbf{x}(k) - \hat{\mathbf{x}}^-(k)\right)\left(\mathbf{x}(k) - \hat{\mathbf{x}}^-(k)\right)^T\right], \tag{1.8}$$

where $\mathbf{x}(k)$ are the true (unknown) values of the states according to Eq. (1.3). Evaluating the prediction error yields

$$\begin{aligned}\mathbf{x}(k) - \hat{\mathbf{x}}^-(k) &= \mathbf{A}_d\mathbf{x}(k-1) + \mathbf{B}_d\mathbf{u}(k-1) + \mathbf{w}(k-1) - \mathbf{A}_d\hat{\mathbf{x}}^+(k-1) - \mathbf{B}_d\mathbf{u}(k-1)\\ &= \mathbf{A}_d\left(\mathbf{x}(k-1) - \hat{\mathbf{x}}^+(k-1)\right) + \mathbf{w}(k-1).\end{aligned} \tag{1.9}$$

Then,

$$\begin{aligned}\mathbf{P}^-(k) &= \mathrm{E}\left[\left(\mathbf{A}_d\left(\mathbf{x}(k-1) - \hat{\mathbf{x}}^+(k-1)\right) + \mathbf{w}(k-1)\right)\left(\left(\mathbf{x}(k-1) - \hat{\mathbf{x}}^+(k-1)\right)^T\mathbf{A}_d^T + \mathbf{w}(k-1)^T\right)\right]\\ &= \mathbf{A}_d\mathrm{E}\left[\left(\mathbf{x}(k-1) - \hat{\mathbf{x}}^+(k-1)\right)\left(\mathbf{x}(k-1) - \hat{\mathbf{x}}^+(k-1)\right)^T\right]\mathbf{A}_d^T + \mathrm{E}\left[\mathbf{w}(k-1)\mathbf{w}(k-1)^T\right]\\ &= \mathbf{A}_d\mathbf{P}^+(k-1)\mathbf{A}_d^T + \mathbf{Q},\end{aligned} \tag{1.10}$$

where the a-posteriori estimation error covariance matrix from the last iteration is introduced as

$$\mathbf{P}^+(k-1) = \mathrm{E}\left[\left(\mathbf{x}(k-1) - \hat{\mathbf{x}}^+(k-1)\right)\left(\mathbf{x}(k-1) - \hat{\mathbf{x}}^+(k-1)\right)^T\right]. \tag{1.11}$$

Note that the two mixed terms of Eq. (1.10), involving the products of $(\mathbf{x} - \hat{\mathbf{x}}^+)$ and $\mathbf{w}$, vanish when evaluating the expectation, due to lack of correlation (as per assumption). A similar property is exploited in the correction step, as well – see Remark 1.1 for a more detailed explanation.

Correction step

Using the a-posteriori state estimates

$$\hat{\mathbf{x}}^+(k) = \hat{\mathbf{x}}^-(k) + \mathbf{K}(k)\left(\mathbf{y}(k) - \mathbf{C}\hat{\mathbf{x}}^-(k)\right), \tag{1.12}$$

the a-posteriori estimation error can be expressed with the true values $\mathbf{x}(k)$ as

$$\begin{aligned}\mathbf{x}(k) - \hat{\mathbf{x}}^+(k) &= \mathbf{x}(k) - \hat{\mathbf{x}}^-(k) - \mathbf{K}(k)\left(\mathbf{y}(k) - \mathbf{C}\hat{\mathbf{x}}^-(k)\right) = \mathbf{x}(k) - \hat{\mathbf{x}}^-(k) - \mathbf{K}(k)\left(\mathbf{C}\mathbf{x}(k) + \mathbf{v}(k) - \mathbf{C}\hat{\mathbf{x}}^-(k)\right)\\ &= \left(\mathbf{I}_{n\times n} - \mathbf{K}(k)\mathbf{C}\right)\left(\mathbf{x}(k) - \hat{\mathbf{x}}^-(k)\right) - \mathbf{K}(k)\mathbf{v}(k),\end{aligned} \tag{1.13}$$

where $\mathbf{I}_{n\times n}$ is the n-by-n identity matrix, abbreviated as $\mathbf{I}$ in the following. The covariance matrix of the a-posteriori estimation error then is

$$\begin{aligned}\mathrm{E}\left[\left(\mathbf{x}(k) - \hat{\mathbf{x}}^+(k)\right)\left(\mathbf{x}(k) - \hat{\mathbf{x}}^+(k)\right)^T\right] &= \mathrm{E}\left[\left(\mathbf{I} - \mathbf{K}(k)\mathbf{C}\right)\left(\mathbf{x}(k) - \hat{\mathbf{x}}^-(k)\right)\left(\mathbf{x}(k) - \hat{\mathbf{x}}^-(k)\right)^T\left(\mathbf{I} - \mathbf{K}(k)\mathbf{C}\right)^T\right]\\ &\quad - \mathrm{E}\left[\left(\mathbf{I} - \mathbf{K}(k)\mathbf{C}\right)\left(\mathbf{x}(k) - \hat{\mathbf{x}}^-(k)\right)\mathbf{v}(k)^T\mathbf{K}(k)^T\right]\\ &\quad - \mathrm{E}\left[\mathbf{K}(k)\mathbf{v}(k)\left(\mathbf{x}(k) - \hat{\mathbf{x}}^-(k)\right)^T\left(\mathbf{I} - \mathbf{K}(k)\mathbf{C}\right)^T\right]\\ &\quad + \mathrm{E}\left[\mathbf{K}(k)\mathbf{v}(k)\mathbf{v}(k)^T\mathbf{K}(k)^T\right].\end{aligned} \tag{1.14}$$

However, since the output noise $\mathbf{v}(k)$ is not correlated with the a-priori estimation error, the two mixed terms in the second and third line of Eq. (1.14) result to be zero, similar to the case of Eq. (1.10) where the mixed terms have been omitted directly. Evaluating the expectations, this leads to

$$\begin{aligned}&\mathrm{E}\left[\left(\mathbf{x}(k) - \hat{\mathbf{x}}^+(k)\right)\left(\mathbf{x}(k) - \hat{\mathbf{x}}^+(k)\right)^T\right] =\\ &\left(\mathbf{I} - \mathbf{K}(k)\mathbf{C}\right)\mathrm{E}\left[\left(\mathbf{x}(k) - \hat{\mathbf{x}}^-(k)\right)\left(\mathbf{x}(k) - \hat{\mathbf{x}}^-(k)\right)^T\right]\left(\mathbf{I} - \mathbf{K}(k)\mathbf{C}\right)^T + \mathbf{K}(k)\mathrm{E}\left[\mathbf{v}(k)\mathbf{v}(k)^T\right]\mathbf{K}(k)^T.\end{aligned} \tag{1.15}$$

With the a-priori estimation error covariance $\mathbf{P}^-(k) = \mathrm{E}\left[\left(\mathbf{x}(k) - \hat{\mathbf{x}}^-(k)\right)\left(\mathbf{x}(k) - \hat{\mathbf{x}}^-(k)\right)^T\right]$ and the a-posteriori estimation error covariance $\mathbf{P}^+(k) = \mathrm{E}\left[\left(\mathbf{x}(k) - \hat{\mathbf{x}}^+(k)\right)\left(\mathbf{x}(k) - \hat{\mathbf{x}}^+(k)\right)^T\right]$, this yields

$$\mathbf{P}^+(k) = \left(\mathbf{I} - \mathbf{K}(k)\mathbf{C}\right)\mathbf{P}^-(k)\left(\mathbf{I} - \mathbf{K}(k)\mathbf{C}\right)^T + \mathbf{K}(k)\mathbf{R}\mathbf{K}(k)^T \tag{1.16}$$

in which $\mathbf{R} = \mathrm{E}\left[\mathbf{v}(k)\mathbf{v}(k)^T\right]$ in accordance with Eq. 1.6.

In order to find the optimal Kalman gain matrix $\mathbf{K}(k)$, the orthogonality principle can be used [16], as was done in Kalman's original work [17]:

$$\begin{aligned}\mathrm{E}\left[\left(\mathbf{x}(k) - \hat{\mathbf{x}}^+(k)\right)\mathbf{y}(k)^T\right] &= \mathrm{E}\left[\left(\mathbf{x}(k) - \hat{\mathbf{x}}^-(k) - \mathbf{K}(k)\left(\mathbf{C}\left(\mathbf{x}(k) - \hat{\mathbf{x}}^-(k)\right) + \mathbf{v}(k)\right)\right)\left(\mathbf{C}\mathbf{x}(k) + \mathbf{v}(k)\right)^T\right]\\ &= \mathrm{E}\left[\left(\left(\mathbf{I} - \mathbf{K}(k)\mathbf{C}\right)\left(\mathbf{x}(k) - \hat{\mathbf{x}}^-(k)\right) - \mathbf{K}(k)\mathbf{v}(k)\right)\left(\mathbf{x}(k)^T\mathbf{C}^T + \mathbf{v}(k)^T\right)\right]\\ &= \left(\mathbf{I} - \mathbf{K}(k)\mathbf{C}\right)\mathbf{P}^-(k)\mathbf{C}^T - \mathbf{K}(k)\mathbf{R}.\end{aligned} \tag{1.17}$$

Remark 1.1 *The orthogonality principle can be seen as a lack of correlation between the a-posteriori estimation error and the measured outputs, which only is the case for optimal values of* $\mathbf{K}(k)$ *that are to be found. In other words, a suitable choice of* $\mathbf{K}(k)$ *will achieve complete*[8] *denoising of the state estimates* $\hat{\mathbf{x}}^+(k)$*, such that no stochastic influence of the output noise* $\mathbf{v}$ *(via the measured outputs* $\mathbf{y}$*) gets into the state estimates, characterized by the described absence of correlation between outputs and a-posteriori estimation error.*

Thus, the approach is to set

$$\mathrm{E}\left[\left(\mathbf{x}(k) - \hat{\mathbf{x}}^+(k)\right)\mathbf{y}(k)^T\right] = \mathbf{0}. \tag{1.18}$$

Then, using Eq. (1.17),

$$\begin{aligned}
\mathbf{0} &= \left(\mathbf{I} - \mathbf{K}(k)\mathbf{C}\right)\mathbf{P}^-(k)\mathbf{C}^T - \mathbf{K}(k)\mathbf{R} \\
&= \mathbf{P}^-(k)\mathbf{C}^T - \mathbf{K}(k)\mathbf{C}\mathbf{P}^-(k)\mathbf{C}^T - \mathbf{K}(k)\mathbf{R} && (1.19) \\
\mathbf{P}^-(k)\mathbf{C}^T &= \mathbf{K}(k)\mathbf{C}\mathbf{P}^-(k)\mathbf{C}^T + \mathbf{K}(k)\mathbf{R} \\
&= \mathbf{K}(k)\left(\mathbf{C}\mathbf{P}^-(k)\mathbf{C}^T + \mathbf{R}\right) && (1.20) \\
\mathbf{K}(k) &= \mathbf{P}^-(k)\mathbf{C}^T\left(\mathbf{C}\mathbf{P}^-(k)\mathbf{C}^T + \mathbf{R}\right)^{-1}. && (1.21)
\end{aligned}$$

An intuitive demonstration of the optimality of this solution can be found in [18]. Considering the expression (1.16) for the a-posteriori covariance, together with some algebraic manipulations and introduction of the symbol $\mathbf{S}(k) = \mathbf{C}\mathbf{P}^-(k)\mathbf{C}^T + \mathbf{R}$, it is possible to find the algebraic Riccati equation which describes the discrete-time dynamics of the covariance

$$\mathbf{P}^+(k) = \mathbf{P}^-(k) - \mathbf{P}^-(k)\mathbf{C}^T\mathbf{S}(k)^{-1}\mathbf{C}\mathbf{P}^-(k) + \left(\mathbf{K}(k) - \mathbf{P}^-(k)\mathbf{C}^T\mathbf{S}(k)^{-1}\right)\mathbf{S}(k)\left(\mathbf{K}(k) - \mathbf{P}^-(k)\mathbf{C}^T\mathbf{S}(k)^{-1}\right)^T. \tag{1.22}$$

The goal still is to minimize the covariance using suitable choices of Kalman gain $\mathbf{K}(k)$, which can be done directly and intuitively considering this expression. In there, the latter summand $(...)\mathbf{S}(k)(...)^T$ is a quadratic form – therefore, the covariance will always grow if this term does not vanish. The best possible choice in such circumstances is therefore to eliminate the quadratic term using

$$\mathbf{K}(k) = \mathbf{P}^-(k)\mathbf{C}^T\mathbf{S}(k)^{-1} = \mathbf{P}^-(k)\mathbf{C}^T\left(\mathbf{C}\mathbf{P}^-(k)\mathbf{C}^T + \mathbf{R}\right)^{-1}. \tag{1.23}$$

The algorithm equations together with this solution can be implemented in real time-capable embedded systems, ideally using an optimized library for linear algebra. Apart from hardware-level C or even assembly programming, Matlab/Simulink allows to automatically generate and compile C code for the specified hardware target, making the implementation not trivial, but very straight-forward – the most challenging part still is to formulate a sensible system model to base the observer on.

1.2 Disturbance compensation for control purposes

In its most basic form, feedback control works by comparing the actual value of a state with the desired value and then influencing the system accordingly to reach the desired value. Many systems can be controlled satisfactorily using PID action on the control error (the difference between desired and actual value) of a selected system output. In such cases, a more dynamic response of the closed loop system can only be achieved using larger gains in the P, I, and D paths. Apart from peaking problems and actuator saturation, such a *high-gain* control system is prone to extreme amplification of the measurement noise that propagates into the system inputs, which can lead to increased energy

[8] *Perfect* denoising can obviously only be achieved if the system model (including parameters) and the covariances $\mathbf{Q}, \mathbf{R}$ are known with absolute certainty, which is never completely true in real life. However, KF *do* perform very well in practice, even if multiple system parameters are only approximately known (or if they are modelled as constants but change during operation). In practice, $\mathbf{Q}, \mathbf{R}$ are used as tuning parameters, manual or automatically-adaptive.

consumption and actuator wear-out failure. Denoising the feedback signals using a state observer or a filter[9] can mitigate this effect to a certain degree, but will generalle leave the closed-loop system either unrobust or (relatively) slow. Thus, it is generally advisable to "unburden" the controller gains as much as possible by exploiting additional knowledge about the system. This is especially true if there is a trajectory to be tracked, that is, a time-varying reference signal as a desired value in the control loop. This can be taken into account using an additional feedforward control action over the desired trajectory signal (and its derivatives), e.g. exploiting the *differential flatness* property of a system [19]. Taking this approach one step further, it is possible to compensate observable disturbances that act on the system by adjusting the inputs accordingly, which works for matched and unmatched disturbances[10]. This was done in chapters 5 and 6, where electromagnetic actuators/motors are to be controlled in a velocity tracking task. The disturbance torques (and their derivatives) are estimated by (E)KFs and used within the control laws, allowing for lower controller gains, which improves both the noise/chattering levels and the tracking performance. While there are many different approaches to estimate disturbances, only those that incorporate model knowledge (like the KF) perform well under pressure, which becomes obvious in chapter 5, where KF-based disturbance estimation clearly outperforms two other traditional disturbance estimation methods that rely exclusively on measurement signal processing.

1.3 State and parameter monitoring for fault detection and diagnosis

Beyond feedback control, another field where observers are advantageously employed is fault detection and diagnosis. In safety-critical systems, it is often mandated by legislation to fulfill certain non-functional requirements. Notable cases are passenger cars and airplanes, where critical components like sensors often need to be installed redundantly[11] and to be constantly monitored for correct functionality. This task can be achieved using *fault detection* methods, where measured data is checked for plausibility (allowing to infer the physical integrity of components) using additional information sources. Observer-based fault detection strategies can be derived by directly measuring the state that is to be supervised using a sensor, and, in parallel, running an observer that does *not* incorporate the same sensor signal, but only measurements for the other states of the system (provided it is still observable like that). Divergence of the resulting estimate and the measured value indicates faults in the system which can then be treated accordingly. Using multiple sensors improves the operational safety even more and allows for further diagnosis, i.e. determining where the fault is located.

Aside from operational safety, monitoring non-measurable states can prolong the functional lifetime of devices and components, as was done in [21], where an observer is used to determine the temperature of a car seat heater based on conductive fibers. As too high temperatures can permanently damage the conductive layer of the polymer fibers, which still is far away from temperature ranges dangerous to the passenger, the controller needs to make sure not to exceed the temperature limits. A similar purpose is pursued in chapter 2, where the state of charge of Lithium-ion cells is determined using a novel kind of EKF. It needs to be monitored in all devices and vehicles powered by rechargable batteries in order to prolong the cells' lifetime and to avoid thermal runaway (catastrophic destruction due to inextinguishable fire), among other reasons.

[9] Low-pass filters without model-based prediction capabilities always introduce phase delay (at least if the filter is causal, i.e. implementable), which is usually undesirable, especially for very fast systems. Using a state observer instead can reduce this delay significantly while yielding the same denoising power, thanks to the model knowledge that can be incorporated.

[10] disturbance signals appearing in the same differential equation as the inputs (matched) or in other ones (unmatched).

[11] and also need to be *used* redundantly, which was not done by Boeing, directly leading to the tragic 737 MAX crashes [20]

2 Polynomial augmented extended Kalman filter to estimate the state of charge of Lithium-ion batteries

This chapter deals with adaptive estimation of the state of charge (SOC) of Lithium-ion cells, as they are used, for example, in traction batteries, medical technology or consumer electronics. Since such cells exhibit considerable voltage drops as a function of electrical current, SOC, and state of health (SOH – depending on age and number of cycles), it is not possible to infer the SOC from the cell voltage during operation, as it is possible (though to a limited extent) with some non-rechargeable battery types. However, if the cell has been at rest for many hours, it is possible to find a characteristic curve (often stored as a lookup table/LUT) for the cell that links the open circuit voltage (OCV) with the SOC (which is defined between 0 and 1). However, this curve results to be quite flat for common Li-ion cell chemistries, especially in the middle SOC range around 50%, which reduces the invertibility and, hence, the observability. This makes it necessary to estimate the SOC in the battery management system (BMS) with a suitable state observer. A common approach is to assign the initially measured cell voltage (right before current is drawn, upon activation of the device) to an initial condition (IC) for the SOC estimator by means of an inverse lookup table and, starting from this, to determine the amount of charge removed or added *time-integratively* by means of numeric integration of the measured current. Related to a nominal "capacity" (more precisely: nominal charge quantity, assumed to be known), an estimate of the current SOC can be obtained from this. This method, called Coulomb-counting, works very well if the nominal maximum charge and the intial SOC are exactly known and if the current measurement is accurate, but can suffer from drifting problems due to the integrating behaviour and from offsets due to the flatness of the characteristic curve. It also does not allow an exact SOC determination without frequent re-calibrations. Erroneous SOC estimation can lead to accidental deep discharge, overcharging or a limited usable energy storage capacity – in the worst case, it can even trigger catastrophic thermal runaways, though standard BMS' integrate multiple other failsafes against this. In this chapter, a more sophisticated method for SOC estimation is developed, based on an extended Kalman filter, which combines the Coulomb-counting method and an electrical resistive-capacitive (RC) equivalent circuit model (ECM) with an online adaptation of the polynomial curve fit of the characteristic OCV curve. This allows all over/undervoltages at the electrodes and in the electrolyte to be represented by two capacitor voltages, resulting in a more accurate, calibration-free SOC estimation, thus allowing safe and better utilization of the existing electrode material and better fault detection capabilities.

2.1 State of the art

Due to the fast ongoing technical development of electrical devices, the requirement for safe and durable energy storage becomes very important. Lithium-ion batteries (LIB) are frequently used for mobile or mains-independent devices, but perhaps most importantly, in electric vehicles. There, a necessary information for the driver is the battery's state of charge, as an indicator of the remaining range the vehicle can travel. It can also be used as part of a fault-detection system, together with other measurements, to ensure the passengers' safety. For example, in [22], SOC estimates are used for current or voltage sensor fault detection. There, SOC estimates obtained using Coulomb counting (which will be detailed later) are compared to SOC estimates given by a Kalman filter-based observer. In other applications, incorrect determination of the SOC can e.g. lead to data loss or impaired user experience.

In practice, it is not feasible to measure the SOC directly, so approximate methods must be employed. Various approaches are described in the literature for the estimation of the SOC of a battery: from the most empirical ones through data-based methods (e.g. using various neural network structures) to physical model-based methods. In general, empirical methods tend to be either too inaccurate or too complex, rendering it difficult to successfully apply them. From a technical point of view, observers based on electrical ECMs are a good compromise between complexity and estimation accuracy. In this chapter, the following classification is taken into consideration, not only in terms of review of the existing literature, but also to compare the results:

- Empirical methods, and
- Physical model based methods.

Table 2.1: Variables and example parameters

Symbol	Value	Description
$SOC(t)$	–	State of charge ($0 \leq SOC \leq 1$)
$OCV(SOC)$	–	Open circuit voltage (relaxed cell)
$u_T(t)$	–	Measurable terminal voltage
$i(t)$	–	Current (positive for discharging)
p_1	–	Polynomial slope-related coefficient
p_2	–	Polynomial "S"-related coefficient
p_3	–	Polynomial coefficient, $OCV(0.5)$
C_1	5.6 kF	Capacitance (fast electrode dynamics)
R_1	0.6 mΩ	Resistance (fast electrode dynamics)
C_2	54 kF	Capacitance (slow electrode dynamics)
R_2	8.2 mΩ	Resistance (slow electrode dynamics)
R_0	20 mΩ	Ohmic resistance of the cell
Q_{full}	9087 C	Maximum electric charge of the cell
T_s	100 ms	Fundamental sampling time

2.1.1 Empirical methods

The simplest empirical method is the measurement of the battery terminal voltage during operation and using it directly as an indicator of the remaining charge. Influences like temperature dependence, voltage drops caused by the current through internal ohmic resistances and aging effects render this method inaccurate [23]. While measurement of the OCV, and direct usage as remaining charge indicator, mitigates some of these issues, it constitutes a very tedious process due to the necessity of long relaxation phases and therefore is not suitable for online operation. Empirical methods based on impedance measurement for SOC estimation are described in [24]. There, information about the dynamics of the battery cell by means of impulse response is gained from the impedance measurement. However, this method is not suitable for real-time applications [24, 25]. In [26], impulse responses are also used, in combination with a LUT to estimate the SOC. Other examples of mapping a measurement to the SOC estimate through an empirical model can be found in [27], [28]. The so-called Coulomb counting method, including an integral of the current from and to the battery, is one of the most widely used methods to estimate the SOC of a battery. In situations where there are many dicontinuities in the measured current (e.g. from switching, which can happen in hybrid vehicles due to large dynamic transients in current and power demand upon gear shifts), numerical problems can arise and render this method inaccurate [29], [30]. However, Coulomb counting still yields very accurate SOC estimates in case of a known initial SOC, known battery parameters and accurate fast current measurements.

2.1.2 Physical model based methods

Many modern methods are based on a physical model. The battery model that is most prevalent in literature is an ECM that was first presented by Chen and Rincón-Mora in 2006 [31]. In general, observers based on ECMs like this offer similarly precise results as electrochemical model-based observers, but with less computational load. This ECM consists of a variable voltage source for the open circuit voltage, a resistor representing the internal ohmic cell resistance, and two RC branches for the fast and slow electrode dynamics in the positive branch of the circuit. Up to this point, it is identical to the well-known dual-polarization model [32]. The second part of the model proposed in [31] includes another RC network and a current source. It characterizes the SOC-OCV-relationship and the cell's SOH.

In fact, knowledge about the SOC-OCV characteristics is a crucial aspect for SOC estimation purposes if the observer depends on a known, certain SOC-OCV model. In the past, but also still in recent contributions, it is identified offline and then stored within LUTs [33,34] or in piecewise linear fits which interpolate the characteristics between sampling points [35]. Different polynomial parametrizations of the SOC-OCV characteristics are also very often used in order to refine the interpolation between samples. In [36–41], for instance, polynomials are used to approximate the SOC-OCV characteristics.

In [42] characteristic SOC-OCV curves at different temperatures are studied using the most commonly used mathematical basis function: exponential, polynomial, sum of thrigonometric functions, and also Gaussian model fitting. In all these considered bases the measurements are conducted with pulse test data.

Estimation of the SOC also depends on the accuracy of the ECM parameters. In the literature, different approaches are adopted to estimate these parameters:

- directly (within the state observer),
- indirectly (outside of the state observer), and each either
- online (adaptive), or
- offline (pre-identified).

Direct methods integrate the parameters to be estimated as random variables into the state observer, for example inside a KF, while indirect methods use a dedicated observer to provide for parameter estimation which are then passed to the state observer. In [33] for instance, a recursive least square method with forgetting factor is used to estimate the parameters, while a KF is used to estimate the states of the battery model. This cascaded method represents a quite common way to reduce the calculation load in case the time scales of the changes of states and parameters differ greatly. Nevertheless, the difficulty to tune the initial value of the covariance matrices together with the forgetting factor is often a reason to prefer other methods. Another classical indirect method is the well known dual KF which consists of two intertwined KFs: one of them dedicated to the estimation of the parameters and the other one dedicated to the estimation of the states. A recent example of the dual KF approach applied to LIBs can be found in [41] where a model based on an ECM with two RC branches is combined with an SOC-OCV relationship that is modelled as a standard 6th order polynomial with 7 coefficients. The state EKF estimates the SOC and the two RC voltages while the cascaded parameter EKF, operating on the assumption of much slower dynamics than the states (i.e. a multi-time-scale-approach), estimates all the physical parameters as well as all polynomial coefficients – 13 parameters in total. In [43], an adaptive KF algorithm is used to estimate the SOC of a LIB. To be more precise, a dual KF is adopted in order to estimate the parameters and to use the estimated parameters to adapt the KF which is dedicated to the estimation of the SOC. Additionally, the well-known Sage-Husa algorithm [44] is employed to adapt the process and measurement covariance matrices.

In general, drawbacks of indirect parameter estimation methods include the following aspects.

- Concerning the need to involve a model as virtual measurement, we can consider, for example, the case that some ohmic resistance R is to be estimated online from measurements of current

and voltage. Then, numerically unfavourable operations (i.e. division by a measured quantity) will occur in the model. Including the parameter estimation into the state observer, e.g. by introducing a physically meaningful differential equation based on Kirchhoff's laws that contains the ohmic resistance as a parameter, can numerically robustify the parameter estimation.

- A delay of at least one sampling time step, due to the cascaded intertwined architecture, has to be introduced in the estimation of either the states or the parameters in order to avoid algebraic loops. For slow systems like batteries, the impact of this issue is low.
- For a clear and expedient separation, or decoupling, of state and parameter estimation, their change rates should be different. This is oftentimes the case if the parameters are, in fact, physical parameters.

For relatively slow systems, where a sampling time of 0.1 seconds is sufficient, the reduction of the calculation load is not a strong motivation. Furthermore, as will be detailed in the following sections, in the case presented here, the parameters to be estimated have *no physical meaning*, but are more abstract. In particular, they are supposed to change in a time scale similar to that the states. In such cases, a single KF which directly estimates parameters and states together is to prefer, despite the calculation load.

The proposed contribution deals with an extended Kalman filter for SOC estimation of a LIB. The following points are taken into consideration:

- An EKF based on an ECM of the battery is derived in order to estimate the SOC.
- The SOC-OCV nonlinearity in the system output is modelled using a single high-order polynomial with a reduced number of adaptive (state-dependent) parameters, rendering the method suitable to cells with qualifying SOC-OCV curve forms.
- To achieve the adaptivity, coefficients of said polynomial are estimated by the augmented EKF, as well, while their ICs are pre-computed offline using a standard nonlinear Least Squares method from measured SOC-OCV data sets. These ICs must constitute an SOC-OCV decomposition that is roughly in the vicinity of the true (unknown) one in order to yield EKF convergence, but increased robustness is demonstrated.
- The polynomial parameters, estimated online by the EKF, change just as rapidly as the states (SOC and electrode overvoltages), in contrast to e.g. [41]. Even if the SOC-OCV evaluation using the adaptive parameters does not represent a curve that fits the uncertain initial SOC-OCV data points well, accurate SOC estimates can be obtained.
- Two variants of the EKF from the same family (same polynomial SOC-OCV model) are compared, where one includes a supplementary state representing the terminal voltage, which yields better results, also compared to the existing literature. Thanks to an increased *stochastization* of the output model representation inside the EKF, the bias effect in the estimation results can be reduced.

The chapter is organized as follows. In Section 2.2 the model of the battery is described. Section 2.3 and 2.4 show the polynomial EKF with the output nonlinearity and the state nonlinearity variant, respectively. In Section 2.5 the structure of the experimental setup is discussed. Finally, Section 2.6 presents the measured results together with a comparison with similar observer structures, followed by some conclusions.

2.2 Modeling

The SOC is defined as the fraction of the full charge that is currently available in a battery. It is necessary to specify the sign of the current for charging and discharging; the most common practice is to assign

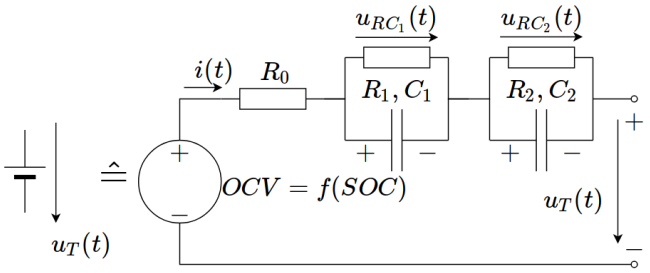


Figure 2.1: Dual polarization-like model with two RC circuits representing fast and slow electrode dynamics and a variable, SOC-dependent voltage source as OCV.

positive currents to the process of discharging and use negative currents for charging. Consequently, the SOC is

$$SOC(t) = SOC(t_0) - \frac{\int_{t_0}^{t} \eta i(\tau) d\tau}{Q_{full}}. \tag{2.1}$$

Here, $\eta = \eta(i, SOC, SOH, t, T, ...)$ is a usually unknown coefficient quantifying the (dis-)charging efficiency and accomodating other uncertainties when charging or discharging and Q_{full} is the full amount of electrical charge that can be stored in the cell. The terminal voltage is modelled subject to electrical and electrochemical influences, which heavily depend on the direction and the magnitude of the current, and can be expressed as, see e.g. equation 3.20 at p. 52 of [45],

$$u_T(i, SOC) = OCV(SOC) \overbrace{-u_A(i, SOC) + u_C(i, SOC)}^{\text{Anode and cathode overpotentials}} - \underbrace{\Delta\varphi_M(i)}_{\text{Ion losses}} - R_0 i. \tag{2.2}$$

The first summand, $OCV(SOC)$, is the open circuit (no load) cell voltage in a relaxed state. The anode and cathode exhibit a current-dependent effect that manifests as spatial voltage drops, called overpotentials or overvoltages, indicated by $u_{A,C}$ for the anode and cathode, respectively. Ohmic losses can be distinguished into two categories, namely the ion losses in the electrolyte and further ohmic resistances. For the observer design, however, a simpler model based on ECM is employed. In particular, part of the model from [31] is combined with a polynomial fit of measured $OCV(SOC)$ data to represent the output nonlinearity. The complete Li-ion single cell model that is used in this contribution can be seen in Fig 2.1.

Remark 2.1 *The model parameters R_0, C_1, R_1, C_2, R_2 and Q_{full} are assumed to be constant in the model and were determined iteratively by hand in extensive simulation studies, using the equivalent circuit model as shown in Fig. 1 and comparing with measured currents and voltages. The assumption of constant parameters is not a strong one, since it will be relaxed using the system model uncertainty matrix (process noise)* $\mathbf{Q}$ *in the EKF design.*

The output of the system is the battery terminal voltage

$$u_T(t) = OCV(SOC) - u_{RC_1}(t) - u_{RC_2}(t) - R_0 i(t). \tag{2.3}$$

$$\dot{u}_{RC_1}(t) = \frac{i(t)}{C_1} - \frac{u_{RC_1}(t)}{R_1 C_1}, \; \dot{u}_{RC_2}(t) = \frac{i(t)}{C_2} - \frac{u_{RC_2}(t)}{R_2 C_2} \tag{2.4}$$

represent the fast (RC1) and slow (RC2) electrode dynamics. In state-space form, the system dynamics can be described as

$$\mathbf{x}(t) = \begin{bmatrix} SOC(t) & u_{RC_1}(t) & u_{RC_2}(t) \end{bmatrix}^T, \tag{2.5}$$

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}i(t) \tag{2.6}$$

with the simplification $\eta = 1$ and system input $i(t)$ and

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\frac{1}{R_1 C_1} & 0 \\ 0 & 0 & -\frac{1}{R_2 C_2} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} -\frac{1}{Q_{full}} \\ \frac{1}{C_1} \\ \frac{1}{C_2} \end{bmatrix}. \tag{2.7}$$

The output, however, is nonlinear due to $OCV(SOC)$:

$$u_T(t) = OCV(SOC) - u_{RC_1}(t) - u_{RC_2}(t) - R_0 i(t). \tag{2.8}$$

Remark 2.2 *If the true $OCV(SOC)$ is a known relationship with $\frac{\mathrm{d}OCV(SOC)}{\mathrm{d}SOC} \neq 0$ for $0 \leq SOC \leq 1$, and if $R_1 C_1 \neq R_2 C_2$, the system described by this model is observable [46].*

In the contribution at hand, relationship $OCV(SOC)$ is modelled using a polynomial fit of order 12. Instead of realizing all 13 independent coefficients of the curve fit, a custom function is introduced

$$OCV = (2.5p_1(SOC - 0.5) + 1)\left(p_2(SOC - 0.5)^{11} + p_3\right). \tag{2.9}$$

An example can be seen in Fig. 2.4a, where three SOC-OCV data sets (consisting of 15, 22 and 25 data points, respectively) for three cells are shown together with curve fits according to Eq. (2.9).

Remark 2.3 *The SOC-OCV data sets can be obtained with several different tedious measurement procedures. In our case, data set 3 was measured with an alternation of discharging and relaxation preriods: First, the cell was charged with the standard CC-CV strategy up to 4.2 V until virtually no further charging current was drawn. Then, using Coulomb counting technique, varying but known amounts of charge (mAh) were drawn from the cell with an iMAX B6 charger/discharger. Between the discharging periods, the cell was allowed to relax for 30 minutes. Data sets 1 and 2, on the other hand, correspond to other cells. They are included in this work, as well, to demonstrate a certain degree of robustness of the estimator against uncertainty in $p_{1,2,3}(0)$ – since these three numbers are the only information about the SOC-OCV-relationship that is available to the observer.*

Note that data sets 1 and 2 are quite similar. In fact, the initial polynomial curve fit coefficients $p_{1,2,3}(0)$ (computed using Matlab's nonlinear least squares function `lsqcurvefit`) for data sets 1 and 2 are almost identical. Data set 3 corresponds to significantly different initial polynomial curve fit coefficients, see Table 2.2.

Remark 2.4 *Apart from the low number of coefficients to be optimized, an additional advantage of a fit like Eq. (2.9) is the intuitive interpretability of said coefficients: $2.5p_1$ is the slope of the curve in the middle range at 50% SOC, while p_2 influences the steepness near 0% and 100% SOC. Note that the given odd exponent, 11, within the second parenthesis, is necessary to represent the saddle-shaped curve and can be changed to other odd numbers to match the cell more closely, affecting the width of the linear region for median SOC values: Larger exponents correspond to wider regions of near-linear behavior. Finally, p_3 is the voltage at exactly 50% SOC. This specific function is able to describe*

1. *SOC-OCV characteristics with a certain degree of rotational symmetry around 50% SOC (data sets 1 and 2), if the given interpretations of the parameters are to hold,*
2. *SOC-OCV characteristics where the slope reduces drastically in the left half-plane but then retains a positive, yet small value in the rest of the field (like data set 3).*

In the latter case, however, the parameters are harder to interpret.

Though most Lithium ion cells fall into one of the mentioned categories [47], a single polynomial with such a low number of parameters is still a rather restrictive limitation, but this is relaxed by *adapting the parameters online*. Hereafter, an integrated combined estimation and adaption algorithm based on EKF is presented.

Table 2.2: Initial polynomial curve fit coefficients

ICs for data set 1	ICs for data set 2	ICs for data set 3
$p_1(0) = 0.0380584$	$p_1(0) = 0.0359893$	$p_1(0) = 0.0586138$
$p_2(0) = 1404.69$	$p_2(0) = 1398.47$	$p_2(0) = 736.048$
$p_3(0) = 3.39572$	$p_3(0) = 3.38153$	$p_3(0) = 3.53325$

2.3 Polynomial EKF: Output nonlinearity variant

The curve fit parameters $p_{1,2,3}$ are now considered as states to be estimated, which allows for a robustification of the SOC-estimation against the uncertain SOC-OCV relationship.

Remark 2.5 *As a consequence of this state augmentation, the observability proof from [46] mentioned before does not hold for the resulting augmented system. However, it will be shown by experiment that the SOC estimate is stable, even though estimation error convergence for the other states is not guaranteed. In particular, the uncertain SOC-OCV model (represented by the polynomial coefficients) cannot be identified. But in practice, estimates for $u_{RC1,2}$ and the polynomial parameters $p_{1,2,3}$ are not required (being abstract and conceptional in nature, anyway), as long as they constitute a decomposition of the terminal voltage dependency on the SOC. By allowing for variation of the coefficients $p_{1,2,3}$ within one (dis-)charging process, i.e. relatively fast changes similar to that the system states, the SOC estimate itself can become more robust against SOC-OCV uncertainty and also adaptive to changes.*

The EKF design is based upon a state-space model of the system, so the following states are introduced:

$$\mathbf{z}(t) = \begin{bmatrix} SOC(t) & u_{RC_1}(t) & u_{RC_2}(t) & p_1 & p_2 & p_3 \end{bmatrix}^T, \tag{2.10}$$

where polynomial coefficients $p_{1,2,3}$ are modelled as uncertain constants *for now*. In general, dynamics of a nonlinear system in state-space form can be expressed as field $\mathbf{g}$

$$\dot{\mathbf{z}}(t) = \mathbf{g}(\mathbf{z}(t), \mathbf{u}(t), t) \tag{2.11}$$

with states $\mathbf{z}(t)$ and inputs $\mathbf{u}(t)$. However, just like in Eq. (2.6), the internal dynamics are linear in this case, as well:

$$\dot{\mathbf{z}}(t) = \begin{bmatrix} \mathbf{A} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix} \mathbf{z}(t) + \begin{bmatrix} \mathbf{B} \\ \mathbf{0}_{3\times1} \end{bmatrix} \underbrace{\mathbf{u}(t)}_{i(t)}, \tag{2.12}$$

while the system output function $u_T = h(\mathbf{z})$ still is

$$h(\mathbf{z}) = OCV(SOC(t)) - u_{RC_1}(t) - u_{RC_2}(t) - R_0 i(t). \tag{2.13}$$

Note that function $OCV(SOC(t))$ is the polynomial parametrization (2.9) with coefficients $p_{1,2,3}$ and is, naturally, evaluated using the estimated states and parameters. Implementation of the EKF algorithm in a microcontroller calls for discrete-time equations, so the system introduced above is discretized using an explicit Euler scheme with sampling time $T_s = 0.1\,\mathrm{s}$, i.e.

$$\dot{\mathbf{z}}(t) \approx \frac{\mathbf{z}(k) - \mathbf{z}(k-1)}{T_s} = \mathbf{g}(\mathbf{z}(k-1), \mathbf{u}(k-1)). \tag{2.14}$$

A so-called a-priori estimate $\hat{\mathbf{z}}^-(k)$ of the states (prediction step of the EKF algorithm) is obtained by solving Eq. (2.14) for $\hat{\mathbf{z}}^-(k)$, where $\hat{\mathbf{z}}^-(k)$ substitutes $\mathbf{z}(k)$ and the last step's a-posteriori estimate $\hat{\mathbf{z}}^+(k-1)$ substitutes $\mathbf{z}(k-1)$:

$$\hat{\mathbf{z}}^-(k) = T_s \mathbf{g}(\hat{\mathbf{z}}^+(k-1), \mathbf{u}(k-1)) + \hat{\mathbf{z}}^+(k-1), \tag{2.15}$$

or in this specific case

$$\hat{\mathbf{z}}^-(k) = \begin{bmatrix} SOC(k-1) - T_s \frac{i(k-1)}{Q_{full}} \\ u_{RC1}(k-1) + T_s \left(\frac{i(k-1)}{C1} - \frac{u_{RC1}(k-1)}{R1C1} \right) \\ u_{RC2}(k-1) + T_s \left(\frac{i(k-1)}{C2} - \frac{u_{RC2}(k-1)}{R2C2} \right) \\ \begin{bmatrix} p_1(k-1) & p_2(k-1) & p_3(k-1) \end{bmatrix}^T \end{bmatrix}. \tag{2.16}$$

Like mentioned above, the states at time $(k-1)$ are in fact the a-posteriori estimates from the last iteration $\hat{\mathbf{z}}^+(k-1)$, e.g. $SOC(k-1)$ stands for $\hat{z}_1^+(k-1)$, just to aid comprehensibility and traceability of the derivation. Since the internal system dynamics is linear in this variant, the Jacobian is not state-dependent. It is defined as

$$\mathbf{J}(k) = \begin{bmatrix} \frac{\partial \hat{z}_1^-(k)}{\partial \hat{z}_1^+(k-1)} & \frac{\partial \hat{z}_1^-(k)}{\partial \hat{z}_2^+(k-1)} & \cdots & \frac{\partial \hat{z}_1^-(k)}{\partial \hat{z}_n^+(k-1)} \\ \frac{\partial \hat{z}_2^-(k)}{\partial \hat{z}_1^+(k-1)} & \frac{\partial \hat{z}_2^-(k)}{\partial \hat{z}_2^+(k-1)} & \cdots & \frac{\partial \hat{z}_2^-(k)}{\partial \hat{z}_n^+(k-1)} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \hat{z}_n^-(k)}{\partial \hat{z}_1^+(k-1)} & \frac{\partial \hat{z}_n^-(k)}{\partial \hat{z}_2^+(k-1)} & \cdots & \frac{\partial \hat{z}_n^-(k)}{\partial \hat{z}_n^+(k-1)} \end{bmatrix}, \tag{2.17}$$

$$\mathbf{J}(k) = \mathbf{J} = \operatorname{diag}\left(1, 1 - \frac{T_s}{R_1 C_1}, 1 - \frac{T_s}{R_2 C_2}, 1, 1, 1\right). \tag{2.18}$$

The Jacobian of the output, on the other hand, is a complicated function of the states. It is defined as

$$\mathbf{h}(k) = \begin{bmatrix} \frac{\partial h(\hat{\mathbf{z}}^-(k))}{\partial \hat{z}_1^-(k)} & \frac{\partial h(\hat{\mathbf{z}}^-(k))}{\partial \hat{z}_2^-(k)} & \cdots & \frac{\partial h(\hat{\mathbf{z}}^-(k))}{\partial \hat{z}_n^-(k)} \end{bmatrix}, \tag{2.19}$$

resulting in this specific case as

$$\mathbf{h}(k) = \begin{bmatrix} h_1(\hat{\mathbf{z}}^-) \\ -1 \\ -1 \\ (2.5\hat{z}_1^- - 1.25)(\hat{z}_5^-(\hat{z}_1^- - 0.5)^{11} + \hat{z}_6^-) \\ (\hat{z}_1^- - 0.5)^{11}(1 + (2.5\hat{z}_1^- - 1.25)\hat{z}_4^-) \\ 1 + \hat{z}_4^-(2.5\hat{z}_1^- - 1.25) \end{bmatrix}^T \tag{2.20}$$

where discrete time arguments (k) were omitted, with

$$h_1(\hat{\mathbf{z}}^-) = 2.5\hat{z}_4^-(\hat{z}_5^-(\hat{z}_1^- - 0.5)^{11} + \hat{z}_6^-) + 11\hat{z}_5^-(\hat{z}_1^- - 0.5)^{10}(1 + \hat{z}_4^-(2.5\hat{z}_1^- - 1.25)). \tag{2.21}$$

The output Jacobian is needed for the Taylor-like linearization approach upon which the EKF theory is based [7]. An a-priori estimate of the covariance results as

$$\mathbf{P}^-(k) = \mathbf{J}\mathbf{P}^+(k-1)\mathbf{J}^T + \mathbf{Q}, \tag{2.22}$$

where $\mathbf{Q} \in \mathbb{R}^{6\times 6}$, all elements ≥ 0, is used as a tuning matrix quantifying the system uncertainty. Greater numbers on the principal diagonal signify less trust in the corresponding states' modelled system dynamics. This is useful for states $z_{4,5,6}$, which are the polynomial parameters that were modelled as stochastic constants. Off-diagonal elements are not as easy to interpret, they correspond to the system covariance between states. Or, described in a more intuitive way, when considering $\mathbf{Q}$ a tuning parameter instead of a system property, they effectively weight the influence which the measured output has (mapped through the output Jacobian to the state assigned to the specific column of $\mathbf{Q}$) on the estimated state represented in the specific row of $\mathbf{Q}$. In particular, for this EKF variant,

$$\mathbf{Q} = \operatorname{diag}(2 \times 10^{-3}, 10^2, 10^2, 10^{-2}, 0, 10^{-1}), \tag{2.23}$$

i.e. off-diagonal elements equal zero. In the first time step, $\mathbf{P}^+(0)$ is a user-supplied initial value for the covariance matrix. Since there is only one measured output that is used in the update step of the EKF algorithm, the denominator in the following Kalman gain expression is scalar, allowing for this concise notation:

$$\mathbf{k}(k) = \frac{\mathbf{P}^-(k)\mathbf{h}(k)^T}{\mathbf{h}(k)\mathbf{P}^-(k)\mathbf{h}(k)^T + r}. \tag{2.24}$$

Here, $r = 0.1$ is a parameter quantifying measurement uncertainty related to the measured battery terminal voltage u_T. Using the Kalman gain, an a-posteriori estimation of states and covariance is obtained (update step of the EKF algorithm):

$$\hat{\mathbf{z}}^+(k) = \hat{\mathbf{z}}^-(k) + \mathbf{k}(k)\left(u_T(k) - h(\hat{\mathbf{z}}^-(k))\right), \tag{2.25}$$

$$\mathbf{P}^+(k) = \left(\mathbf{I}_{6\times 6} - \mathbf{k}(k)\mathbf{h}(k)\right)\mathbf{P}^-(k). \tag{2.26}$$

Again, in the first time step the algorithm requires user-specified ICs, in particular $\hat{z}_1^+(0) = SOC(0)$. In implementation, this initial SOC could be roughly approximated using an inverted OCV-SOC LUT, fed with the measured terminal voltage $u_T(0)$ plus an approximated cell voltage drop $R_0 i(0)$ using the measured current, as indicated in Fig. 2.2a. However, in this contribution, the initial SOC is assumed to be unknown and hardcoded to 0.5 in the Kalman filter implementation. This necessitates a large initial state covariance $P_{1,1}^+(0)$ as discussed in section 2.6. The other ICs are chosen as $\hat{z}_2^+(0) = 0$, $\hat{z}_3^+(0) = 0$, $\hat{z}_4^+(0) = p_{01}$, $\hat{z}_5^+(0) = p_{02}$ and $\hat{z}_6^+(0) = p_{03}$. As mentioned before, the last three ICs, for the polynomial coefficients, are calculated offline using given measurements for OCV. An overview of the EKF algorithm can be obtained from Fig. 2.2a where the dashed lines represent data transfer through iterations. The initial conditions for this variant are

$$\begin{aligned}\hat{\mathbf{z}}^+(0) &= \begin{bmatrix}0.5 & 0 & 0 & 0.0381 & 1404.7 & 3.3957\end{bmatrix}^T \\ \mathbf{P}^+(0) &= \mathrm{diag}(10^6, 1, 1, 10^3, 10^3, 10^3).\end{aligned} \tag{2.27}$$

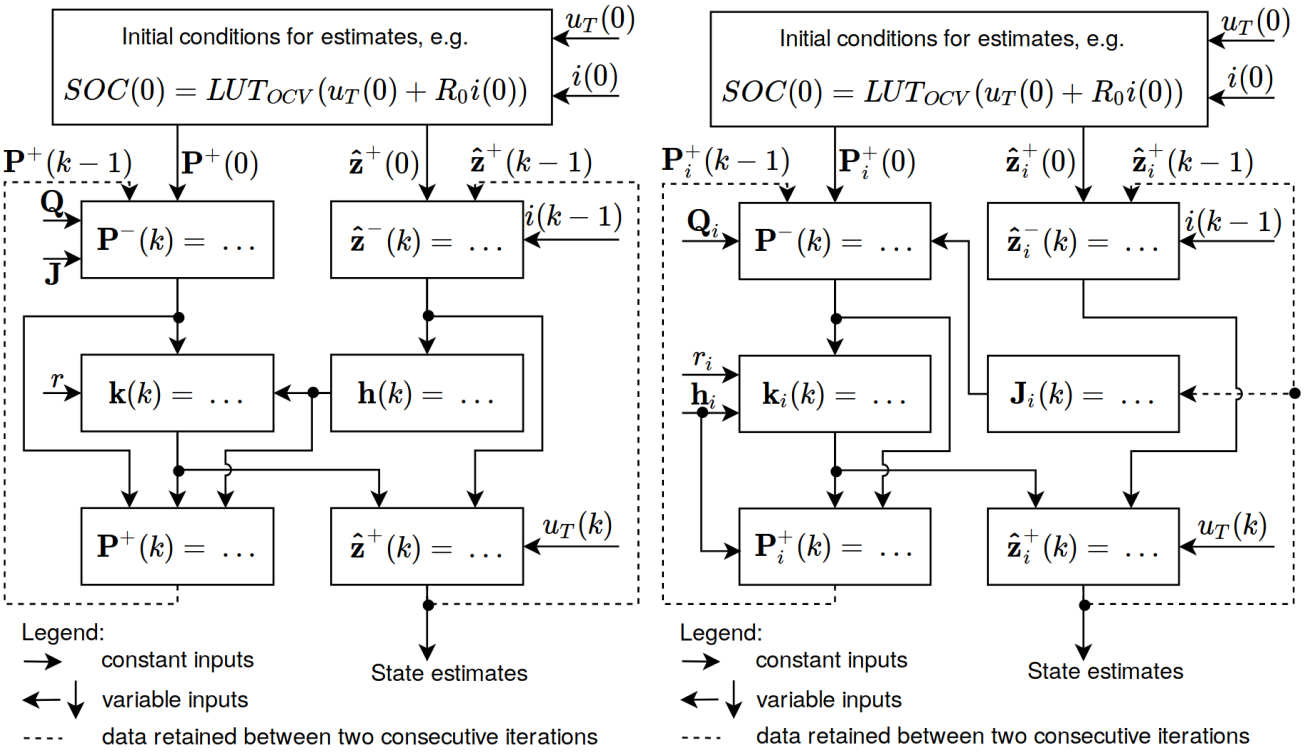
2.4 Polynomial EKF: State nonlinearity variant

In order to investigate a possible improvement of the estimation, terminal voltage $u_T(t)$ can be considered as an additional state.

Remark 2.6 *This additional state represents a redundancy since the polynomial parameters $p_{1,2,3}$ already represent most of the uncertain part of the nonlinear output model, i.e. the SOC-OCV relationship. An analogy to this concept of redundancy can be observed in [48], where orthonormal bases are extended to frames in form of additional redundant bases, achieving improved results of mathematical decompositions. Since this will yield infinite permutations of decompositions, that do not necessarily hold physical meaning any longer, some kind of optimization using a cost function is necessary to handle the power that this concept provides – in our case, this is the Kalman filter algorithm, which is a covariance minimizer and has the sole goal of estimating the SOC.*

The conducted state augmentation using u_T implies treatment of Eq. (2.8) as a stochastic relationship instead of a trusted equation like in Eq. (2.13) and facilitates a more targeted tuning influence (compared to the measurement uncertainty r) by adding new uncertainty tuning parameters, represented by the corresponding new elements in matrix $\mathbf{Q}$. The state vector of the EKF base model is extended accordingly:

$$\mathbf{z}_i^T = \begin{bmatrix}SOC & u_{RC_1} & u_{RC_2} & p_1 & p_2 & p_3 & u_T\end{bmatrix}, \tag{2.28}$$



(a) EKF algorithm: Output nonlinearity variant

(b) EKF algorithm: Internal nonlinearity variant

Figure 2.2: Variants of the EKF algorithm.

where time arguments have been omitted. Since an expression of the dynamics of u_T is required for the state-space model, a usually straight-forward approach would be to take its derivative, using Eq. (2.9), yielding

$$\dot{u}_T(t) = \frac{\mathrm{d}}{\mathrm{d}t}\left[OCV(SOC(t)) - u_{RC_1}(t) - u_{RC_2}(t) - R_0 i(t)\right] \tag{2.29}$$

with the derivative

$$\frac{\mathrm{d}}{\mathrm{d}t}OCV(SOC(t)) = \frac{\mathrm{d}OCV(SOC)}{\mathrm{d}SOC}\frac{\mathrm{d}SOC}{\mathrm{d}t} = \frac{-j_1(\mathbf{z}_i)i(t)}{Q_r}, \tag{2.30}$$

where $j_1 = \frac{\mathrm{d}OCV(SOC)}{\mathrm{d}SOC}$ is similar to h_1 from Eq. (2.21), and continuous time arguments (t) are omitted for better readability:

$$j_1(\mathbf{z}_i) = 2.5 z_{i,4}(z_{i,5}(z_{i,1} - 0.5)^{11} + z_{i,6}) + 11 z_{i,5}(z_{i,1} - 0.5)^{10}(1 + z_{i,4}(2.5 z_{i,1} - 1.25)). \tag{2.31}$$

In the literature, a common approximation when calculating $\dot{u}_T(t)$ is $\frac{\mathrm{d}i(t)}{\mathrm{d}t} \approx 0$ with the quite restricting rationale that only relatively slow changes occur in the current, at least compared to the other system dynamics, see e.g. [49]. This approach would lead to

$$\dot{u}_T(t) = \frac{-j_1(\mathbf{z}_i)i(t)}{Q_r} - \dot{u}_{RC_1}(t) - \dot{u}_{RC_2}(t). \tag{2.32}$$

To obtain the a-priori state estimate from this, an explicit Euler discretization can be applied, yielding

$$\hat{\mathbf{z}}_i^-(k) = \begin{bmatrix} SOC(k-1) - T_s\frac{i(k-1)}{Q_{full}} \\ u_{RC1}(k-1) + T_s\left(\frac{i(k-1)}{C1} - \frac{u_{RC1}(k-1)}{R1C1}\right) \\ u_{RC2}(k-1) + T_s\left(\frac{i(k-1)}{C2} - \frac{u_{RC2}(k-1)}{R2C2}\right) \\ \begin{bmatrix} p_1(k-1) & p_2(k-1) & p_3(k-1) \end{bmatrix}^T \\ \hat{z}_{i,7}^-(k) \end{bmatrix} \tag{2.33}$$

where, again, the states at time $(k-1)$ are in fact the a-posteriori estimates from the last iteration $\hat{\mathbf{z}}_i^+(k-1)$, and with

$$\hat{z}_{i,7}^-(k) = u_T(k-1) + T_s i(k-1)\left(\frac{C_1+C_2}{C_1C_2} - \frac{j_1(\hat{\mathbf{z}}_i^+(k-1))}{Q_r}\right) - T_s\left(\frac{u_{RC1}(k-1)}{R_1C_1} + \frac{u_{RC2}(k-1)}{R_2C_2}\right). \quad (2.34)$$

Using this approach, however, the modelled terminal voltage no longer depends on the internal cell resistance R_0, due to the employed approximation $\frac{\mathrm{d}i(t)}{\mathrm{d}t} \approx 0$. It is desirable to obtain an expression where R_0 is present because the reason to include u_T into the state vector to be estimated is the *stochastization* of this important parameter. A somewhat unorthodox alternative to using a discretization of the terminal voltage dynamics, avoiding said approximation, is to formulate the terminal voltage model directly in discrete time, depending on last iteration's a-posteriori estimates:

$$\hat{z}_{i,7}^-(k) = OCV(SOC(k-1)) - u_{RC1}(k-1) - u_{RC2}(k-1) - R_0 i(k-1). \quad (2.35)$$

Note that, again, function $OCV(SOC(k-1))$ stems from the polynomial parametrization (2.9) with coefficients $p_{1,2,3}(k-1)$. While not obviously corresponding to a well-defined state dynamics $\dot{u}_T(t)$, this approach has the advantage of retaining the system input $i(t)$, together with the highly uncertain (since time-varying and SOH-dependent) parameter R_0. Since it is possible in this variant to assign uncertainty to the terminal voltage model, the estimation (if well-tuned) results more robust to errors in the underlying SOC-OCV data. Using this, the now time-variant Jacobian results as

$$\mathbf{J}_i(k) = \begin{bmatrix} \multicolumn{6}{c}{\mathbf{J}} & \mathbf{0}_{6\times 1} \\ \begin{bmatrix} j_1(\hat{\mathbf{z}}_i^+(k-1)) & -1 & -1 & 0 & 0 & 0 \end{bmatrix} & & & & & & 0 \end{bmatrix} \quad (2.36)$$

where j_1 from Eq. (2.31) is evaluated at the last iteration's a-posteriori state estimates. The output Jacobian simplifies to

$$\mathbf{h}_i = \begin{bmatrix} \mathbf{0}_{1\times 6} & 1 \end{bmatrix} \quad (2.37)$$

while the rest of the filter equations stay almost the same:

$$\mathbf{P}_i^-(k) = \mathbf{J}_i(k)\mathbf{P}_i^+(k-1)\mathbf{J}_i(k)^T + \mathbf{Q}_i \quad (2.38)$$

with $r_i = 0.1$ and

$$\mathbf{Q}_i = \begin{bmatrix} 10^{-3} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10^{-2} & 0 & 0 & 10 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10^{-1} & 10^3 \\ 0 & 0 & 0 & 10 & 0 & 10^3 & 1 \end{bmatrix}. \quad (2.39)$$

Note that the non-zero off-diagonal elements in the last row and column have shown to be important in order to allow for enough innovation in the polynomial parameters, in particular p_1 which is proportional to the mid-section slope, and the 50%-SOC-voltage p_3.

$$\mathbf{k}_i(k) = \frac{\mathbf{P}_i^-(k)\mathbf{h}_i^T}{\mathbf{h}_i\mathbf{P}_i^-(k)\mathbf{h}_i^T + r_i}, \quad (2.40)$$

$$\hat{\mathbf{z}}_i^+(k) = \hat{\mathbf{z}}_i^-(k) + \mathbf{k}_i(k)\left(u_T(k) - \hat{z}_{i,7}^-(k)\right), \quad (2.41)$$

$$\mathbf{P}_i^+(k) = \left(\mathbf{I}_{7\times 7} - \mathbf{k}_i(k)\mathbf{h}_i\right)\mathbf{P}_i^-(k). \quad (2.42)$$

The modified EKF algorithm with the nonlinearity transformed into the internal system equations can be retraced in Fig. 2.2b. The initial conditions for this variant are

$$\begin{aligned} \hat{\mathbf{z}}_i^+(0) &= \begin{bmatrix} 0.5 & 0 & 0 & 0.0381 & 1404.7 & 3.3957 & u_T(0) \end{bmatrix}^T \\ \mathbf{P}_i^+(0) &= \mathrm{diag}(10^9, 1, 1, 10^3, 10^3, 10^3, 10^9). \end{aligned} \quad (2.43)$$

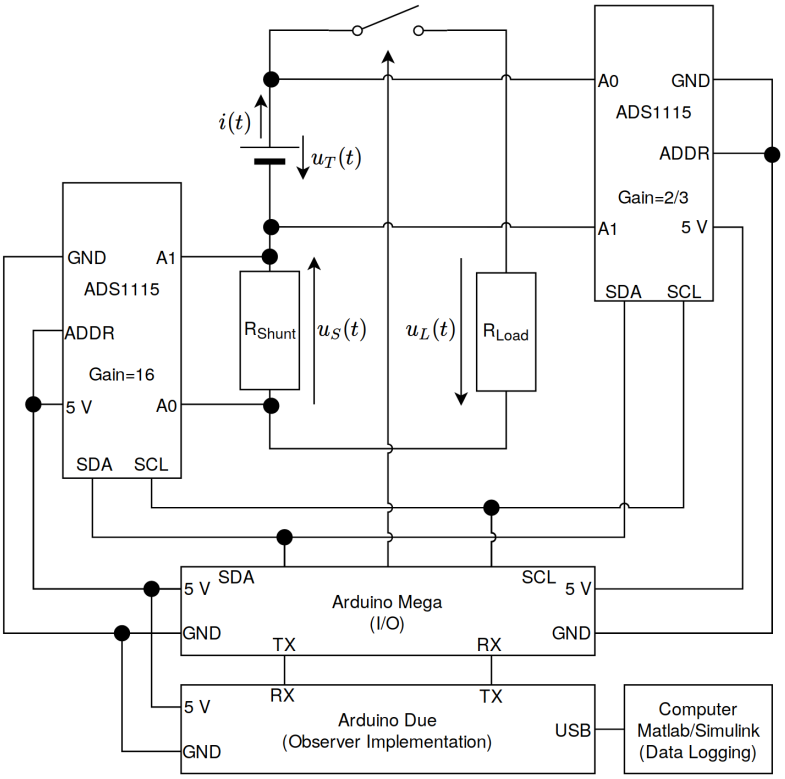
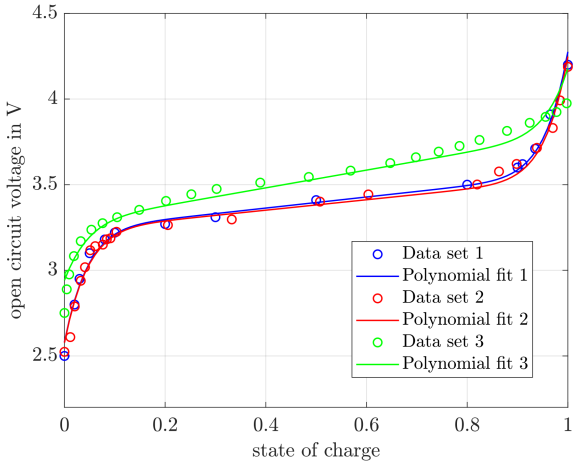


Figure 2.3: Measurement setup using two I^{2}C 16 bit differential ADCs (TI ADS1115) with programmable gain. $R_{Shunt} = 50\,\mathrm{m\Omega}$ (0.5%, $< 30\,\mathrm{ppm/K}$, 4T sensing) and $R_{Load} = 2.2\,\Omega$ (10 W, 5%).

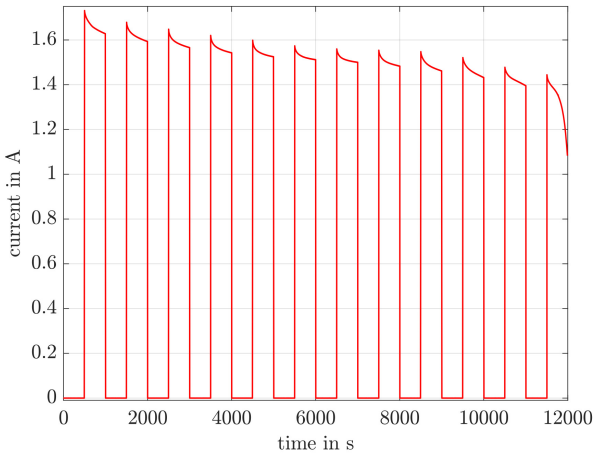
2.5 Experimental setup

In Fig. 2.3 the measurement setup can be examined. Due to as of yet unresolved I^{2}C compatibility issues, the analog-to-digital converters (ADCs) were not connected to the Arduino Due directly but to an additional Arduino Mega, which subsequently handles all I/O tasks and sends them to the Due over the serial interface. (The serial interface is also used for real-time synchronization of the two microcontrollers and the PC, which is equipped with Matlab, and receives all the data for logging.) The EKF presented in Section 2.3, the output nonlinear variant, and that described in Section 2.4, the state nonlinearity variant, were implemented on the Arduino Due with the given parameters and initial conditions. The Lithium ion cell that was used in the experiment is a standard Samsung ICR18650-26F cell. The discharging mode is characterized by a constant load resistance $R_{Load} = 2.2\,\Omega$ and a pulsed current between 0 (relay open) and max. 1.7 A (relay closed) with a duty cycle of 0.5 and a period of 1000 s, see Fig. 2.4b.

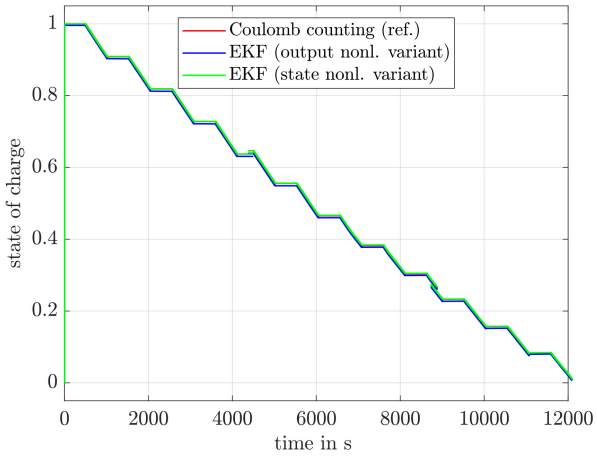
Remark 2.7 *This simple load profile is similar to the Hybrid Pulse Power Characterization (HPPC) test profile which is used in the literature [36], [43]. HPPC is used to identify various model parameters, but not the SOC, which is tested using realistic driving cycles in the cited works. In the contribution at hand, however, parameters and states are estimated together and online, so separate experiments are not suitable. While using a realistic driving cycle test profile would certainly be a practice-relevant experiment, a more challenging estimation task is posed to the EKF by testing with a pulsed on-off load profile. This is due to the phases of smooth or even constant signals, where, in general, EKFs can exhibit a phenomenon commonly described as "falling asleep" or "dropping off" due to lack of information/innovation. The load steps, including theoretically infinite frequencies, represent the most dynamic case, where stability and robustness of the observer algorithm come into play.*



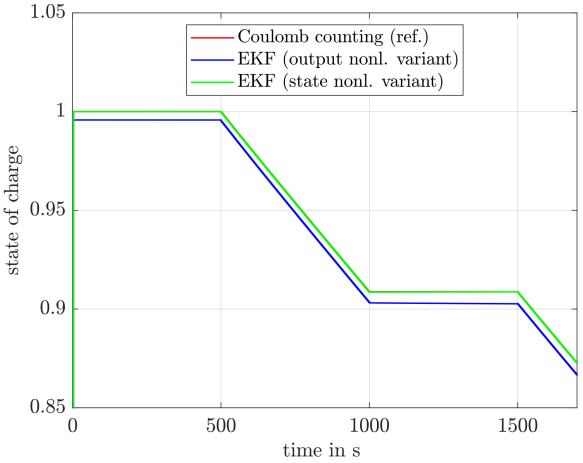
(a) SOC-OCV data sets and their curve fits.



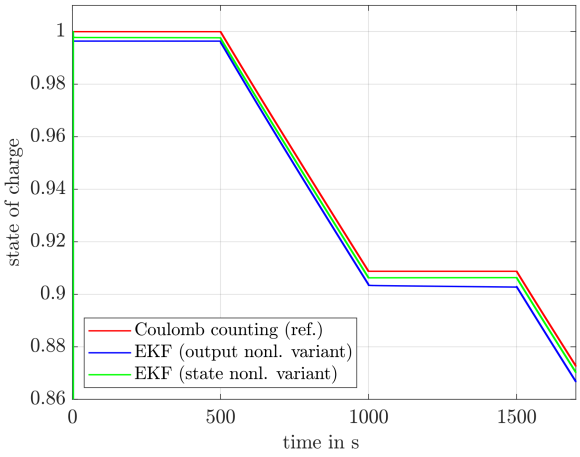
(b) Measured current i for the experiment.



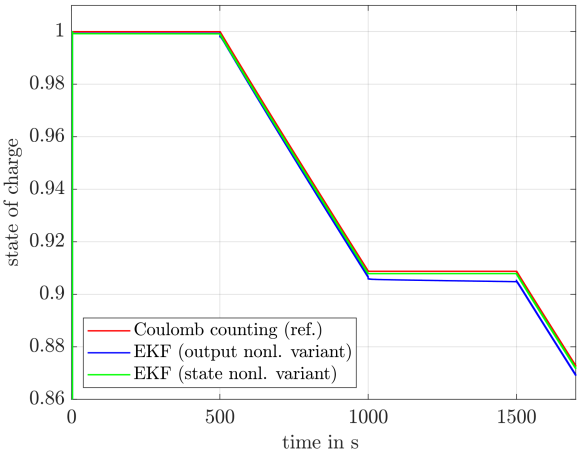
(c) Resulting estimates for the state of charge (data set 1).



(d) SOC estimates in detail: red and green lines overlap (data set 1).

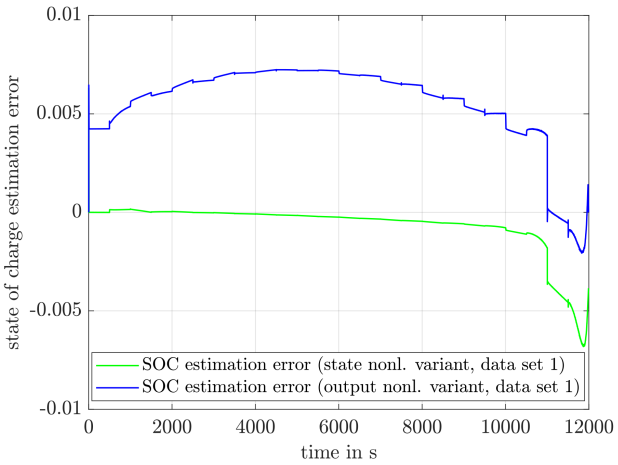


(e) SOC estimates in detail: red and green lines overlap (data set 2).

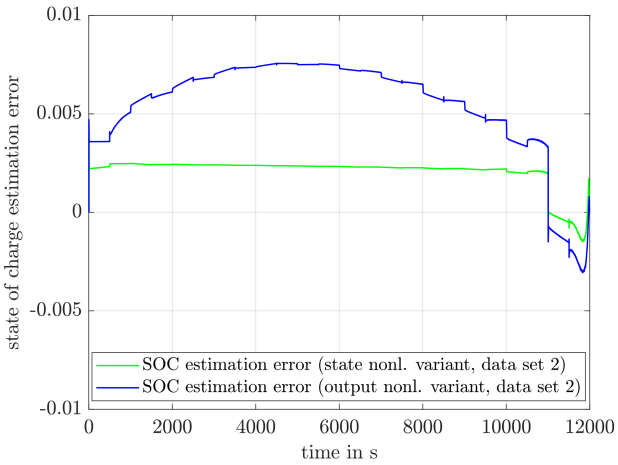


(f) SOC estimates in detail: red and green lines overlap (data set 3).

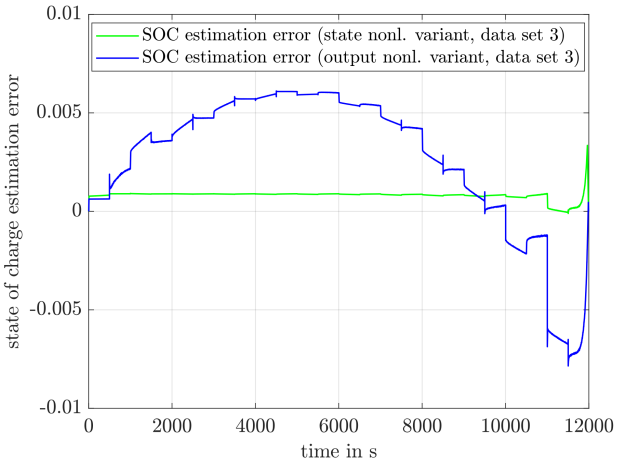
Figure 2.4: Results obtained from the experiment.



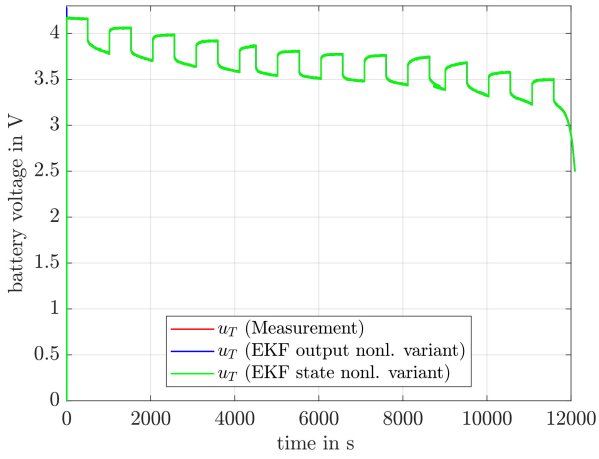
(a) SOC estimation error (data set 1).



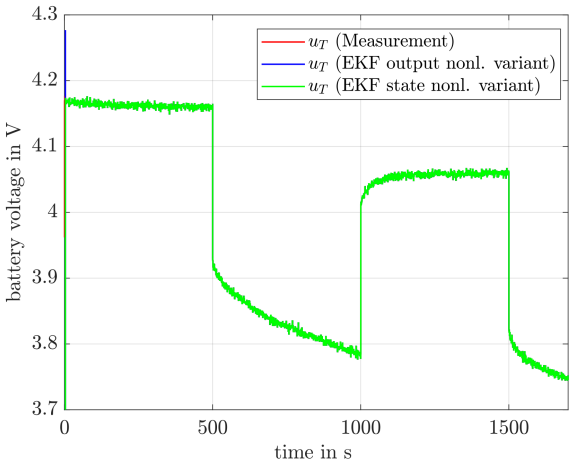
(b) SOC estimation error (data set 2).



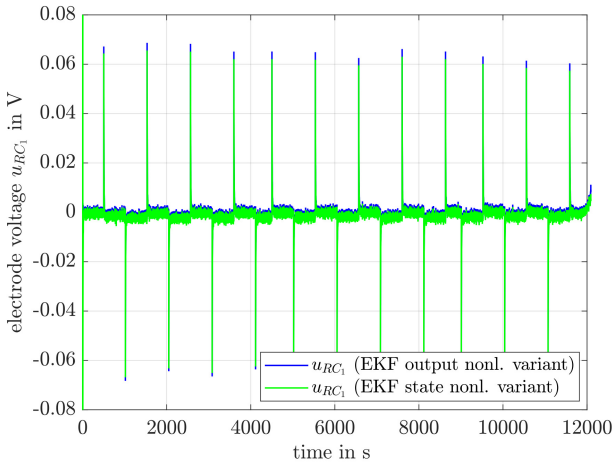
(c) SOC estimation error (data set 3).



(d) Estimates for the output voltage u_T (data set 1, visually identical for 2 and 3).

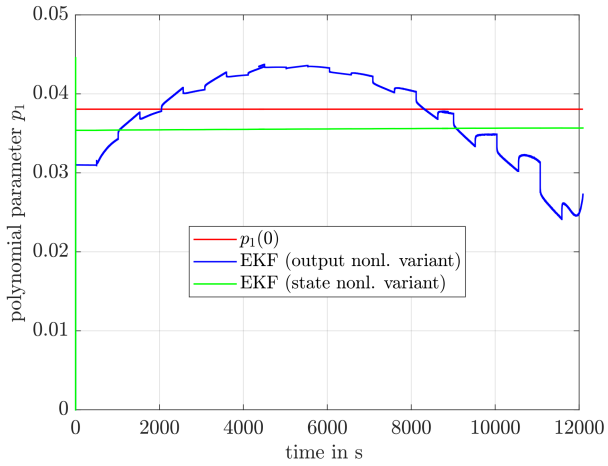


(e) Estimates for the output voltage u_T (detail). Note that all graphs overlap.

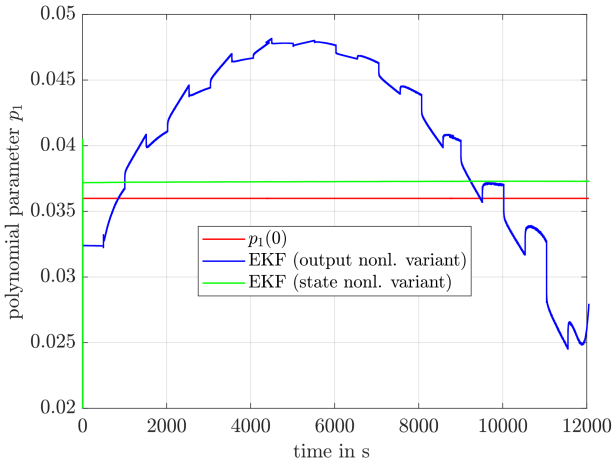


(f) Estimates for electrode dynamics voltage u_{RC_1} for data set 1. Note that it is virtually identical for data sets 2 and 3.

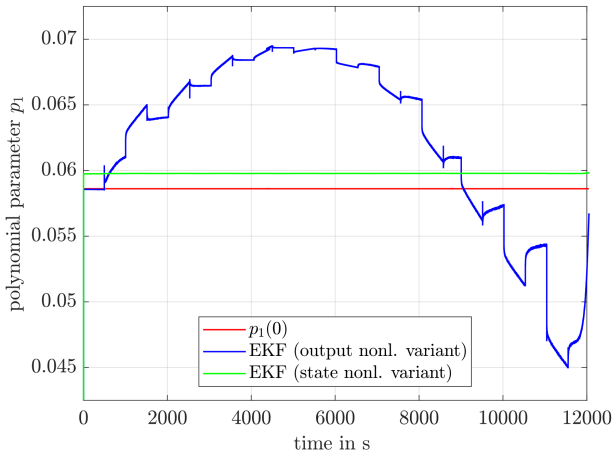
Figure 2.5: Results obtained from the experiment.



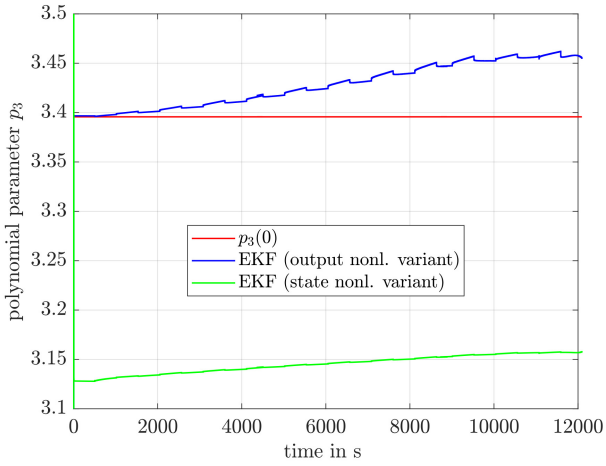
(a) Estimates for polynomial parameter p_1 (data set 1).



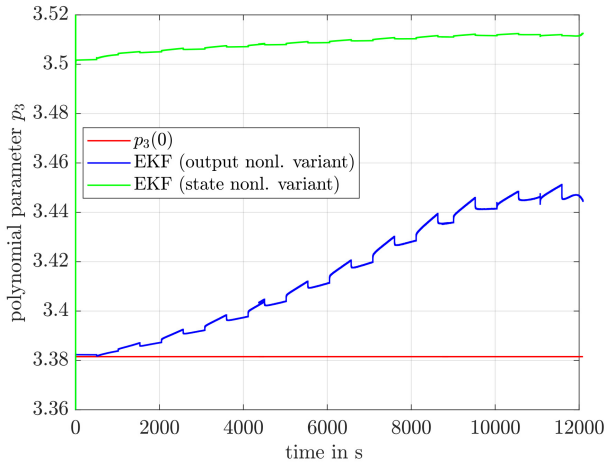
(b) Estimates for polynomial parameter p_1 (data set 2).



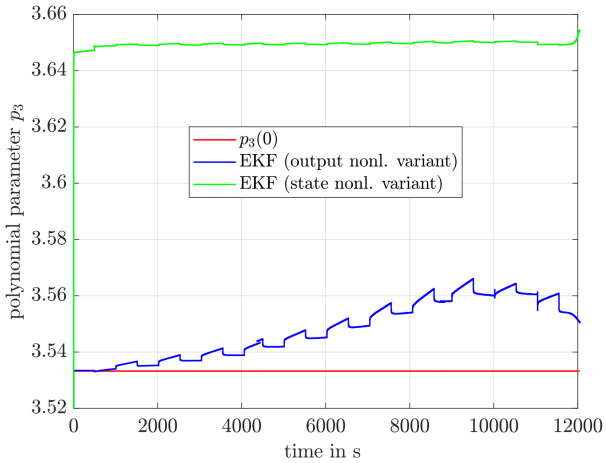
(c) Estimates for polynomial parameter p_1 (data set 3).



(d) Estimates for polynomial parameter p_3 (data set 1).

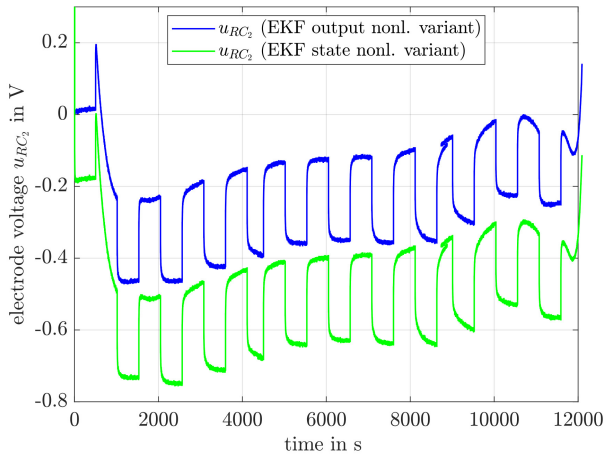


(e) Estimates for polynomial parameter p_3 (data set 2).

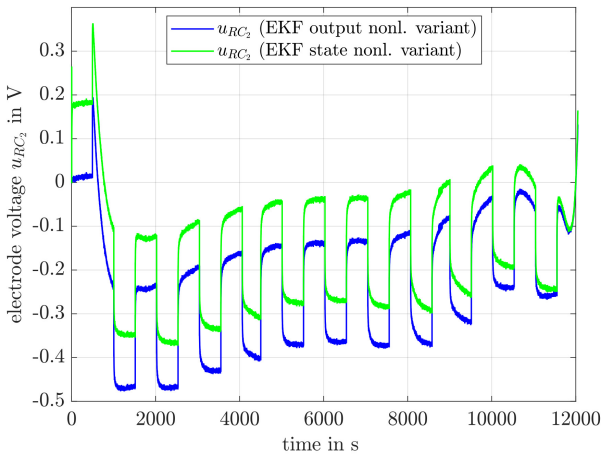


(f) Estimates for polynomial parameter p_3 (data set 3).

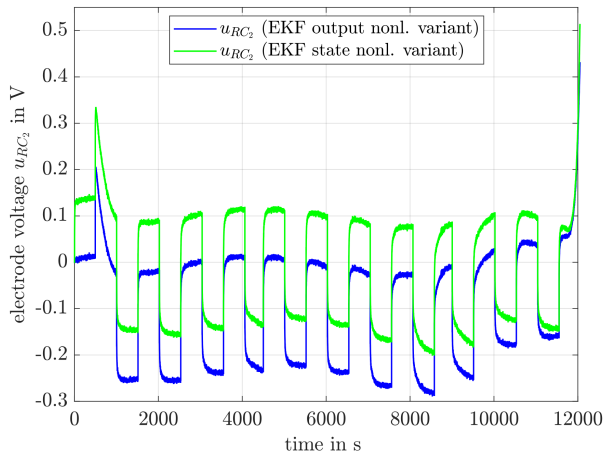
Figure 2.6: Results obtained from the experiment.



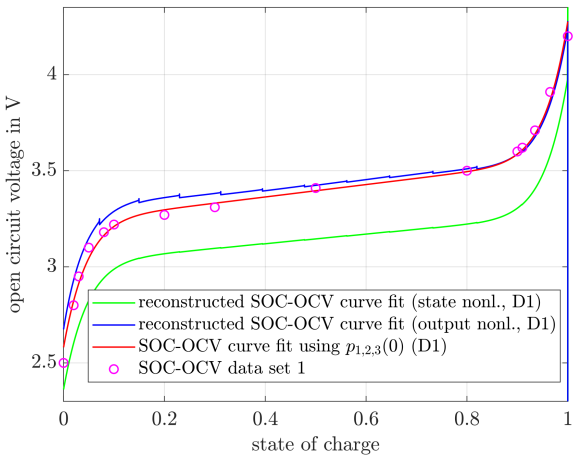
(a) Estimates for electrode dynamics voltage u_{RC_2} (data set 1).



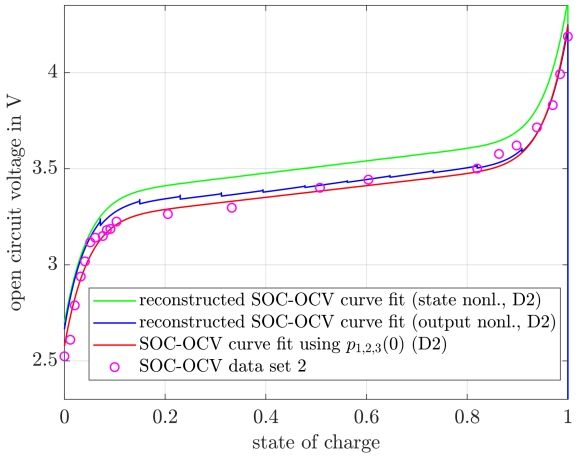
(b) Estimates for electrode dynamics voltage u_{RC_2} (data set 2).



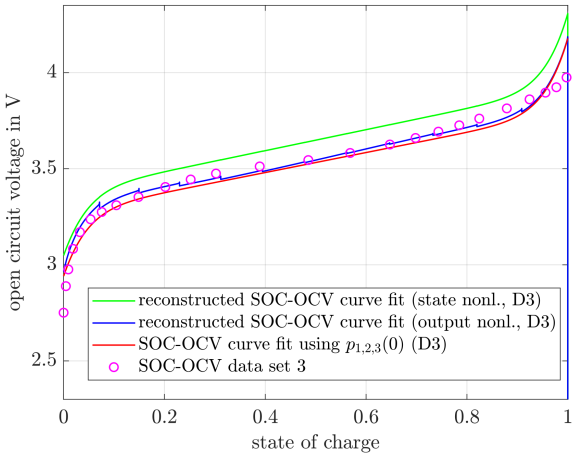
(c) Estimates for electrode dynamics voltage u_{RC_2} (data set 3).



(d) Reconstructed curve fits (2.9) using the EKF-estimated $p_{1,2,3}$ for both variants (IC set 1).



(e) Reconstructed curve fits (2.9) using the EKF-estimated $p_{1,2,3}$ for both variants (IC set 2).



(f) Reconstructed curve fits (2.9) using the EKF-estimated $p_{1,2,3}$ for both variants (IC set 3).

Figure 2.7: Results obtained from the experiment.

2.6 Experimental results and comparisons

In the literature, EKFs for estimation of the SOC have been reported many times, e.g. in [36], where the state vector contains the SOC and two polarization overpotentials. This is equivalent to the physical base model shown in Eqs. (2.5-2.6) of the contribution at hand. In [37], the same model is used, but with a fractional order. Compared to these works, this one extends the state vector by three parameters of the polynomial output model (2.9), as described in section 2.3, and also by the terminal voltage as described in section 2.4. Thanks to this extension, an improved SOC estimation will be demonstrated.

2.6.1 Description of the results

For the experiment, three sets of initial estimation conditions for polynomial coefficients $p_{1,2,3}(0)$ were tested for each EKF variant in order to demonstrate its robustness against output model uncertainties. These ICs, shown in Table 2.2, correspond to the three SOC-OCV data sets. Again, these ICs, together with the initial covariance matrix elements, are the only information about the real SOC-OCV-characteristics that are incorporated into the EKF algorithms. Thus, they require offline pre-identification. If they do not correspond to a relatively close curve fit, *the EKF does not converge.* (The EKF estimation problem in general is not convex, so neither convergence nor optimality are ever guaranteed.) While the values differ quite a bit between the sets shown in Table 2.2, they each still constitute a good decomposition in terms of a curve fit of similar SOC-OCV characteristics. The SOC estimates are shown in Fig. 2.4c and in more detail in Figs. 2.4d-2.4f. Figs. 2.5a-2.5c depict the SOC estimation error. It can be seen that both EKF variants perform very well, especially the state nonlinearity variant. In Table 2.3 a comparison of both EKF variants with each other is given, and also with an adaptive EKF reported in the literature [43]. It can be seen that, using several common error metrics like the integral of the absolute error (IAE), mean value, maximum absolute error, and variance, both introduced EKFs yield promising results. As reference signal, the Coulomb Counting method, see Eq. (2.1), is used together with $SOC(0) = 1$ for the fully charged cell at the beginning. The point of reference for 0% SOC arises when computing the total amount of charge dispensed during the experiment, namely Q_{full}. The output nonlinearity variant of the EKF has a small, potentially negligible residual SOC estimation offset and is unable to compensate it for the duration of the experiment. The state nonlinearity variant, however, matches the reference SOC more closely. Note that both Kalman filter variants operate on the assumption of $SOC(0) = 0.5$, but thanks to large $\mathbf{P}^+(0)$ matrices this initial error can be corrected quickly using measurement innovation. In fact, the observers reconstruct the measured terminal voltage u_T almost perfectly using the estimated states, see Figs. 2.5d and 2.5e where all signals visually overlap completely. The estimation results for u_T look identical for all three sets of initial conditions corresponding to the three data sets, so only one is shown. Concerning the comparison between the two Kalman filters, it is possible to observe some interesting properties from Figs. 2.6a-2.6c, 2.6d-2.6f and 2.7a-2.7c. Fig. 2.6a shows estimates of polynomial parameter p_1, which is proportional to the slope of the SOC-OCV curve in the middle region around 50% SOC. While neither Kalman filter retains the initial values of the polynomial parameters, the state nonlinearity variant stays close and has little variation, but the output nonlinearity variant shows distinct fluctuation and influences of the measured current and voltage. Clearly, parameter p_1 is dominant for low and high SOC, i.e. near 0 or 1 at the start and beginning of the experiment where the variation of estimated p_1 is more pronounced than in the middle, which is due to multiplication of p_1 with $(SOC - 0.5)$ in model (2.9). p_3 on the other hand is the only remaining influence at an SOC of 0.5, which shows in Figs. 2.6d-2.6f. Here, the most variation occurs in the middle of the experiment, when the cell is at about half capacity.

2.6.2 Interpretation of the results

In Figs. 2.7d-2.7f, the polynomial SOC-OCV curve fit from Eq. (2.9) is reconstructed using the parameters $p_{1,2,3}$ that are estimated online by the EKFs. These curve fits are not used anywhere in the observer algorithm but they are shown to demonstrate an interesting property. They are compared

to a curve fit that is evaluated using the ICs $p_{1,2,3}(0)$, stemming from the offline pre-identification. The "smaller" EKF variant (terminal voltage not included in the state vector but only in the output equation) tends to stick closer to the fit reconstructed using constant values than the larger variant.

Remark 2.8 *The larger difference for the state nonlinearity variant can be explained by the redundancy that was introduced by augmenting the output u_T as a state into the EKF. This redundancy affects the output equation (3) for u_T. As can be seen in Figs. 2.7a and 2.7d, the difference is compensated for within the estimation of state u_{RC2}. Metaphorically speaking, the "line of least resistance" for the information flow inside the state nonlinearity variant of the EKF algorithm is somewhat ambiguous because of this redundance. This effect leads to loss (or reduction) of physical interpretability of each participating element of the decomposition, but combined they still constitute a correct terminal voltage decomposition, however with better SOC estimation performance. As we demonstrate, using the presented strategy of aggressively adapting $p_{1,2,3}$, knowledge of a static SOC-OCV relationship is not as crucial for an accurate SOC estimation as it may seem, which in our opinion is an interesting result.*

Table 2.3: Benchmarks

	ICs from data set 1	ICs from data set 2	ICs from data set 3
state nonl.	IAE = 8.3672 VAR = 1.7682×10^{-6} MEAN = -6.7484×10^{-4} MAX = 0.0068	IAE = 25.9530 VAR = 6.3472×10^{-7} MEAN = 0.0021 MAX = 0.0046	IAE = 9.7195 VAR = 5.3548×10^{-8} MEAN = 8.1031×10^{-4} MAX = 0.0033
output nonl.	IAE = 67.9906 VAR = 4.5719×10^{-6} MEAN = 0.0055 MAX = 0.0072	IAE = 68.6260 VAR = 6.2699×10^{-6} MEAN = 0.0054 MAX = 0.0076	IAE = 47.3463 VAR = 1.2644×10^{-5} MEAN = 0.0027 MAX = 0.0078
from [43]	VAR = 5.75×10^{-5} MEAN = 0.0106 MAX = 0.0254		

This interesting effect is also linked to the weakened observability of the augmented system model (the one with u_T versus the one without) and depends non-trivially on uncertainty matrices $\mathbf{P}, \mathbf{Q}$. Regarding this aspect, the output nonlinearity variant of the EKF turns out to have a wider convergence region in terms of the initial covariance of the SOC, $P^+_{1,1}(0)$, compared to the state nonlinearity variant. The two initial covariance convergence regions are characterized by the same upper bound, but by different lower bounds. Using the state nonlinearity EKF variant, a 10^3 times larger initial covariance value is needed, compared to the lower bound belonging to the output nonlinearity variant. The reason of that can be heuristically explained considering that the level of observability reduces if the number of states increases. In general, the required minimum initial covariance is non-trivially related to the number of states to be estimated. The more states, the higher the required initial covariance. Intuitively speaking, there is a larger inertia to be overcome, the SOC being the state with the most inertia in terms of observability. This can be seen as a disadvantage of the state nonlinearity variant. On the other hand, a drawback of the output nonlinearity variant consists of a more pronounced bias in the estimation. This effect can be explained intuitively, considering that the output nonlinearity variant trusts the nonlinear model Eq. (2.13) in the a-posteriori state estimation step of the EKF algorithm and thus errors or uncertainties in this part of the model (e.g. in the internal cell resistance R_0) are directly visible in the estimation in form of a bias. On the other hand, for the state nonlinearity variant, the nonlinearity is moved inside an extended stochastic dynamics of the KF structure and thus can include the uncertainties of the output model inside the extended process noise covariance matrix $\mathbf{Q}$.

This can reduce the bias effect in the estimation results. In [40] an EKF described in form of the output nonlinearity variant is implemented and tested in which the OCV is approximated using a polynomial function of the 9^{th} order. The results are compared with a nonlinear estimator which outperforms the described EKF. However, no details are given on the utilized EKF, especially regarding delicate tuning parameters like the initial covariance. As we show in this contribution, a sensible tuning of r as well as matrices $\mathbf{Q}$, $\mathbf{P}$ can results in satisfactory EKF performance.

3 An extended Kalman filter for time delays inspired by a fractional order model

In this chapter, different variants of an EKF are compared, with the goal to estimate the variable time delay between two noisy signals – online and adaptively. A prerequisite for the observers developed for this purpose is that the signals are periodic with a period duration that is greater than the time delay to be estimated. The model used in the observer design is not an exact description of the time delay, but a rough approximation (attained by early truncation of its Taylor series), which practically results as a PT1 system (1st order low pass). This approximation is robust, yet too imprecise to be used as-is. Hence, in addition to the variable time delay or "PT1 time constant" T, a variable exponent α is introduced as an additional augmented state, representing the *non-integer order* of the low-pass filter.

For this purpose, methods from the field of *fractional calculus* (FC) are used, in particular the definition of the non-integer derivative according to Grünwald and Letnikov, since it not based on fractional integral expressions which are difficult to implement. Compared to integer derivatives, which operate *locally* on the signal[1], the fractional derivative has a memory that theoretically goes back infinitely far, so an infinitely large memory would actually be required for correct digital implementation. This problem was bypassed by the introduction of another approximation in the time-discrete Grünwald-Letnikov derivative definition. The resulting delay system, apart from the augmented states T and $\alpha \geq 1$ (which now no longer corresponds to the system order), always is of order 1, thanks to said approximations, which effectively avoids the aforementioned infinite-memory problem and results in a significantly improved time delay estimate (compared to the PT1-based observer).

This chapter is organized in the following way. Section 3.1 starts with a short literature survey about fractional methods in modelling and control, followed by an explanation of the practical motivation to design such an observer. In Section 3.3 the model of the considered abstracted time delay system is taken into account, followed by the first order approximation. Next, Section 3.5 is devoted to the description of the fractional model structure of the delay and the resulting EKF variants. In Section 3.6 the simulated results are shown and the conclusions close the chapter.

3.1 Fractional calculus in modelling and control

The area of FC emerged at the same time as the classical differential calculus and it deals with derivatives and integrals to an arbitrary order (real or even complex order), [50–53]. However, its inherent complexity postponed the application of the associated concepts. Nowadays, the FC theory is applied in many fields in science and engineering after recognizing its ability to yield a superior modelling and control in many dynamical systems, [50, 53, 54]. Very recent works like [55] pointed out advantages in terms of precision and optimality in control systems through the use of fractional PI controllers. In the literature, we can find several different definitions for the fractional integration and differentiation of arbitrary order, [50, 51, 53]. One of the most well-known definitions is given by the Grünwald-Letnikov approach. In [56], a study of the *differential flatness* property of linear time-invariant fractional systems is proposed. A framework of polynomial matrices of the fractional derivative operator is given together with a characterization of fractionally flat outputs and a simple algorithm to compute them. In terms of observation, recently some contributions related to fractional order system have appeared. In [57], linear matrix inequalities are proposed to ensure the stability of a class of uncertain fractional-order

[1] In discrete-time, for example, the first derivative of a signal can be approximated using the current and the last sample, whereas the second derivative additionally requires the sample right before the last one, etc.

linear systems by means of a fractional-order deterministic observer. In [58], by introducing a continuous frequency-distributed equivalent model and using an indirect Lyapunov approach, the sufficient condition for asymptotic stability of the full-order observer error dynamics is presented. It is notable that the stabilization problem of fractional order nonlinear systems is even more difficult [59] than that of integer systems, as shown in [60, 61].

As can be seen, fractional methods are an active field of research within modelling and control engineering. The presented approach, however – modelling the elemental time delay phenomenon using fractional methods and build observers on this basis – has not been described before in the literature.

3.2 Motivation

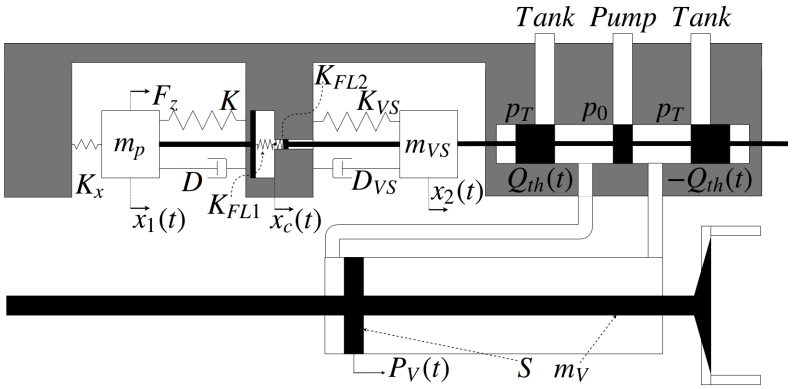


Figure 3.1: Model of the whole actuator (with a linearized version of the oil pressure dynamics). From [62].

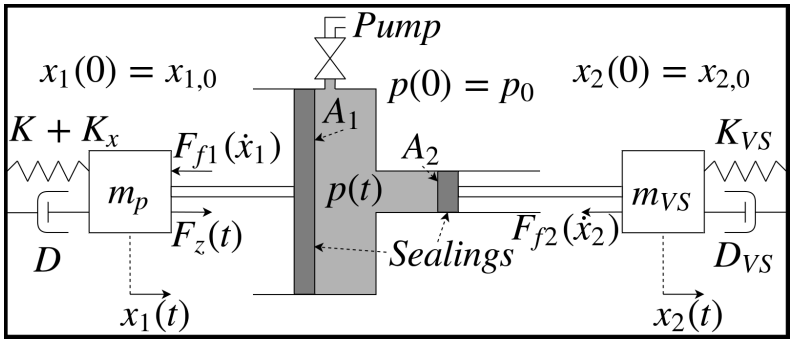


Figure 3.2: Hydraulic displacement amplifier (the middle part is clamped). Friction hysteresis is caused by the sealings. From [62].

In systems with time delays, it can be advantageous to know these delays in order to design compensators. This is especially true when a pre-defined or periodic desired trajectory must be tracked, since it can simply be antedated by a certain amount of time and fed into a feedforward action. This chapter proposes an EKF to estimate *variable* time delays online, based on (or rather, inspired by) FC methods. The motivation for this isolated, specific estimation task was a problem that occurred during the control design for a fully variable valve control within internal combustion engines by means of a piezo-hydraulic multistage actuator (which is shown in Fig. 3.1 and is also treated, in more detail, in chapter 4). In test-bench experiments with this actuator, using simple PD control on the position with additional feedforward action, the engine valve position could already be controlled sufficiently well – with the exception of an engine speed-dependent time delay behavior, caused by static friction hysteresis at the seals of a hydraulic displacement amplifier that is part of the multistage actuator. It is depicted

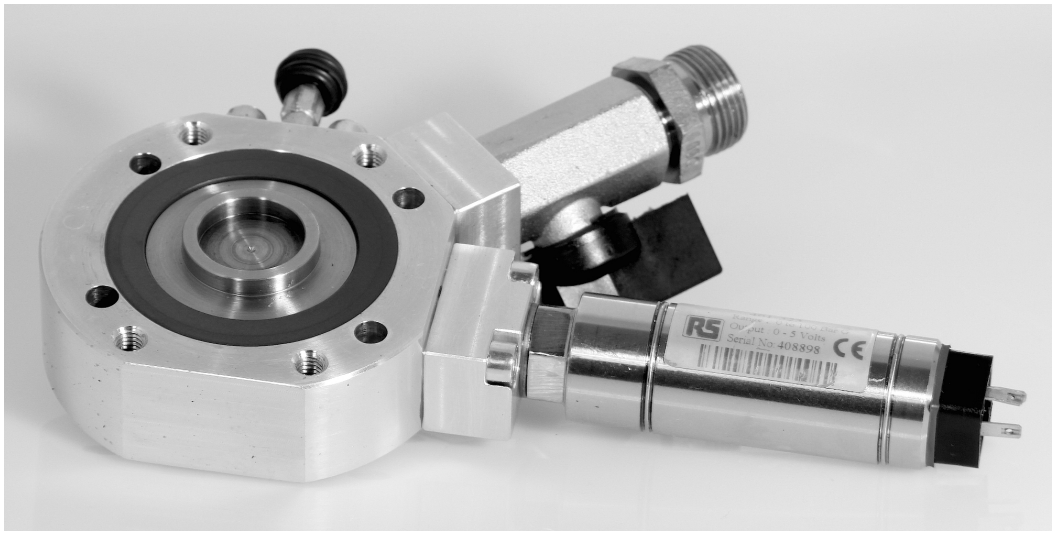


Figure 3.3: Hydraulic displacement amplifier with installed pressure sensor. Source: Ostfalia University of Applied Sciences, Faculty of Automotive Engineering. N. Werner, G. Bergholz and U. Becker.

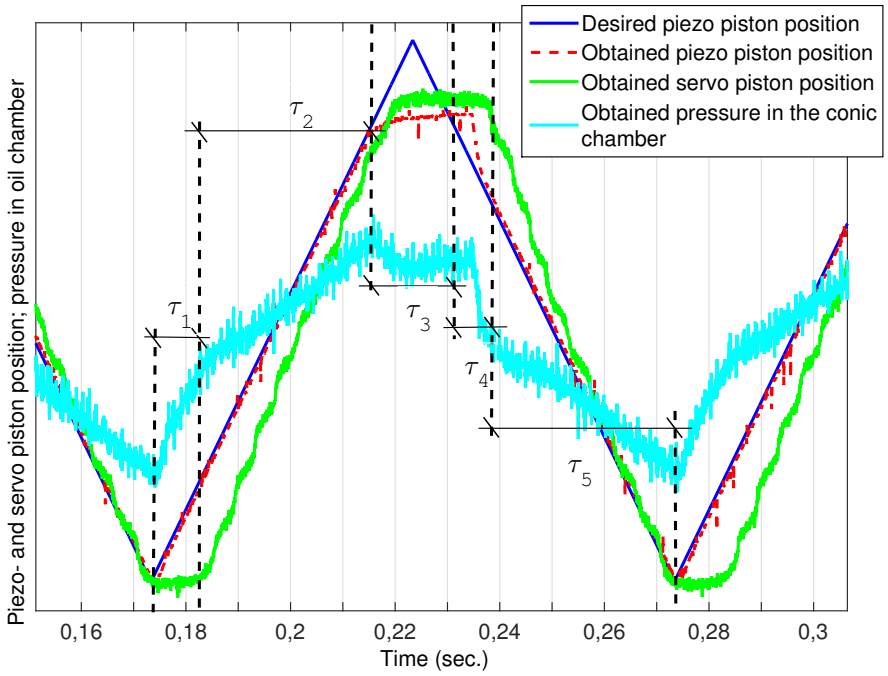
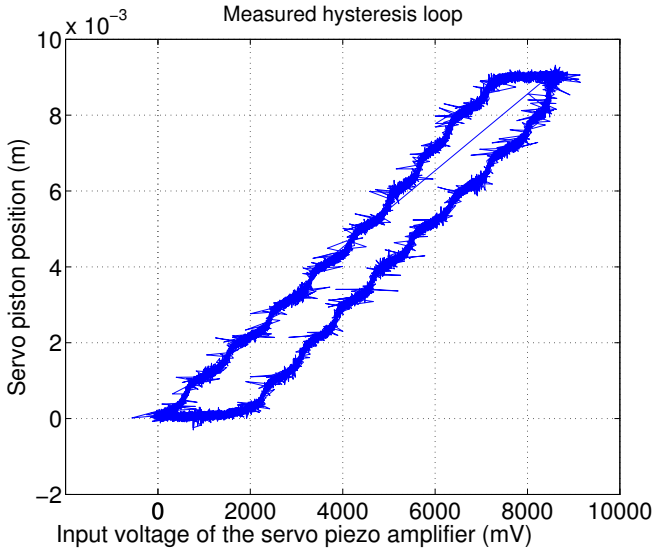
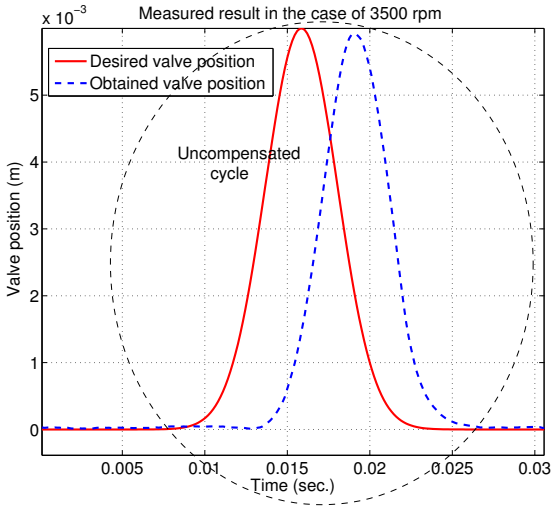


Figure 3.4: Consequences of static friction effects, caused by the sealings of the hydraulic displacement amplifier that is part of the actuator. From [62].

in Figs. 3.2 (schematic) and 3.3 (picture). This hysteresis effect can be retraced in Figs. 3.4, where the pressure inside the oil chamber (cyan line) increases for a duration τ_1 after the input piston moves (red line) and before the output piston moves (green line). As soon as the chamber volume increases again, the pressure increase is less pronounced (τ_2) and stays constant after that (τ_3). Naturally, this behavior also happens for movements in the other direction (τ_4 and τ_5). In order to compensate for the resulting time delay phenomenon, which only occurs in this form during periodic back and forth movements due to the friction hysteresis effect depicted in Fig. 3.5a, the signal of the feedforward part of the control law



(a) Hysteresis loop between the actuator input voltage and the position of the valve spool.



(b) Measured engine valve position using a combined control strategy that neglects the presence of friction hysteresis.

Figure 3.5: Hysteresis effects

can be antedated by a corresponding time delta, which must be known. The time delay phenomenon is clearly visible in Fig. 3.5b, where measurements of the engine valve positions are shown.

3.3 Modelling

It is commonly known that the time delay

$$y(t) = u(t - T) \tag{3.1}$$

can be described by a transcendental transfer function

$$G(s) = \frac{Y(s)}{U(s)} = e^{-Ts} = \frac{1}{e^{Ts}}. \tag{3.2}$$

A rather rough approximation can be obtained by truncating the Taylor polynomial

$$e^x = 1 + x + \frac{1}{2}x^2 + ... + \frac{1}{n!}x^n + \mathcal{O}(x^{n+1}) \tag{3.3}$$

after the first term ($n = 1$):

$$G(s) = e^{-Ts} = \frac{1}{e^{Ts}} \approx \frac{1}{1 + Ts}. \tag{3.4}$$

Of course, the accuracy of this approximation grows with more terms, but so does the order of the resulting transfer function, which constitutes numerical challenges when implementing it in an EKF and it clearly counteracts estimation robustness. In this application, a second-order approximation already caused problems, and the first-order one results to be much more robust.

3.4 EKF using a first order time delay model

Transforming the resulting transfer function back to the time domain yields the differential equation

$$y(t) + T\dot{y}(t) = u(t). \tag{3.5}$$

Please note that, because of legibility reasons, symbols y and u are neither renamed in this section nor in the next, but they are different than in Eq. (3.1) due to the approximation. Following a simple forward Euler discretization given a small enough sampling time t_s,

$$y(k-1) + \frac{T}{t_s}(y(k) - y(k-1)) = u(k-1), \tag{3.6}$$

it is now assumed that T is a state and that it is constant (this strong assumption will be relaxed later, using a non-zero value in the EKF parameter matrix $\mathbf{Q}_1$). The resulting discrete nonlinear system equations

$$y(k) = y(k-1)\left(1 - \frac{t_s}{T(k-1)}\right) + u(k-1)\frac{t_s}{T(k-1)} \tag{3.7}$$

$$T(k) = T(k-1) \tag{3.8}$$

are used for calculating the a-priori-estimate of the states $\hat{\mathbf{z}}_1^-(k) = [y(k)\ T(k)]^T$ in the prediction step, using the a-posteriori estimates of the last iteration $\hat{\mathbf{z}}_1^+(k-1)$ for the $(k-1)$th values. The initial values $\hat{\mathbf{z}}_1^+(0)$ must be defined by the user. The discrete Jacobian is now calculated symbolically as

$$\mathbf{J}_{1,k} = \begin{bmatrix} \frac{T(k-1)-t_s}{T(k-1)} & \frac{t_s(y(k-1)-u(k-1)}{T(k-1)^2} \\ 0 & 1 \end{bmatrix}. \tag{3.9}$$

The a-priori covariance then is

$$\mathbf{P}_{1,k}^- = \mathbf{J}_{1,k}\mathbf{P}_{1,k-1}^+\mathbf{J}_{1,k}^T + \mathbf{Q}_1\ . \tag{3.10}$$

In the first time step, $\mathbf{P}_{1,k-1}^+ = \mathbf{P}_{1,0}^+$ is a given initial value; otherwise, it represents the a-posteriori estimate from the previous time step. The matrix $\mathbf{Q}_1$ must be specified by the user, and it can be considered as a quantification of the model uncertainty represented by corresponding noise processes. The denominator in the following equation is scalar, so the Kalman gain can be expressed in a compact form as

$$\mathbf{k}_{1,k} = \frac{\mathbf{P}_{1,k}^-\mathbf{c}_1}{\mathbf{c}_1^T\mathbf{P}_{1,k}^-\mathbf{c}_1 + r_1}\ . \tag{3.11}$$

The vector $\mathbf{c}_1 = [1\ 0]^T$ specifies which state serves as measurement innovation, in the given case the measured output of the (real) time delay system. Now, in the update step, the a-posteriori estimation of the states and the covariance of the estimation error follow from

$$\hat{\mathbf{z}}_1^+(k) = \hat{\mathbf{z}}_1^-(k) + \mathbf{k}_{1,k}\left(y_{measured} - \mathbf{c}_1^T\hat{\mathbf{z}}_1^-(k)\right), \tag{3.12}$$

$$\mathbf{P}_{1,k}^+ = \left(\mathbf{I}_{2\times 2} - \mathbf{k}_{1,k}\mathbf{c}_1^T\right)\mathbf{P}_{1,k}^-. \tag{3.13}$$

The initial values used in this simple EKF variant are

$$\hat{\mathbf{z}}_1^+(0) = \begin{bmatrix} 0 \\ 0.0005 \end{bmatrix} \text{ and } \mathbf{P}_{1,0}^+ = \begin{bmatrix} 0 & 0 \\ 0 & 100 \end{bmatrix} \tag{3.14}$$

while the parameters were chosen as

$$\mathbf{Q}_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0.0001 \end{bmatrix} \text{ and } r_1 = 1. \tag{3.15}$$

These values were established after many iterations of simulations. The quality of the estimation using these exact parameter values depends in a non-trivial way on the shape and scale of the measured signals. The relatively small value in $\mathbf{Q}_1$ corresponding to the state T, compared to that of the state y, is justified by the assumption that the real time delay to be estimated changes very slowly, if at all,

which is projected in the model, while the relatively large model uncertainty for y is founded in the fact that the value calculated using the approximated model is not to be trusted too much. In $\mathbf{P}_{1,0}^{+}$, however, a large value is used for T since this EKF is not supposed to depend on prior knowledge about the time delay it shall estimate. If such knowledge exists, it can of course be implemented in the initial values, speeding up the convergence of the filter. Unfortunately, for EKFs the convergence in general is not guaranteed at all, and must be investigated on a case-by-case basis, usually by carrying out simulations, as it was done in this contribution. For example, if the initial value for the time delay is chosen as zero or a value smaller than used here, the filter does not converge.

3.5 EKF using fractional order time delay models

To improve the accuracy of the approximation done in Eq. (3.4), a new parameter α is introduced:

$$G_\alpha(s) = e^{-Ts} = \frac{1}{e^{Ts}} \approx \frac{1}{1+Ts^\alpha}, \tag{3.16}$$

or, in a second variant,

$$G_{\alpha,T}(s) = e^{-Ts} = \frac{1}{e^{Ts}} \approx \frac{1}{1+(Ts)^\alpha}. \tag{3.17}$$

Intuitively it is clear that α should be greater than one for the denominator to increase. Also, the Fractional Derivative represented by s^α can be irrational, so a series expansion would have an infinite number of terms, just like the transcendental e^{-Ts}. In order to formulate discrete system equations, the Fractional Derivative definition from [63], Eq. (9a) is used:

$$D^\alpha y(t) \approx \frac{1}{t_s^\alpha} \sum_{j=0}^{m} (-1)^j \binom{\alpha}{j} y(t - jt_s). \tag{3.18}$$

The resulting system for $m = 1$ is

$$y(k) = y(k-1)\left(\alpha(k-1) - \frac{t_s^{\alpha(k-1)}}{T(k-1)}\right) + u(k-1)\frac{t_s^{\alpha(k-1)}}{T(k-1)}, \tag{3.19}$$

$$T(k) = T(k-1), \tag{3.20}$$

$$\alpha(k) = \alpha(k-1) \tag{3.21}$$

or for the second variant

$$y(k) = y(k-1)\left(\alpha(k-1) - \left(\frac{t_s}{T(k-1)}\right)^{\alpha(k-1)}\right) + u(k-1)\left(\frac{t_s}{T(k-1)}\right)^{\alpha(k-1)}, \tag{3.22}$$

$$T(k) = T(k-1), \tag{3.23}$$

$$\alpha(k) = \alpha(k-1). \tag{3.24}$$

For $\alpha = 1$ both variants simplify to the system Eqs. (3.7-3.8). These system equations are used for the prediction step of the "Fractional Extended Kalman Filter". Note that there are, in fact, three approximations used during the derivation of the system equations. The first one is the truncation of the Taylor series for e^{-Ts}, the second one is the approximation of the limit in the Grünwald-Letnikov derivative by a constant sampling time done in [63] and the third one is the small chosen value of m when transforming s^α to the (discrete) time domain. The Jacobian (discrete time arguments are omitted for the sake of brevity) is

$$\mathbf{J}_\alpha = \begin{bmatrix} \alpha - \frac{t_s^\alpha}{T} & (y-u)\frac{t_s^\alpha}{T^2} & y + (u-y)\frac{t_s^\alpha}{T}\log(t_s) \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.25}$$

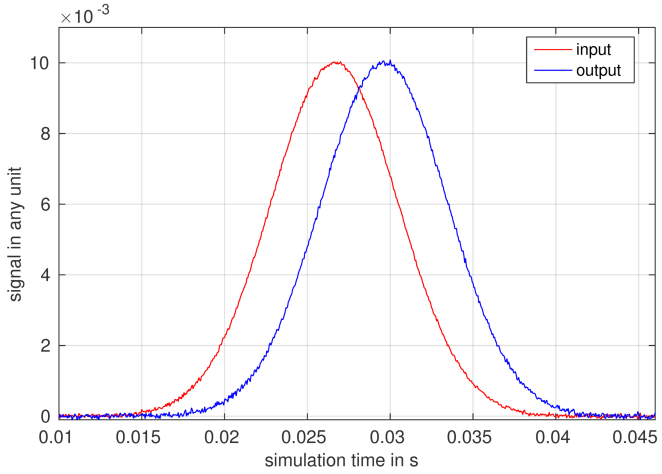


Figure 3.6: Measured noisy input and output, the latter with additional noise.

or, for the second variant,

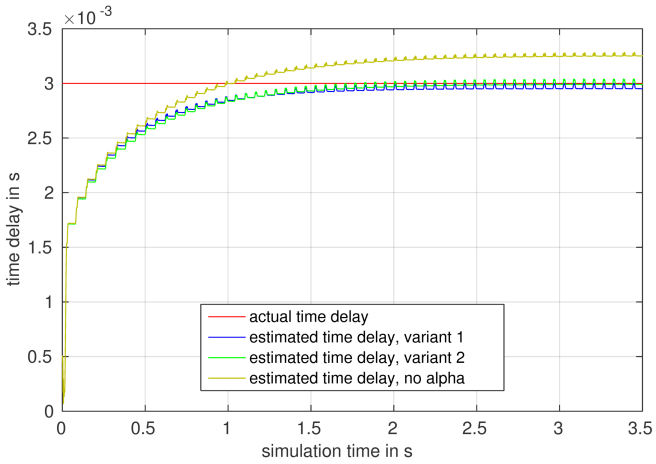
$$\begin{bmatrix} \alpha - \left(\frac{t_s}{T}\right)^{\alpha} & (y-u)\frac{\alpha}{T}\left(\frac{t_s}{T}\right)^{\alpha} & y + (u-y)\left(\frac{t_s}{T}\right)^{\alpha}\log\left(\frac{t_s}{T}\right) \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3.26}$$

The design of the respective EKFs is analogous to the previous section, so the steps are not described further. The only differences (apart from notation) are $c_\alpha = [1\ 0\ 0]^T$ and $\mathbf{I}_{3\times 3}$ in the calculation of the a-posteriori covariance because of the additional state. The parameters, very similar to those in the previous section, are chosen identically for both variants with α:

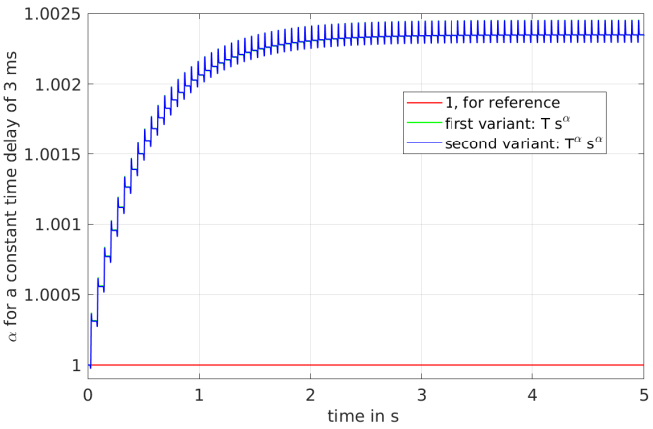
$$\hat{\mathbf{z}}_\alpha^+(0) = \begin{bmatrix} 0 \\ 0.0005 \\ 1 \end{bmatrix},\ \mathbf{P}_{\alpha,0}^+ = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 100 & 0 \\ 0 & 0 & 10 \end{bmatrix}, \mathbf{Q}_\alpha = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.0001 & 0 \\ 0 & 0 & 0.001 \end{bmatrix},\ r_\alpha = 1. \tag{3.27}$$

3.6 Results

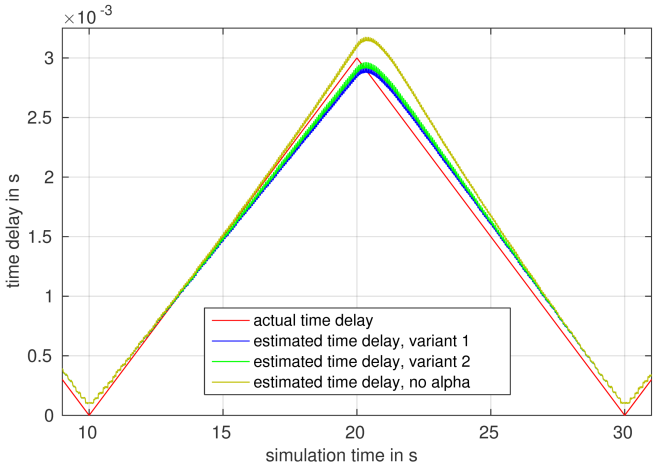
Fig. 3.6 is an example of the input and output signals that were used to validate the proposed approach in simulations. They resemble bell curves, also called Gaussian curves for they have the shape of a Gaussian distribution. This shape proved to be advantageous for combustion engine valve control, which is one possible application. The signal has a periodicity of 60 ms. The shown time delay amounts to 3 ms. In Figs. 3.7a and 3.7b, a step from 0 to 3 ms is applied to the variable time delay function block that is used to implement the time delay in Simulink. The reactions of the EKFs are shown. Note that the variant without the additional parameter α shows a relatively large bias after reaching steady state. Meanwhile, the two variants with α perform well enough for many possible applications. Problematic behaviour can be observed in Fig. 3.7c. There, a triangular signal is used for the variable time delay. The EKFs have no trouble following this slowly changing reference, but they do so with a notable estimation offset. Especially for small time delays, the estimation is quite bad. This is due to the presence of state $T(k-1)$ in the denominator of the system equations from which numerical issues arise. The corresponding values of α can be seen in Fig. 3.7d. This problem is even more visible when a rectangular reference is applied, see Figs. 3.7e and 3.7f.



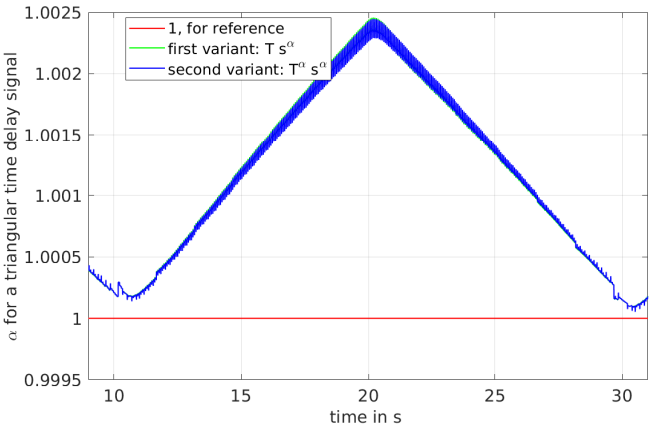
(a) Reaction of the three described EKF variants to a step from 0 to 3 ms.



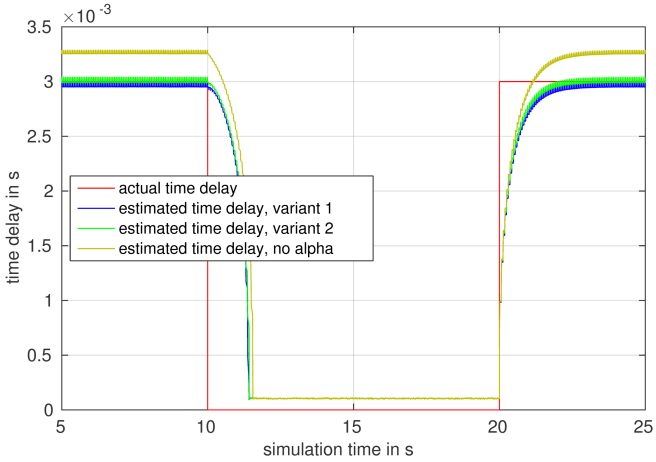
(b) α for constant time delays.



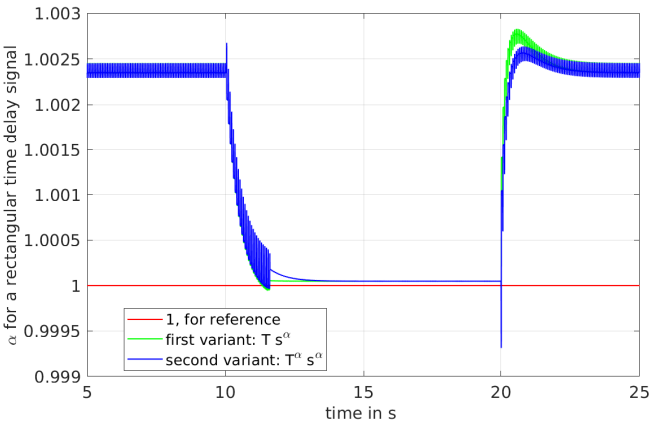
(c) Reaction of the three described EKF variants to triangular time delays.



(d) α for triangular time delays.



(e) Reaction of the three described EKF variants to rectangular time delays.



(f) α for rectangular time delays.

Figure 3.7: Simulation results

4 Tracking control of a piezo-hydraulic actuator using input-output linearization and a cascaded extended Kalman filter structure

Chapter 4 considers the same physical system as in the previous chapter: a piezo-hydraulic multistage actuator for fully variable valve control in internal combustion engines, stemming from a research project by Nils Werner [64]. Unlike the prequel, this chapter deals with the whole control architecture instead of isolated aspects, and in particular with advanced nonlinear control methods instead of PID control, but it does so in simulation studies instead of real test bench experiments. This is a common approach in applied control and requires a validated/verified model of the system, which is present for this actuator. Depictions of the real system and the test bench are given in Fig. 4.1. Note that a newer version of the actuator, which is treated in this chapter, includes a second piezo stack and a second hydraulic displacement amplifier, located on the other side of the double-acting hydraulic cylinder.

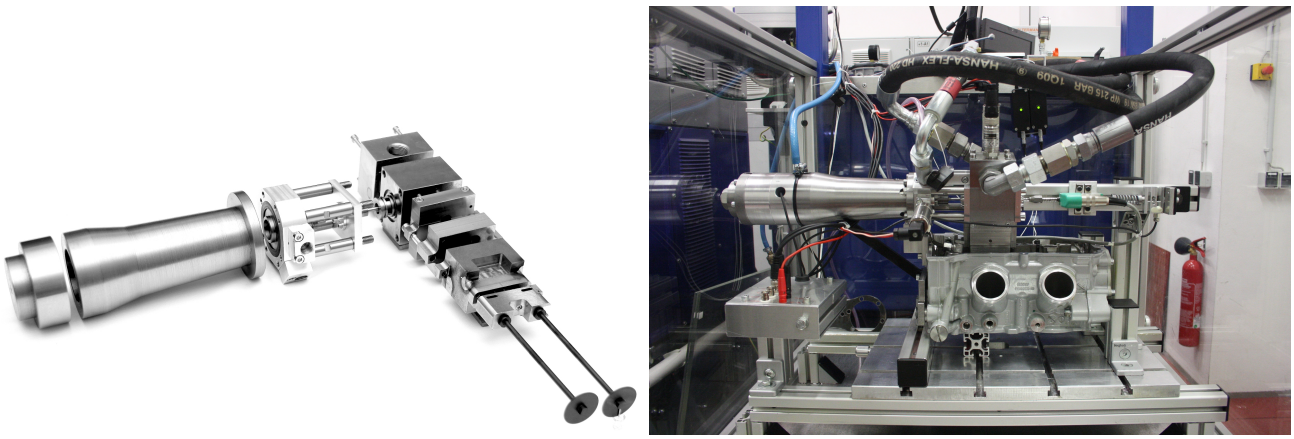


Figure 4.1: Actuator (left, old variant with only one piezo stack) and test bench (right, old variant, with a spring instead of a second piezo stack). Source: Ostfalia University of Applied Sciences, Faculty of Automotive Engineering. N. Werner, G. Bergholz and U. Becker.

Originally, combustion engine valves are opened and closed by the camshaft, with a trajectory profile according to the actual physical shape of the cams, which is immutable during operation (and, thus, cannot be optimized for each set of operating conditions). Due to friction between cams and rocking levers, this traditional approach also suffers from significant thermal friction losses of (at high engine speeds) up to 10% of the engine's nominal mechanical power [64]. Thus, entirely removing the camshaft and related components (and replacing it with fully-variable valve control) achieves a significant reduction of friction losses compared to an engine with camshaft. Using such actuators for each cylinder of the combustion engine, it is possible to optimize and control the valve trajectories in real-time (or even disable multiple cylinders), with large fuel savings potential [64].

Apart from strict soft-landing requirements[1], the intake or exhaust valves of internal combustion engines have to withstand very strong accelerations to open and close within a few milliseconds and, if necessary, to overcome the exhaust gas pressure of up to 100 bar in the cylinder. With an engine speed

[1] A *soft landing* is achieved for closing velocities lower than a certain threshold, often specified between 0.1 and 0.5 $\mathrm{m\,s^{-1}}$, and facilitates an increased expectation of life and better acoustics of the engine valves.

of 6000 RPM, for example, valve accelerations of up to $6\,\mathrm{km\,s^{-2}}$ occur when using the smooth desired valve trajectory proposed in this chapter (and possibly even more, for trajectories with sharp bends or discontinuities in the signal). Note that this is solely due to the prescribed movement and does not even consider exhaust gas forces yet. This application therefore requires actuators with extremely high power densities if the valves are to be controlled individually and fully-variably (which is a prerequisite for more efficient combustion that is optimized to the current operating conditions), while the available installation space is typically limited. For these reasons, double-acting hydraulic cylinders are a sensible choice for moving the engine valves, as they combine high power at high speeds with a low volume (since the large hydraulic power supply can be conveniently placed somewhere else). An actuator is then required in order to move the 4/3-way servo-proportional valve that controls the hydraulic cylinder, typically an electromagnetic one (e.g. voice coil actuators). However, electromagnetic actuators that are both fast and powerful enough would not fit in the available space. Thus, off-the-shelf actuators consisting of stacked piezo crystals were used, which can apply very high forces very quickly, but only over very short distances (max. 100 µm in this case). However, since the servo valve requires a displacement range of up to 10 mm, a hydraulic displacement amplifier was introduced, see Fig. 3.3. In principle, it consists of a small oil chamber with two circular openings for the inlet and outlet pistons. On the input side (towards the piezo stack), the area is large and the piston stroke travel is small, and on the output side (towards the servo valve), the area is small and the piston stroke travel is increased by a corresponding factor. Due to the high pressures inside this oil chamber, expensive sealings are necessary, which, due to static friction effects, cause the hysteresis problem that was already described in chapter 3. (Note that this hysteresis loop is not considered in the model that was used in the simulation studies at the end of this chapter.) Apart from this, the three-stage valve actuator functions satisfactorily at both low and high engine speeds, but the control results to be complex. Thankfully, the piezo actuators come with their own controller box developed by the manufacturer, implementing position control of the output piston (that interfaces the displacement amplifier). Hence, the position can be modelled proportionally to the input voltage.

The oil pressure dynamics in the two chambers of the double-acting hydraulic cylinder (which drives the actual engine valves) can be modelled depending on the positions of the servo valve spool and that of the hydraulic cylinder, using the continuity equation from the field of fluid dynamics (i.e. a volume flow balance), while the dynamics equation of the engine valve itself contains a driving force that is proportional to the difference between these two pressures. To control the position of the servo valve spool, a linearized tenth-order state space model of the hydraulic displacement ratio was derived, which models the elasticity of the hydraulic fluid in the oil chamber as virtual linear springs, whose stiffness is calculated from the elastic modulus of the oil, the piston areas and the oil volume. This linear model was then simplified via automatic model order reduction to a second order system, which can be easily inverted by directly inverting the transfer function (introducing two very fast poles to maintain causality). This way, a feedforward control for the servo valve spool position is obtained, by appropriate specification of the piezo actuators' input voltages. (Feedback effects of the engine valves on the position of the servo valve spool were neglected here). The remaining nonlinear part of the system, which is the pressure dynamics within the hydraulic cylinder, is compensated with the method of feedback linearization, which was mainly coined by the Italian control engineer Alberto Isidori. In this method, states are fed back in such a way that the resulting closed loop (sub)system behaves like an integrator chain – in this case as a single integrator. This enables control of the engine valve position using purely linear control methods. Here, the LQI approach was used, which originates from the field of optimal control and consists of the well-known linear-quadratic regulator (LQR), extended by an integral component for more robust tracking and disturbance rejection. This rather complex combined control strategy, which requires the availability of the entire state vector, is made possible by a novel cascaded EKF variant, in which (in analogy to the control) the nonlinear subsystem is "outsourced" into an additional observer, which forms a loop with the observer for the linear subsystem. In this way, the computational effort can be greatly reduced, which is a necessary condition for real-time implementability. Causality is achieved by adding an artificial delay of one time step inside the loop, which does not harm the quality of the results, as demonstrated by simulations.

4.1 State of the art

Conventional engine valve systems with camshafts offer no variability with respect to the optimal timing for a partial load or full load of the combustion engine. At all speeds of the engine and all load ranges, a constant valve lift curve with a fixed timing is present. Reductions in both the fuel consumption and the exhaust emissions could be achieved by implementing the Miller or Atkinson cycles. This approach, however, would lead to a power reduction as well. Therefore, by charging the filling phase of the cylinder via a compressor or a turbocharger, the power reduction in comparison with conventional internal combustion engines could be compensated. The necessary compromise can be circumvented with a variable valve timing and stroke as proposed here. This can be achieved by replacing the camshaft of the internal combustion engine by actuators for each valve. These are often electromagnetic actuators or combinations with pneumatic or hydraulic servo-systems. As an example for existing systems, the Electro-Hydraulic Valve System (EHVS) by Bosch is depicted in Fig. 4.2. There are four ways by

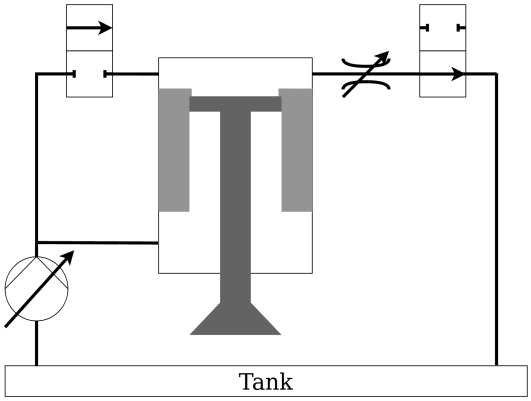


Figure 4.2: Electro-Hydraulic Valve System by Bosch [64].

which the engine valve behaviour can be influenced: Two 2/2-way valves, an adjustable hydraulic pump and a throttle valve. In order to open the engine valve, the right 2/2-way valve is closed and the left one is opened, resulting in a pressure equalization between the upper and lower chamber. However, since the upper active surface of the cylinder piston is larger, the valve slowly starts to open. When the upper holding edges are cleared, the movement is accelerated because the upper surface is slightly enlarged. The valve movement stops shortly after the left 2/2-way valve is closed again, resulting in a force equilibrium holding the engine valve in place. To close the engine valve, the right 2/2-way valve is opened, allowing the fluid to flow through the throttle valve into the tank. The throttle valve must be controlled; it is necessary to decelerate right beforce closing, to avoid a too large contact velocity when attaining its seat (soft landing). The valve lift can be influenced by closing the left 2/2-way valve and, thereby, to initiate the deceleration. The hydraulic pressure supplied by the pump (or rather the pressure difference between pump and tank) determines the acceleration of the engine valve movement. The throttle valve determines the deceleration force and is used to achieve soft landing. This kind of system represents a dedicated actuator for the given application. While control of the engine valve movement seems to be generally intuitive and simple, an exact trajectory tracking is a complicated control problem due to the presence of four inputs and because the 2/2-way valves are on-off valves. In this chapter, a novel kind of system is presented that consists of two antagonistic piezo-electric actuators with hydraulic displacement amplifiers, driving a 4/3-way servo (proportional) valve that is in turn driving one or more double-acting hydraulic cylinders. This architecture allows to combine the advantages of piezo actuators (accuracy and speed) with those of hydraulic systems (high power and compact dimensions of the hydraulic cylinders, being the actual actuators).

4.1.1 Control strategies for hydraulic systems

In recent control applications for hydraulic servo-systems, the basic concepts of nonlinear identification and nonlinear control have already been successfully used. Studies such as [65–68] consider innovative control strategies for similar hydraulic systems. In particular in [65], a decentralised control approach is proposed using Lyapunov techniques and a Sliding Mode control design for the positioning of two coupled cylinders. Moreover, the control structure is extended with a sliding mode disturbance observer. In [66], a cascaded backstepping control is designed, where fast inner control loops determine the difference pressure in each hydraulic cylinder, while the position as well as the generated force are controlled in the outer loop. Here, the design of the outer control loop is based on adaptive backstepping. In [67], a cascaded flatness-based control is developed for the MIMO system. Model parameter uncertainties are estimated by a reduced-order disturbance observer and used for a compensation. Nonlinear techniques such as feedback linearisation, Sliding Mode control and EKF for the estimation of immeasurable state variables become important for feedback control, see [69] for a mechanical system and also [70, 71] for a hydraulic one. In particular, the paper [71] deals with the same system model as the contribution at hand, however, only with a quasi-static description of the pressure dynamics. The positioning of the spool valve is achieved by feedforward control, i.e., a flat inversion of the quasi-static pressure dynamics with desired values. The engine valve position is accurately controlled by Sliding Mode techniques, where a switching control part counteracts model uncertainty. However, the states that are used for the feedback control are assumed to be measurable. In contrast, this work envisages an input-output linearization by an inversion of the pressure dynamics and a linear-quadratic proportional-integral tracking control of the engine valve. Here, a Cascaded Extended Kalman Filter provides the necessary estimates for the state variables, instead of relying on measurements like in [71]. The integral part takes into account model uncertainty. Standard control approaches like PID are very well known because of their high practical relevance. Recent applications, however – in particular those in sophisticated mechatronic systems, where hydraulics is combined with unconventional drives like piezo-elements – call for new approaches. This is especially true considering the technical challenges posed by this specific application. To repeat the initially given example, at 6000 RPM, high accelerations of up to $2700\,\mathrm{m\,s^{-2}}$ are required for lifting the valve, and up to $6000\,\mathrm{m\,s^{-2}}$ at the peak of the valve trajectory to brake and accelerate in the opposite direction [64]. In case of the exhaust valve, even more force is necessary to overcome exhaust gas pressures of up to 100 bar in the cylinder. Furthermore, soft landing is required to reduce material wearout and noise. These requirements justify abandoning the classical control structures like PID, favored by practitioners for their simplicity and intuitive tuneability, and employing more complex algorithms.

4.1.2 Sensorless control

Sensorless control focusses on the estimation of state variables to avoid measurements or to reduce the number of sensors as much as possible. In the given case, the engine valve velocity and the hydraulic pressures in both chambers of the cylinder have to be estimated. Obtaining measurements of these variables tends to be very difficult or too expensive, especially regarding the pressure in the hydraulic chambers. Therefore, an observation strategy is advantageous. In general, local observer design for nonlinear control systems represents one of the key problems both in theory and in applications. In the recent literature, several different sensorless control structures are common. In [72], a Sliding Mode observer is develloped. For non-Gaussian noise, an online state estimation scheme is presented in [73]. In the context of Kalman filters, a large number of papers have been published recently, e.g. the EKF in [74]. In particular, Kalman filter structures such as in [75–78] are often applied in sensorless drive control. Kalman filters guarantee very high performance and optimal estimation in the presence of noisy measurements and model uncertainty if certain conditions are met. In particular, in [79] and [80] innovative Kalman filter structures are employed for the control of electromagnetic actuators. Another very important issue in many applications is the reduction of the computation load. In the literature, there are several papers that contribute to a reduction of the complexity in terms of the numerical

effort required by the proposed observers. A large effort was devoted to reduced-order observers in [81]. Moreover, efficient implementations of EKFs in a cascade structure are presented in [82], [83], and [74].

4.1.3 Contributions and structure of this chapter

The piezo-hydraulic actuator is controlled by an input-output linearization of the nonlinear hydraulic system part, a feedforward compensation of the piezo-mechanical system, and an LQI control of the engine valve position. Thereby, robust trajectory tracking of the engine valve position is achieved. The LQI control involves a linear state feedback controller enhanced by a proportional-integral feedback of the tracking error w.r.t. the engine valve position. As already pointed out, the estimation of the valve velocity and the pressures in both hydraulic chambers is of fundamental importance because access by measurements is very difficult or expensive. Therefore, one of the main contributions of this work is a cascaded EKF. This estimator is the basis of the implementation of a sensorless control of the piezo-hydraulic actuator to be used in a camless engine. The avoidance of sensors is always an advantage, not only from an economical point of view but also in terms of reducing faults and guaranteeing a long life time. To conclude, the main contributions of this work are:

- to propose an innovative actuator for an accurate motion control of engine valves that consists of two antagonistic piezo-elements, a mechanical and a hydraulic part.
- to propose a control structure that involve three main parts: an input-output linearization using appropriate feedback to compensate the nonlinearities in the hydraulic system part, a feedforward action to compensate the system dynamics of the piezo-mechanical part, and an LQI control that renders the trajectory tracking robust against disturbances, parameter uncertainty and unmodeled dynamics.
- to achieve a sensorless motion control engine valve that only requires the measurements of the valve spool position and the position of the engine valve itself. Furthermore, the efficient implementation of the proposed estimator structure needs only a limited number of calculations.
- The numerical complexity of the proposed cascaded EKF structure is compared with a standard EKF, developed for the same application.

This chapter is structured in the following way: Section 4.2, starting with the generation of the desired trajectories, is devoted to the description of the actuator and its mathematical modelling. In Section 4.3, a feedforward control design for the piezo-mechanical subsystem is described together with a input-output linearisation of the hydraulic part. An LQI control of the engine valve position contributes to a robustification of the trajectory tracking. In Section 4.4, the design of the cascaded EKF structure is presented. Moreover, a comparison between the proposed cascaded EKF structure and an alternative standard EKF in terms of the calculation load is provided. Simulation results are presented and discussed in Section 4.5, which show the high accuracy that is achievable with the proposed overall control structure. The chapter closes with conclusions in Section 4.6.

4.2 System modelling and trajectory generation

As basis for the model-based control design, a mathematical model of the physical system is to be developed. Furthermore, in the envisaged application to camless combustion engines, a suitable trajectory generation is important. The overall model consists of two subsystems that are assumed to be decoupled:

- Two antagonistic piezo actuators combined with hydraulic displacement amplifiers acting on the valve spool
  - input signals: input voltage of the corresponding piezo actuator

  - output signal: position of the valve spool

- Servo valve and hydraulic cylinder with the engine valve
  - input signal: position of the valve spool
  - output signal: position of the engine valve

Small retroactive effects of the second subsystem on the first one are neglected in the control-oriented model to simplify the subsequent development of control strategies.

4.2.1 Generation of desired trajectories

The desired engine valve trajectory is chosen as a Gaussian

$$y(t) = He^{-\left(\frac{mt+a+c}{b}\right)^2} , \tag{4.1}$$

which belongs to the class of C^∞ functions. The expression mt denotes a periodic ramp function covering the range – as typical for four-stroke engines – from 0° (crank angle) to 720°. Thus, the slope factor becomes $m = \frac{360°}{60\,\mathrm{s}} n_e$, with the engine speed n_e in RPM (explaining the denominator), whereas the constant a has a value of $a = -360°$. The parameter c can be used online to influence the timing of the valve opening. Thus, the crank angle function $mt + a + c$ should be implemented as a resettable integrator to ensure it maintains in $[-360°, 360°]$. The parameter $b = 65°$ controls the aperture of the Gaussian, and thus the opening duration in dependence of the engine speed. It is proportional to the full time width of the valve curve at a height of $\frac{H}{2}$ with a factor of $\frac{m}{2\sqrt{\ln 2}}$. Here, the parameter $H = 10\,\mathrm{mm}$ denotes the maximum valve lift. Fig. 4.3 shows the shape of the generated trajectory in the worst case of $n_e = 8000\,\mathrm{RPM}$ as well as the relevant region of its frequency spectrum. Here, frequency components larger than 500 Hz can be neglected in good accuracy. This approach is a very simple and convenient way to generate mathematically well-defined intake/exhaust valve trajectories because unlike e.g. polynomials of similar shape, it has only 3 parameters (amplitude H, width b and timing c). These can be expressed in analogy to traditional combustion engines. The generated continuous signal is very *smooth* in the sense that its spectrum declines rapidly towards high frequencies compared to discontinuous signals like triangular ramp-like signals. This allows for more aggressive model order reductions (or similar simplifications of linear models) in order to design feedforward actions. Furthermore, when designing flatness-based feedforward controls (which was not done in this contribution), derivatives of the desired output are required, sometimes even very high-order derivatives. Thus, the main advantage of this symbolic ansatz function is its infinite differentiability. For example, the first time derivative is

$$\dot{y}(t) = -2m\frac{mt+a+c}{b^2}\underbrace{He^{-\left(\frac{mt+a+c}{b}\right)^2}}_{y(t)} . \tag{4.2}$$

4.2.2 Nonlinear modelling of the hydraulic cylinder and the engine valve

The engine valve is actuated by a double-acting hydraulic cylinder as depicted in Fig. 4.4, cf. [70, 71]. The equation of motion of the engine valve is described by the linear second-order ordinary differential equation (ODE)

$$M_v\ddot{y}(t) + b_v\dot{y}(t) = \Delta p(t)S - d(t) . \tag{4.3}$$

Here, $\dot{y}(t)$ represents the engine valve velocity, $\ddot{y}(t)$ is the engine valve acceleration, $M_v = 62.5\,\mathrm{g}$ is the reduced mass of the hydraulic piston and the engine valve and $b_v = 0.1\,\mathrm{kg\,s^{-1}}$ a damping coefficient. The area $S = 0.754\,\mathrm{cm^2}$ is the active surface of the cylinder, $\Delta p(t)$ the pressure difference $p_A(t) - p_B(t)$ and $d(t)$ a disturbance force caused by exhaust gases, which affects the engine valve motion. Please note that $d(t)$ may also take into account any model uncertainties and other unmodelled effects. Since the hydraulic fluid is assumed to be compressible, additional differential equations for the pressure

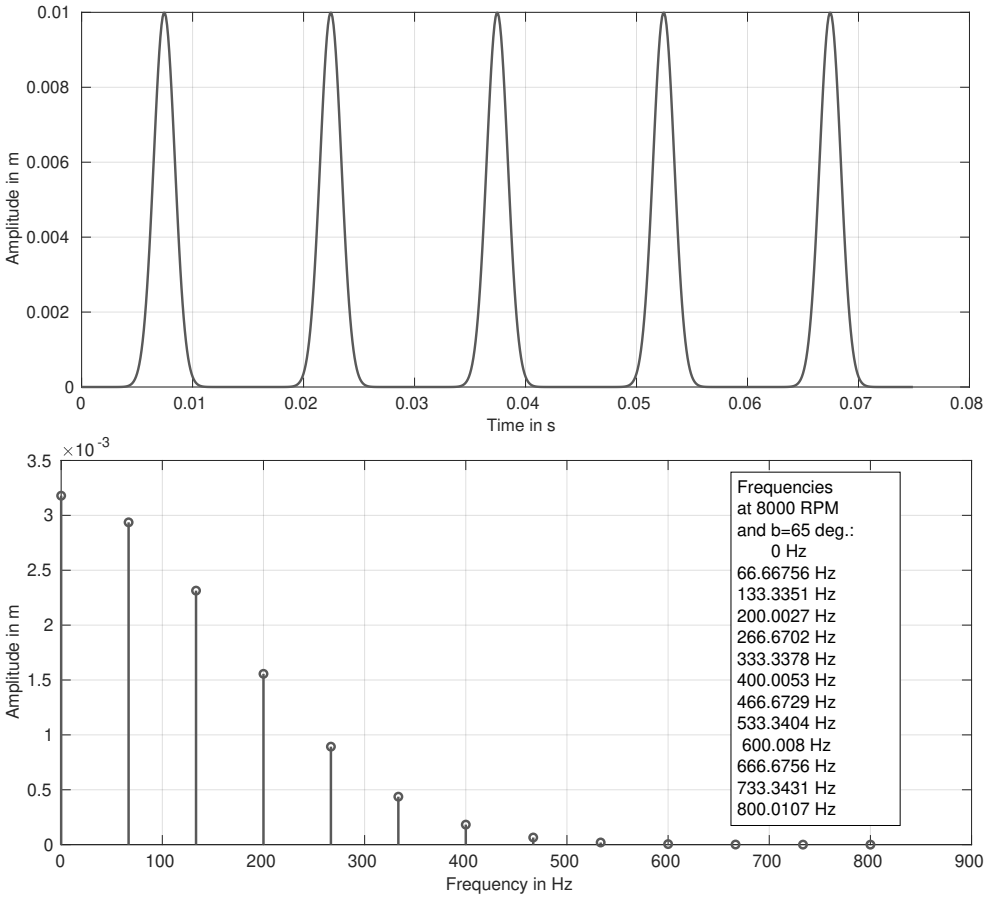


Figure 4.3: Desired valve trajectory and its frequency content (FFT) at $n_e = 8000\,\mathrm{RPM}$.

dynamics in the cylinder chambers need to be derived. They arise from the continuity equation (volume flow balance), see [70], as well as an oil model

$$\dot{p}_A(t) = \frac{E_{Oil}}{V_{A,0} + Sy(t)}(Q_A(t) - S\dot{y}(t) - C_{Li}(p_A(t) - p_B(t)) \ , \tag{4.4}$$

$$\dot{p}_B(t) = \frac{E_{Oil}}{V_{B,0} - Sy(t)}(Q_B(t) + S\dot{y}(t) + C_{Li}(p_A(t) - p_B(t)) \ . \tag{4.5}$$

Here, $V_{A,0} = 0.113\,\mathrm{cm}^3$ and $V_{B,0} = 1.018\,\mathrm{cm}^3$ are the initial volumes of the cylinder chambers that correspond to a completely closed engine valve at the position $y(t) = 0$. The bulk modulus of the hydraulic fluid is given by a constant value of $E_{Oil} = 1.883\,\mathrm{GPa}$, where a dependence on the pressure and on the temperature is neglected. The leakage coefficient C_{Li} characterizes internal leakage, which is directed from chamber A to chamber B. The volume flows $Q_A(t)$ and $Q_B(t)$ into and out of the hydraulic chambers can be adjusted by the valve spool position. Here, a negative sign corresponds to a flow out of the chambers. The pipeline dynamics are not modelled explicitly because their impact is negligible. The valve shown in Fig. 4.5 is a 4/3-way servo valve, which allows for a vanishing flow rate in the case of $x_2(t) = 0$, i.e., the middle position of the valve spool. For a concise symbolic description of the volume flow rates, the function $\mathrm{sg}(x)$ is introduced as follows

$$\mathrm{sg}(x) = \begin{cases} x & for \ \ x > 0 \\ 0 & for \ \ x \leq 0 \end{cases} \ . \tag{4.6}$$

Then, the flow rates result in

$$Q_A(t) = c\ \mathrm{sg}(x_2(t))\sqrt{p_0 - p_A(t)} - c\ \mathrm{sg}(-x_2(t))\sqrt{p_A(t) - p_T} \ , \tag{4.7}$$

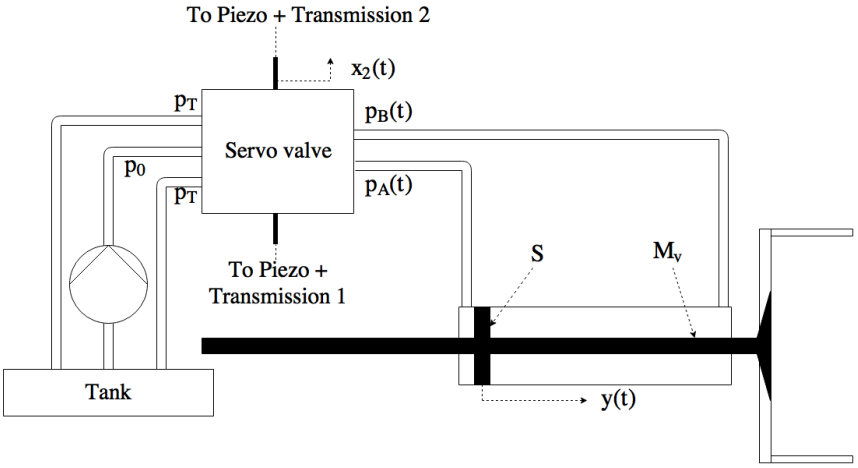


Figure 4.4: Hydraulic system part driving the engine valve.

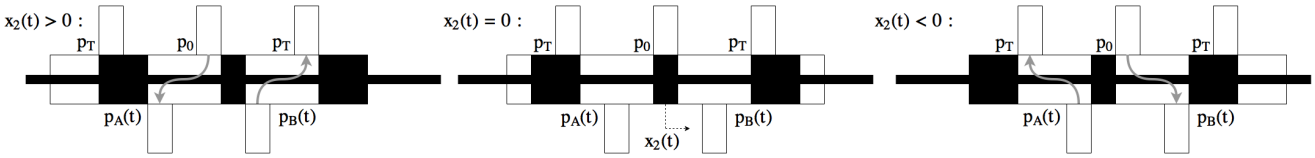


Figure 4.5: Valve spool of the servo valve with indicated flow directions.

$$Q_B(t) = c\ \mathrm{sg}(-x_2(t))\sqrt{p_0 - p_B(t)} - c\ \mathrm{sg}(x_2(t))\sqrt{p_B(t) - p_T}\ , \tag{4.8}$$

where $p_0 = 100\,\mathrm{bar}$ represent the absolute pressure provided by the hydraulic power supply, whereas $p_T = 1\,\mathrm{bar}$ is the pressure of the hydraulic fluid tank. The position of the valve spool is given by $x_2(t)$ and represents the control input for the nonlinear system part. The hydraulic coefficient

$$c = \alpha_D n b \sqrt{\frac{2}{\rho_{oil}}} \tag{4.9}$$

depends on the hydraulic fluid density $\rho_{oil} = 845.6\,\mathrm{kg\,m^{-3}}$, the number n of orifices in the servo valve (2), the width $b = 5\,\mathrm{mm}$ as well the coefficient $\alpha_D = 1$. For further details, see [64]. These five equations lead directly to a nonlinear state-space representation for the hydraulic system part:

$$\dot{\mathbf{z}} = \mathbf{f}(\mathbf{z}, \mathbf{u}, \mathbf{d})\ , \tag{4.10}$$

with the state vector

$$\mathbf{z} = \begin{bmatrix} z_1 & z_2 & z_3 & z_4 \end{bmatrix}^{\mathbf{T}}\ . \tag{4.11}$$

The states are chosen as $z_1 = y(t),\ z_2 = \dot{y}(t),\ z_3 = p_A(t),\ z_4 = p_B(t)$, whereas the input is given by the valve spool position $u = x_2(t)$. The disturbance force is $d(t)$, and the output corresponds to the engine valve position $z_1 = y(t)$. Then, the right-hand side can be stated as

$$\mathbf{f}(\mathbf{z}, \mathbf{u}, \mathbf{d}) = \begin{bmatrix} z_2 \\ \frac{1}{M_v}((z_3 - z_4)S - z_2 b_v - d) \\ \frac{E_{Oil}}{V_{A,0}+z_1 S}(c\ \mathrm{sg}(u)\sqrt{p_0 - z_3} - c\ \mathrm{sg}(-u)\sqrt{z_3 - p_T} - z_2 S - C_{Li}(z_3 - z_4)) \\ \frac{E_{Oil}}{V_{B,0}-z_1 S}(c\ \mathrm{sg}(-u)\sqrt{p_0 - z_4} - c\ \mathrm{sg}(u)\sqrt{z_4 - p_T} + z_2 S + C_{Li}(z_3 - z_4)) \end{bmatrix}. \tag{4.12}$$

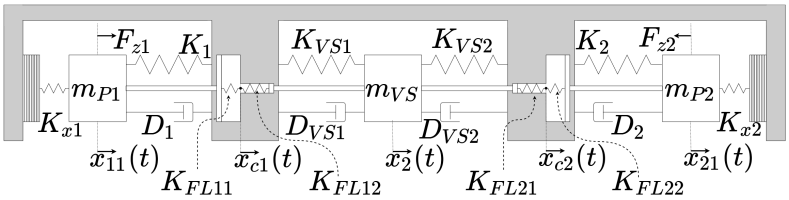


Figure 4.6: Antagonistic structure with two piezos and hydraulic amplifiers, see [84].

4.2.3 Modelling of the piezo-electrically driven valve spool with hydraulic displacement amplifiers

As indicated in Fig. 4.6, the hydraulic fluid is assumed to be compressible. In analogy to the nonlinear part, the bulk modulus is assumed to be constant in this subsystem as well. As proposed in [64], it is possible to calculate a resulting spring constant from the bulk modulus that approximately represents the resulting dynamical behaviour by the following linear relation

$$K_{FL} = S^2 \frac{E_{Oil}}{V_0} \ . \tag{4.13}$$

Here, K_{FL} denotes the spring constant and V_0 the considered fluid volume. This representation takes the two different piston surfaces on both sides – which have a ratio of 100 – into account, and thus two conjoined chambers with identical oil volumes. On the piezo side, the stiffness is given by $K_{FL1} = 4.88\,\mathrm{GN\,m^{-1}}$, whereas the stiffness on the valve side is $K_{FL2} = 488\,\mathrm{kN\,m^{-1}}$. The actual force transmission ratio achieved by this design, cf. [71], depends on the quotients

$$A_{F1} = \frac{A_1}{A_1 + A_2}, \quad A_{F2} = \frac{A_2}{A_1 + A_2} \ . \tag{4.14}$$

Here, A_1 denotes the surface area at the piezo-side, and A_2 surface area at the valve-side. Given the piezo parameters $D_x = 180\,\mathrm{nm\,V^{-1}}$, characterizing the displacement per applied voltage, and the stiffness $K_{x1,2} = 166.67\,\mathrm{MN\,m^{-1}}$, the linear ODEs for the mechanical system with five point masses depicted in Fig. 4.6 become

$$\begin{aligned}
m_{P1}\ddot{x}_{11}(t) &= V_{z1}D_xK_{x1} - x_{11}(t)(K_{x1} + K_1 + A_{F1}K_{FL11}) - \dot{x}_{11}(t)D_1 + x_{c1}(t)K_{FL11} \ , && (4.15)\\
m_{Oil}\ddot{x}_{c1}(t) &= x_{11}(t)A_{F1}K_{FL11} + x_2(t)A_{F2}K_{FL12} - x_{c1}(t)(K_{FL11} + K_{FL12}) \ , && (4.16)\\
m_{VS}\ddot{x}_2(t) &= x_{c1}(t)K_{FL12} + x_{c2}(t)K_{FL21} - \dot{x}_2(t)(D_{VS1} + D_{VS2}) && (4.17)\\
&\quad - x_2(t)(A_{F2}K_{FL12} + K_{VS1} + K_{VS2} + A_{F2}K_{FL21}) \ ,\\
m_{Oil}\ddot{x}_{c2}(t) &= x_2(t)A_{F2}K_{FL21} + x_{21}(t)A_{F1}K_{FL22} - x_{c2}(t)(K_{FL21} + K_{FL22}) \ , && (4.18)\\
m_{P2}\ddot{x}_{21}(t) &= -V_{z2}D_xK_{x2} + x_{c2}(t)K_{FL22} - x_{21}(t)(A_{F1}K_{FL22} + K_2 + K_{x2}) - \dot{x}_{21}(t)D_2 \ . && (4.19)
\end{aligned}$$

Here, the inputs of the piezo elements are the voltages $V_{z1}(t)$ and $V_{z2}(t)$. The other parameters are given by $A_{F1} = 0.9901$, $A_{F2} = 0.0099$, $m_{VS} = 28.1\,\mathrm{g}$, $m_{P1,2} = 70.8\,\mathrm{g}$, $K_{1,2} = 140\,\mathrm{kN\,m^{-1}}$, $K_{FL11,22} = K_{FL1}$, $K_{FL21,12} = K_{FL2}$, $D_{1,2} = 1000\,\mathrm{kg\,s^{-1}}$, $K_{VS1,2} = 3.1\,\mathrm{kN\,m^{-1}}$ and $D_{VS1,2} = 16.8\,\mathrm{kg\,s^{-1}}$. For the subsequent control design, a linear state-space representation can be derived as follows

$$\mathbf{x} = \left[x_{11}, \dot{x}_{11}, x_{c1}, \dot{x}_{c1}, x_2, \dot{x}_2, x_{c2}, \dot{x}_{c2}, x_{21}, \dot{x}_{21},\right]^T , \tag{4.20}$$

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\begin{bmatrix} V_{z1}(t) \\ V_{z2}(t) \end{bmatrix}, \ y(t) = \mathbf{c}^T\mathbf{x}(\mathbf{t}), \ \mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix}. \tag{4.21}$$

Submatrix $\mathbf{A}_{11}$ is given by

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -\frac{K_{x1}+K_1+A_{F1}K_{FL11}}{m_{P1}} & -\frac{D_1}{m_{P1}} & \frac{K_{FL11}}{m_{P1}} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ \frac{A_{F1}K_{FL22}}{m_{Oil}} & 0 & -\frac{K_{FL21}+K_{FL22}}{m_{Oil}} & 0 & \frac{A_{F2}K_{FL12}}{m_{Oil}} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{K_{FL12}}{m_{VS}} & 0 & -\frac{A_{F2}(K_{FL12}+K_{FL21})+K_{VS1}+K_{VS2}}{m_{VS}} \end{bmatrix}, \tag{4.22}$$

whereas $\mathbf{A}_{12} = \mathbf{0}_{4\times5}$ is a zero matrix. The lower block matrices result in

$$\mathbf{A}_{21} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{A_{F2}K_{FL21}}{m_{Oil}} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{4.23}$$

$$\mathbf{A}_{22} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -\frac{D_{VS1}+D_{VS2}}{m_{VS}} & \frac{K_{FL21}}{m_{VS}} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & -\frac{K_{FL21}+K_{FL22}}{m_{Oil}} & 0 & \frac{A_{F1}K_{FL22}}{m_{Oil}} & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & \frac{K_{FL22}}{m_{P2}} & 0 & -\frac{K_{x2}+K_2+A_{F1}K_{FL22}}{m_{P2}} & -\frac{D_2}{m_{P2}} \end{bmatrix}. \tag{4.24}$$

The input matrix can be stated as

$$\mathbf{B} = \begin{bmatrix} 0 & \frac{D_x K_{x1}}{m_{P1}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{D_x K_{x2}}{m_{P2}} \end{bmatrix}^T, \tag{4.25}$$

and the output vector results in

$$\mathbf{c}^T = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{4.26}$$

4.3 Model-based trajectory tracking

4.3.1 Feedforward control design for the linear subsystem

In the sequel, the feedforward control design for the piezo-electrically driven valve spool is considered. Given a desired trajectory for the valve spool position, the necessary input signals for the antagonistic piezo elements are determined. To reduce the implementation effort, a model-order reduction is reasonable. As a consequence, the order of the linear subsystem according to Subsect. 4.2.3 is reduced using Matlab's **modred** function. Hankel singular values shown in Fig. 4.7 indicate that the following second-order system is suitable for the feedforward control design

$$\dot{\mathbf{x}}_{D2}(t) = \mathbf{A}_{D2}\mathbf{x}_{D2}(t) + \mathbf{B}_{D2}\begin{bmatrix} V_{z1}(t) \\ V_{z2}(t) \end{bmatrix},\ y_{D2}(t) = \mathbf{c}_{D2}^T\mathbf{x}_{D2}(t),\ \mathbf{x}_{D2}(t) = \begin{bmatrix} x_2(t) \\ \dot{x}_2(t) \end{bmatrix}. \tag{4.27}$$

Fig. 4.8 demonstrates that this order reduction is justified as long as input signals $V_{z1}(t)$ – apart from inevitable noise – do not contain frequency components larger than 1 kHz. The feedforward control strategy is based on the use of symmetric input voltages, which follows intuitively when looking at the antagonistic structure in Fig. 4.6

$$\overline{V_z} = \frac{1}{2}(V_{z1}(t) + V_{z2}(t)) = const. \tag{4.28}$$

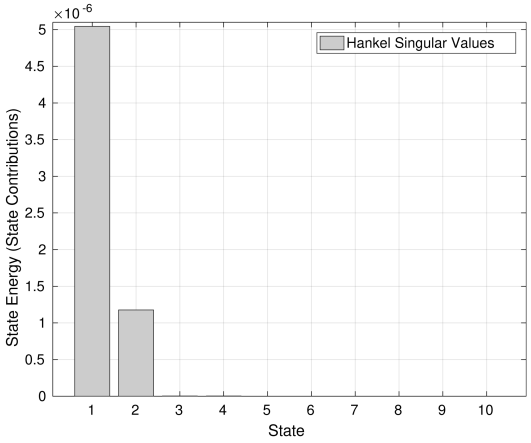


Figure 4.7: Hankel singular values.

Here, the constant mean value $\overline{V_z}$ can be specified by the user. The authors chose a mean value of $\overline{V_z} = 400\,\mathrm{V}$ because the piezo actuators have an input voltage range from $0\,\mathrm{V}$ to $1000\,\mathrm{V}$. The signals $V_{z1}(t)$ and $V_{z2}(t)$ are symmetric around this mean value because $V_{z2}(t)$ is calculated according to $2\overline{V_z} - V_{z1}(t)$. In order to obtain an invertible SISO system model, $\overline{V_z}$ is introduced as an additional state. Furthermore, the 2×2 matrix $\mathbf{B}_{D2}$ is partitioned into two columns

$$\mathbf{B}_{D2} = \begin{bmatrix} \mathbf{B}_{D2,1} & \mathbf{B}_{D2,2} \end{bmatrix} . \tag{4.29}$$

The extended system model can be stated as

$$\dot{\mathbf{x}}_{D3}(t) = \mathbf{A}_{D3}\mathbf{x}_{D3}(t) + \mathbf{B}_{D3}V_{z1}(t),\ y_{D3}(t) = \mathbf{c}_{D3}^T\mathbf{x}_{D3}(t) , \tag{4.30}$$

where the matrices and vectors are given by

$$\mathbf{x}_{D3}(t) = \begin{bmatrix} \mathbf{x}_{D2}(t) \\ \overline{V_z} \end{bmatrix}, \mathbf{A}_{D3} = \begin{bmatrix} \mathbf{A}_{D2} & 2\mathbf{B}_{D2,2} \\ \mathbf{0}_{1\times 3} \end{bmatrix}, \mathbf{B}_{D3} = \begin{bmatrix} 2\mathbf{B}_{D2,1} \\ 0 \end{bmatrix}, \mathbf{c}_{D3} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^T . \tag{4.31}$$

The corresponding transfer function can be determined using Matlab's **ss2tf** routine, which yields

$$G_{D3}(s) = \frac{b_1 s}{a_3 s^3 + a_2 s^2 + a_1 s} . \tag{4.32}$$

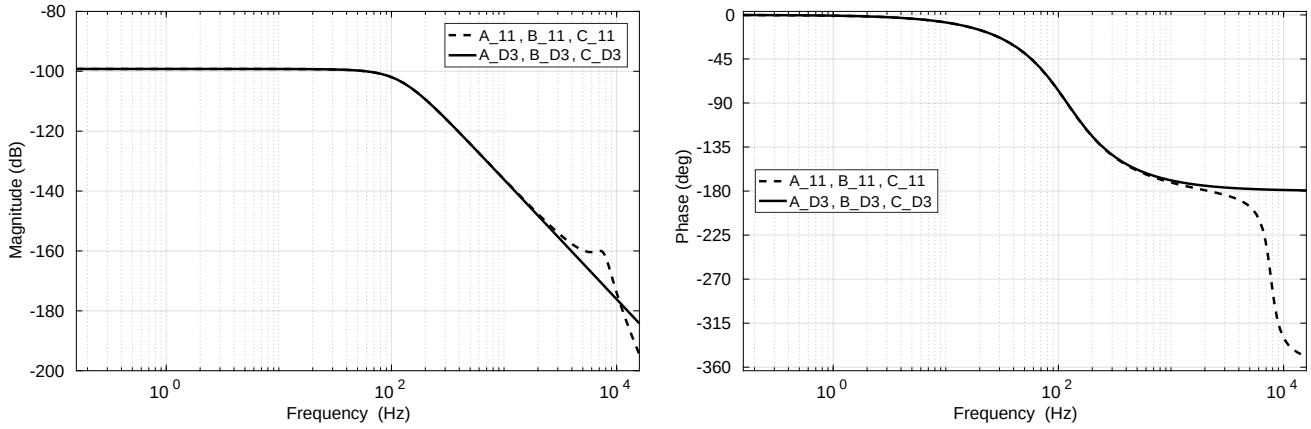


Figure 4.8: Bode diagram of the original model (4.21) and the reduced-order model (4.27),(4.30).

The feedforward control involves an inversion of this transfer function

$$G_{D3}^{-1}(s) = \frac{a_3 s^3 + a_2 s^2 + a_1 s}{b_1 s} \ . \tag{4.33}$$

Since this is not a proper system, two relatively fast poles are added to obtain an implementable inverse

$$G_{D3-Ffwd}(s) = \frac{a_3 s^2 + a_2 s + a_1}{b_1 (10^{-9} s + 1)^2} \ . \tag{4.34}$$

4.3.2 Input-output linearization

A widely used approach to control nonlinear systems is input-output linearization, see in particular the pioneer work of [85]. Moreover, application-oriented presentations can bei found in [70, 86]. This allows to cancel nonlinearities in the input-output behaviour, resulting in an integrator chain [85]. Afterwards, this linear system has to be stabilized in a final design step using a suitable linear control strategy. If the relative degree, which characterizes the order of the input-output dynamics, is smaller than the system order, an internal dynamics exist. A condition for the implementability of the input-output linearization is the stability of the internal dynamics, an unobservable part of the overall system dynamics. Typically, the stability analysis is simplified by considering only the zero dynamics. It represents the linearized internal dynamics with a vanishing output as well as vanishing time-derivatives of the output and the corresponding input signal. A further condition is the non-existence of any singularities in the input signal.

The overall control strategy is a cascaded one, which is outlined in Fig. 4.9. Given the chosen output $\Delta F = (p_A - p_B)S$, only the nonlinear part of the system (4.12) is considered in the sequel. Performing a time-derivation of the controlled ouput

$$\Delta \dot{F}(t) = (\dot{p}_A(t) - \dot{p}_B(t))S \tag{4.35}$$

and inserting the differential equations for $\dot{p}_A$ and $\dot{p}_B$ leads to the occurrence of the control input x_2. Thus, the relative degree of the nonlinear part of the second subsystem is one, and the internal dynamics is of order one. A suitable state variable for the internal dynamics is either p_A or p_B. Eq. (4.12) shows that the zero dynamics is asymptotically stable in the case of internal leackage. In the next design step, the highest derivative of the controlled output $\Delta \dot{F}(t)$ is chosen as stabilizing control input $v(t)$. Solving for the desired input variable, $x_2(t)$ is renamed as $x_{2d}(t)$, and the inverse dynamics can be stated as

$$x_{2d} = \frac{v(t) + \left(\frac{E_{Oil}}{V_{A,0}+y(t)S} + \frac{E_{Oil}}{V_{B,0}-y(t)S}\right)\left(\dot{y}(t)S^2 - SC_{Li}(p_A(t) - p_B(t))\right)}{E_{Oil}Sc\left[\left(\frac{\sqrt{p_0 - p_A(t)}}{V_{A,0}+y(t)S} + \frac{\sqrt{p_B(t)-p_T}}{V_{B,0}-y(t)S}\right)H(x_{2d}(t-T_s)) + \left(\frac{\sqrt{p_A(t)-p_T}}{V_{A,0}+y(t)S} + \frac{\sqrt{p_0 - p_B(t)}}{V_{B,0}-y(t)S}\right)H(-x_{2d}(t-T_s))\right]} \tag{4.36}$$

where $x_{2d}(t)$ is delayed by one sampling time step on the right-hand side according to $sg(x_{2d}(t)) = x_{2d}(t)H(x_{2d}(t - T_s))$ with the Heaviside function H, to avoid any algebraic loops. All the other signals – with the exception of the stabilizing input $v(t)$ – are available from measurements and/or estimations and employed in the inverse dynamics. Using estimates instead of noisy measurements often yields better results because noise properties can be exploited adequately, e.g., by an (extended) Kalman filter approach. Employing other filtering techniques like model-free low-pass filtering would provide smooth signals as well, however, only with a certain time lag. Also, the Kalman filter is useful as an estimator of unmeasurable states by exploiting a-priori knowledge of the system in the form of the system model equations. In the given case, the engine valve position is the only measurable signal used for feedback, whereas the remaining ones are estimated by a Cascaded Extended Kalman Filter (CEKF) structure that will be described in one of the next chapters. For these two reasons, estimates are used for all the signals.

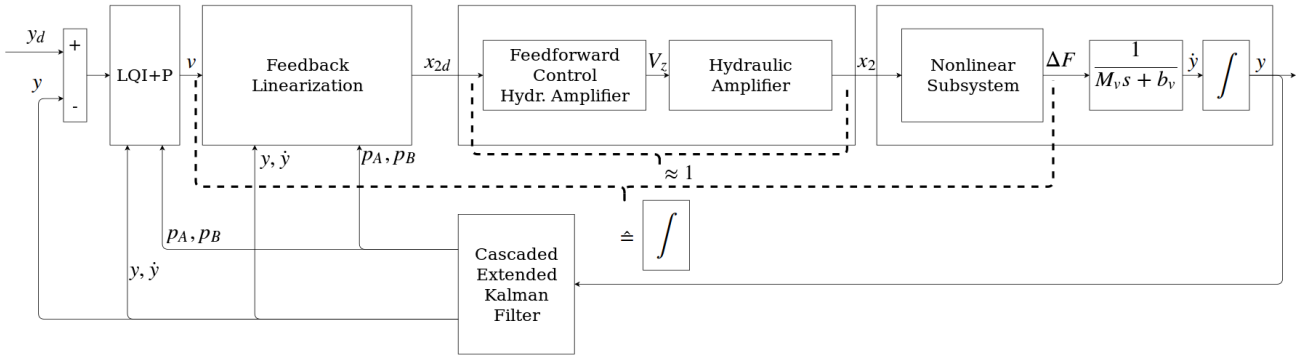


Figure 4.9: Block diagram of the overall control strategy.

4.3.3 LQI control design for the tracking of the engine valve position

The stabilizing control law $v(t)$ is chosen as an LQI control, which consists of a proportional-integral feedback of the tracking error w.r.t. the engine valve position and state feedback. The gain vector $K_{lqi} = \begin{bmatrix} K_x & K_I \end{bmatrix}$ is calculated by Matlab's `lqi(ss(Al,bl,cl,dl),Ql,rl)` command, using the matrices and vectors (see Fig. 4.9)

$$\mathbf{A}_l = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\frac{b_v}{M_v} & \frac{1}{M_v} \\ 0 & 0 & 0 \end{bmatrix}, \ \mathbf{b}_l = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \ \mathbf{b}_d = \begin{bmatrix} 0 \\ -\frac{1}{M_v} \\ 0 \end{bmatrix}, \ \mathbf{c}_l^T = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}, \ d_l = 0 \ , \tag{4.37}$$

which belong to the following state-space representation

$$\mathbf{x}(t) = \begin{bmatrix} y(t) & \dot{y}(t) & F(t) \end{bmatrix}^T, \ \dot{\mathbf{x}}(t) = \mathbf{A}_l \mathbf{x}(t) + b_l v(t) + b_d d(t). \tag{4.38}$$

The controlled output is given by the first state variable $y(t)$, and the design parameters are chosen as

$$\mathbf{Q}_l = \begin{bmatrix} 10^7 & 0 & 0 & 0 \\ 0 & 10^{10} & 0 & 0 \\ 0 & 0 & 10^6 & 0 \\ 0 & 0 & 0 & 10^7 \end{bmatrix}, \ r_l = 10^{-6}, \ P = 10^{12} \ . \tag{4.39}$$

The element $\mathbf{Q}_{l2,2}$ turned out to be the most important to achieve disturbance rejection because the corresponding state, the valve velocity $\dot{y}(t)$, is the first to be affected by the disturbance force. Fig. 4.10 depicts the modelled disturbance force with a maximum value of 500 N acting against the engine valve motion. It is caused by retroactive effect of the compressed exhaust gas in the cylinder. The linear slope is maybe a little idealistic. In any case, between the openings – due to the limitation of the engine valve position at 0 mm, $d(t)$ does not have any effect anyway on the valve motion. Please note that the modelled disturbance is only used for validation in simulation. This a-priori knowledge about the general shape of the signal is not exploited in the control algorithm. This might be done in a future contribution.

4.4 Cascaded extended Kalman filter structure

Since only the pressure dynamics for p_A and p_B of the hydraulic subsystem are nonlinear, a Cascaded EKF structure consisting of a linear KF and an EKF is proposed, which is depicted in Fig. 4.11. This approach has several advantages over an alternative EKF for the whole subsystem – primarily related to the reduced computational effort – as can be clearly seen in Tables 4.1 and 4.2. In the following comparison, the integers n, m, p represent the dimension of the state, output and input vectors of the

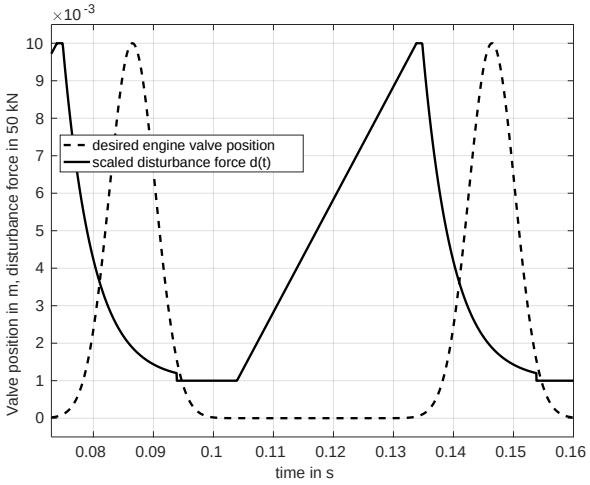


Figure 4.10: Modelled disturbance force at 2000 RPM.

Table 4.1: Kalman estimator: arithmetic operations for a standard EKF structure.

	Number of multiplications/divisions (n=4, m=1, p=1)	Number of additions/substractions (n=4, m=1, p=1)
$\hat{\mathbf{z}}_{k+1}^{+}$	$n^2 + nm$ (20)	$n^2 - n + nm$ (16)
$\mathbf{P}_{k+1}^{+}$	$2n^3$(128)	$2n^3 - n^2$ (112)
$\mathbf{K}_{k+1}^{+}$	$n^2m + 2np^2 + p^3$ (25)	$n^2m + 2np^2 + p^3 - 2np$ (17)
$\hat{\mathbf{z}}_{k+1}^{-}$	$2nm + pm$ (9)	$2nm + pm$ (9)
$\mathbf{P}_{k+1}^{-}$	$n^3 + n^2p$ (80)	$n^3 + n^2p - n^2$ (64)
Total	262	218

standard Kalman filter, whereas n_{NL}, m_{NL}, p_{NL} and n_L, m_L, p_N denote the corresponding dimensions of the cascaded Kalman filters. The number of the individual arithmetic operations are stated in Tables 4.1 and 4.2, please see [74] for further details. In Table 4.2, the contribution 2 pm according to (2) stems from the calculation of the term $\mathbf{z}_{NL,k+1}$ within the Runge-Kutta scheme. Since the developed control strategy is flashed in its entirety into a field-programmable gate array (FPGA) at the test stand and the two cascaded Kalman Filters run in parallel, only the EKF is relevant for a comparison of the computational effort. It shows that one large EKF needs approximately more than twice as long as the small EKF. Considering the small sampling time $T_s = 500\,\text{ns}$, which is necessary to correctly estimate the very fast pressure dynamics, this represents a significant improvement. The linear part of the subsystem (4.12) with $z_1 = y(t)$, $z_2 = \dot{y}(t)$ and $u = \Delta F(t) = (p_A(t) - p_B(t))S$ can be described by the following state-space representation

$$\dot{\mathbf{z}}_L(t) = \mathbf{A}_L\mathbf{z}_L(t) + \mathbf{b}_L u(t),\ y(t) = \mathbf{c}_L^T\mathbf{z}_L(t)\ , \tag{4.40}$$

where the matrix as well as the vectors are as follows

$$\mathbf{A}_L = \begin{bmatrix} 0 & 1 \\ 0 & -\frac{b_v}{M_v} \end{bmatrix}, \mathbf{b}_L = \begin{bmatrix} 0 \\ \frac{1}{M_v} \end{bmatrix}, \mathbf{c}_L^T = \begin{bmatrix} 1 & 0 \end{bmatrix}. \tag{4.41}$$

Here, the disturbance force $d(t)$ has been neglected. A time discretization using the explicit Euler approach, i.e., $\mathbf{A}_{d,L} = \mathrm{I}_{2\times2} + \mathbf{A}_L T_s$ and $\mathbf{b}_{d,L} = \mathbf{b}_L T_s$, allows for an a-priori estimation of the states

$$\hat{\mathbf{z}}_L^{-}(k+1) = \mathbf{A}_{d,L}\hat{\mathbf{z}}_L^{+}(k) + \mathbf{b}_{d,L}u(k) \tag{4.42}$$

Table 4.2: Arithmetic operations for the CEKF structure.

	Number of multiplications/divisions $(n_{NL}=3,\, n_L=2)$ $(m=m_f=m_s=1)$ $(p=p_f=p_s=1)$	Number of additions/substractions $(n_{NL}=3,\, n_L=2)$ $(m=m_f=m_s=1)$ $(p=p_f=p_s=1)$
$\hat{\mathbf{z}}^+_{NL,k+1}$	$n_{NL}^2+n_{NL}m$ (12)	$n_{NL}^2-n_{NL}+n_{NL}m$ (9)
$\hat{\mathbf{z}}^+_{L,k+1}$	$n_L^2+n_Lm$ (6)	$n_L^2-n_L+n_Lm$ (4)
$\mathbf{P}^+_{NL,k+1}$	$2n_{NL}^3$(54)	$2n_{NL}^3-n_{NL}^2$ (45)
$\mathbf{P}^+_{L,k+1}$	$2n_L^3$(16)	$2n_L^3-n_L^2$ (12)
$\mathbf{k}_{NL,k+1}$	$n_{NL}^2m+2n_{NL}p^2+p^3$ (16)	$n_{NL}^2m+2n_{NL}p^2+p^3-2n_{NL}p$ (10)
$\mathbf{k}_{L,k+1}$	$n_L^2m+2n_Lp^2+p^3$ (9)	$n_L^2m+2n_Lp^2+p^3-2n_Lp$ (5)
$\hat{\mathbf{z}}^-_{NL,k+1}$	$2n_{NL}m+pm$ (7)+2pm (2)	$2n_fm+pm$ (7)+2pm (2)
$\hat{\mathbf{z}}^-_{L,k+1}$	$2n_sm+pm$ (5)	$2n_sm+pm$ (5)
$\mathbf{P}^-_{NL,k+1}$	$n_{NL}^3+n_{NL}^2p$ (36)	$n_{NL}^3+n_{NL}^2p-n_{NL}^2$ (27)
$\mathbf{P}^-_{L,k+1}$	$n_L^3+n_L^2p$ (12)	$n_L^3+n_L^2p-n_L^2$ (8)
Total	175	127

$$\mathbf{A}_{d,L}=\begin{bmatrix}1 & T_s\\ 0 & 1-\frac{T_sb_v}{M_v}\end{bmatrix},\mathbf{b}_{d,L}=\begin{bmatrix}0\\ \frac{T_s}{M_v}\end{bmatrix}. \tag{4.43}$$

Furthermore, the covariance matrix of the estimation error can be estimated a-priori

$$\mathbf{P}^-_{L,k+1}=\mathbf{A}_{d,L}\mathbf{P}^+_{L,k}\mathbf{A}^T_{d,L}+\mathbf{Q}_L\ . \tag{4.44}$$

Considering that $\mathbf{c}_L^T\mathbf{P}^-_{L,k+1}\mathbf{c}_L+r_L$ is scalar, the Kalman gain simplifies to

$$\mathbf{k}_{L,k+1}=\frac{\mathbf{P}^-_{L,k+1}\mathbf{c}_L}{\mathbf{c}_L^T\mathbf{P}^-_{L,k+1}\mathbf{c}_L+r_L}\ . \tag{4.45}$$

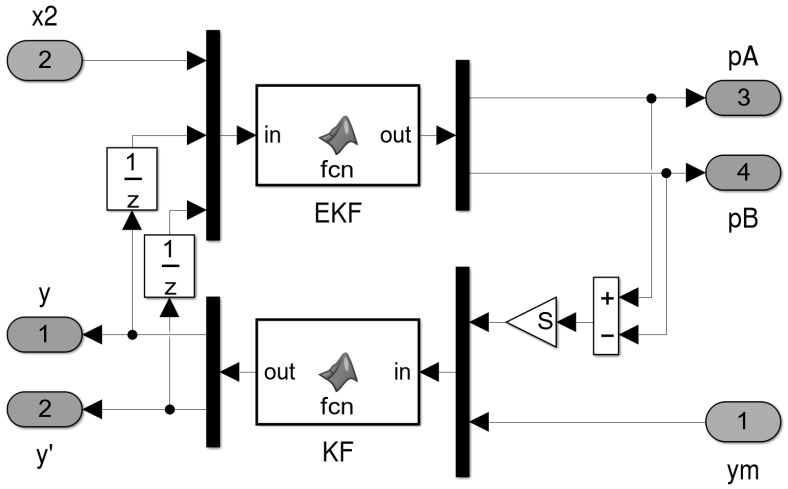


Figure 4.11: Implementation of the CEKF in Simulink. The delay z^{-1} is necessary to avoid an algebraic loop. Given the small sampling time, this is not an issue. The estimates are used within the inverse dynamics of the input-output linearization and the LQI controller.

As soon as a new measurement y_m is available, an a-posteriori estimation of both the states and the covariance of the estimation error becomes possible

$$\hat{\mathbf{z}}_L^+(k+1) = \hat{\mathbf{z}}_L^-(k+1) + \mathbf{k}_{L,k+1}\left(y_m - \mathbf{c}_L^T\hat{\mathbf{z}}_L^-(k+1)\right) \ , \tag{4.46}$$

$$\mathbf{P}_{L,k+1}^+ = \left(\mathbf{I}_{2\times 2} - \mathbf{k}_{L,k+1}\mathbf{c}_L^T\right)\mathbf{P}_{L,k+1}^- \ . \tag{4.47}$$

Please note that – within the CEKF structure – for the EKF described in the following, the engine valve velocity estimated by the KF is used as "measurement" innovation in the update step of the EKF algorithm, and the estimated engine valve position is employed as an additional input. The nonlinear part of (4.12) is then reformulated as $\dot{\mathbf{z}}_{NL}(t) = \mathbf{f}_{NL}(\mathbf{z}_{NL}, \mathbf{u})$. Again neglecting disturbances and setting $z_1 = \dot{y}(t)$, $z_2 = p_A(t)$, $z_3 = p_B(t)$, $u_1 = y(t)$ and $u_2 = x_2(t)$, the right-hand side results in

$$\begin{bmatrix} \frac{1}{M_v}((z_2 - z_3)S - z_1 b_v) \\ \frac{E_{Oil}}{V_{A,0}+u_1 S}(c\ \mathrm{sg}(u_2)\sqrt{p_0 - z_2} - c\ \mathrm{sg}(-u_2)\sqrt{z_2 - p_T} - z_1 S - C_{Li}(z_2 - z_3)) \\ \frac{E_{Oil}}{V_{B,0}-u_1 S}(c\ \mathrm{sg}(-u_2)\sqrt{p_0 - z_3} - c\ \mathrm{sg}(u_2)\sqrt{z_3 - p_T} + z_1 S + C_{Li}(z_2 - z_3)) \end{bmatrix} . \tag{4.48}$$

In the cascaded filter structure, it is necessary that both Kalman filters involve the equation of motion for the engine valve. The continuous-time system $\mathbf{f}_{NL}$ now is time discretized using using a third-order Runge-Kutta scheme, which provides a higher accuracy than the simple Euler approach. This was necessary to obtain an accurately working EKF in simulation. The a-priori state estimate becomes

$$\hat{\mathbf{z}}_{NL}^-(k+1) = \hat{\mathbf{z}}_{NL}^+(k) + T_s\left(\frac{1}{6}\mathbf{f}_{NL,A} + \frac{4}{6}\mathbf{f}_{NL,B} + \frac{1}{6}\mathbf{f}_{NL,C}\right) \ , \tag{4.49}$$

where the evaluation of the right-hand sides are

$$\mathbf{f}_{NL,A} = \mathbf{f}_{NL}(\hat{\mathbf{z}}_{NL}^+(k), \mathbf{u}(k)) \ , \tag{4.50}$$

$$\mathbf{f}_{NL,B} = \mathbf{f}_{NL}(\hat{\mathbf{z}}_{NL}^+(k) + \frac{T_s}{2}\mathbf{f}_{NL,A}, \mathbf{u}(k)) \ , \tag{4.51}$$

$$\mathbf{f}_{NL,C} = \mathbf{f}_{NL}(\hat{\mathbf{z}}_{NL}^+(k) - T_s\mathbf{f}_{NL,A} + 2T_s\mathbf{f}_{NL,B}, \mathbf{u}(k)) \ . \tag{4.52}$$

For the covariance prediction, the continuous-time Jacobian of the system is calculated symbolically

$$\mathbf{J}_{\mathbf{f}_{NL}}(\mathbf{z}_{NL}, \mathbf{u}) = \begin{bmatrix} \frac{\partial f_{NL1}}{\partial z_1} & \frac{\partial f_{NL1}}{\partial z_2} & \frac{\partial f_{NL1}}{\partial z_3} \\ \frac{\partial f_{NL2}}{\partial z_1} & \frac{\partial f_{NL2}}{\partial z_2} & \frac{\partial f_{NL2}}{\partial z_3} \\ \frac{\partial f_{NL3}}{\partial z_1} & \frac{\partial f_{NL3}}{\partial z_2} & \frac{\partial f_{NL3}}{\partial z_3} \end{bmatrix} \ , \tag{4.53}$$

which results in

$$\mathbf{J}_{\mathbf{f}_{NL}} = \begin{bmatrix} -\frac{b_v}{M_v} & \frac{S}{M_v} & -\frac{S}{M_v} \\ -\frac{E_{Oil}S}{V_{A,0}+u_1 S} & J_{22} & \frac{E_{Oil}C_{Li}}{V_{A,0}+u_1 S} \\ \frac{E_{Oil}S}{V_{B,0}-u_1 S} & \frac{E_{Oil}C_{Li}}{V_{B,0}-u_1 S} & J_{33} \end{bmatrix} \ , \tag{4.54}$$

with

$$J_{22} = \frac{E_{Oil}}{V_{A,0} + u_1 S}\left(-C_{Li} - \frac{\frac{1}{2}c\mathrm{sg}(u_2)}{\sqrt{p_0 - z_2}} + \frac{\frac{1}{2}c\mathrm{sg}(-u_2)}{\sqrt{z_2 - p_T}}\right) \tag{4.55}$$

and

$$J_{33} = \frac{E_{Oil}}{V_{B,0} - u_1 S}\left(-C_{Li} + \frac{\frac{1}{2}c\mathrm{sg}(-u_2)}{\sqrt{p_0 - z_3}} - \frac{\frac{1}{2}c\mathrm{sg}(u_2)}{\sqrt{z_3 - p_T}}\right) . \tag{4.56}$$

This Jacobian is discretized as before using a Runge-Kutta scheme except the substitution of $\mathbf{z}_{NL}$ with the a-priori estimates $\hat{\mathbf{z}}_{NL}^-(k+1)$:

$$\mathbf{J}_{\mathbf{f}_{NL},k+1} = \mathrm{I}_{3\times 3} + T_s\left(\frac{1}{6}\mathbf{J}_A + \frac{4}{6}\mathbf{J}_B + \frac{1}{6}\mathbf{J}_C\right) \ . \tag{4.57}$$

The evaluation of the right-hand terms is given by

$$\mathbf{J}_A = \mathbf{J}_{\mathbf{f}_{NL}}(\hat{\mathbf{z}}^-_{NL}(k+1), \mathbf{u}(k)) \ , \tag{4.58}$$

$$\mathbf{J}_B = \mathbf{J}_{\mathbf{f}_{NL}}(\hat{\mathbf{z}}^-_{NL}(k+1) + \frac{T_s}{2}\mathbf{f}_{NL,A}, \mathbf{u}(k)) \ , \tag{4.59}$$

$$\mathbf{J}_C = \mathbf{J}_{\mathbf{f}_{NL}}(\hat{\mathbf{z}}^-_{NL}(k+1) - T_s\mathbf{f}_{NL,A} + 2T_s\mathbf{f}_{NL,B}, \mathbf{u}(k)) \ . \tag{4.60}$$

Finally, the a-priori estimates for the covariance of the estimation errors can be calculated

$$\mathbf{P}^-_{NL,k+1} = \mathbf{J}_{\mathbf{f}_{NL},k+1}\mathbf{P}^+_{NL,k}\mathbf{J}^T_{\mathbf{f}_{NL},k+1} + \mathbf{Q}_{NL} \ . \tag{4.61}$$

In the first time step, $\mathbf{P}^+_{NL,k} = \mathbf{P}^+_{NL,0}$ is a given initial value; otherwise, it represents the a-posteriori estimate from the previous time step. The matrix $\mathbf{Q}_{NL}$ must be specified by the user, and can be considered as a quantification of the model uncertainty in the form of corresponding noise processes. Again, the denominator of Kalman gain is scalar and follows as

$$\mathbf{k}_{NL,k+1} = \frac{\mathbf{P}^-_{NL,k+1}\mathbf{c}_{NL}}{\mathbf{c}^T_{NL}\mathbf{P}^-_{NL,k+1}\mathbf{c}_{NL} + r_{NL}} \ . \tag{4.62}$$

The vector $\mathbf{c}^T_{NL} = [1\ 0\ 0]^T$ specifies which state serves as measurement innovation, in the given case the valve velocity $\dot{y}_{si}$ estimated by the linear KF. The a-posteriori estimation of the states and the covariance of the estimation error follow from

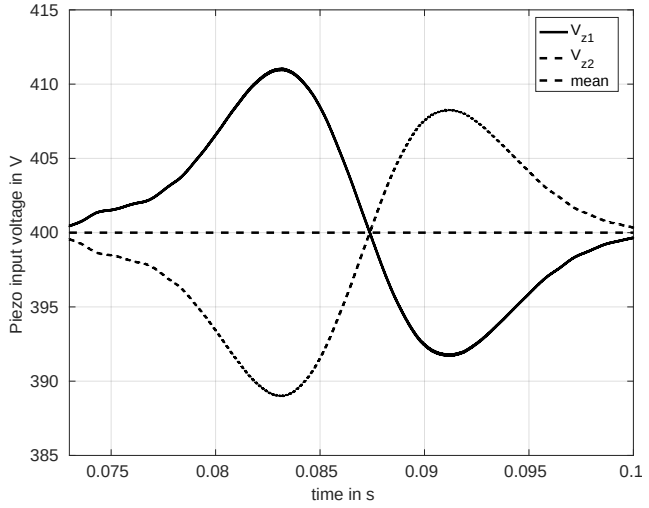
$$\hat{\mathbf{z}}^+_{NL}(k+1) = \hat{\mathbf{z}}^-_{NL}(k+1) + \mathbf{k}_{NL,k+1}\left(\dot{y}_{si} - \mathbf{c}^T_{NL}\hat{\mathbf{z}}^-_{NL}(k+1)\right) \tag{4.63}$$

$$\mathbf{P}^+_{NL,k+1} = \left(\mathbf{I}_{3\times3} - \mathbf{k}_{NL,k+1}\mathbf{c}^T_{NL}\right)\mathbf{P}^-_{NL,k+1} \ . \tag{4.64}$$

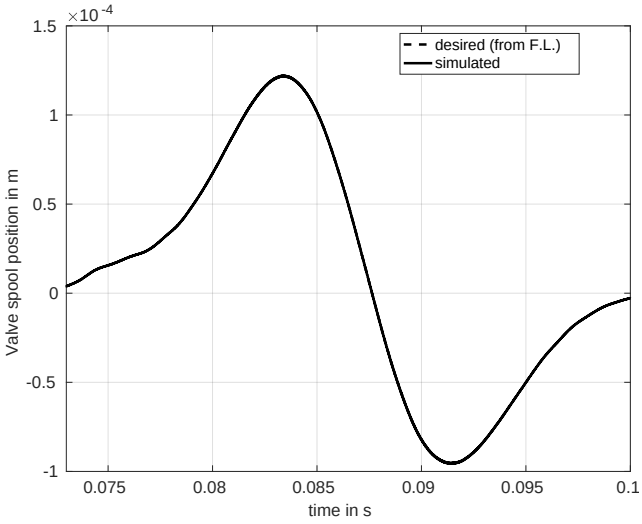
The estimates used by both the input-output linearization and by the linear feedback controller are contained in the vector $\hat{\mathbf{z}}^+_{NL}(k+1)$. The used noise variance parameters r_L (linear KF) and r_{NL} (EKF) have been determined iteratively in extensive simulation studies. They do, of course, depend on the given output measurement noise (or how it is modelled in simulation), including all influences, e.g. quantization errors from the measurement AD conversion. r_{NL} can be chosen very small because the EKF receives the estimated valve velocity from the linear KF as "measurement" innovation. This estimated signal is naturally much cleaner than a real measurement (which is not available). Instead of the employed trial-and-error approach, it is possible to *estimate* the two covariance parameters Q and R for Kalman estimators using Least Squares methods like in [87, 88].

4.5 Simulation results and possible implementation

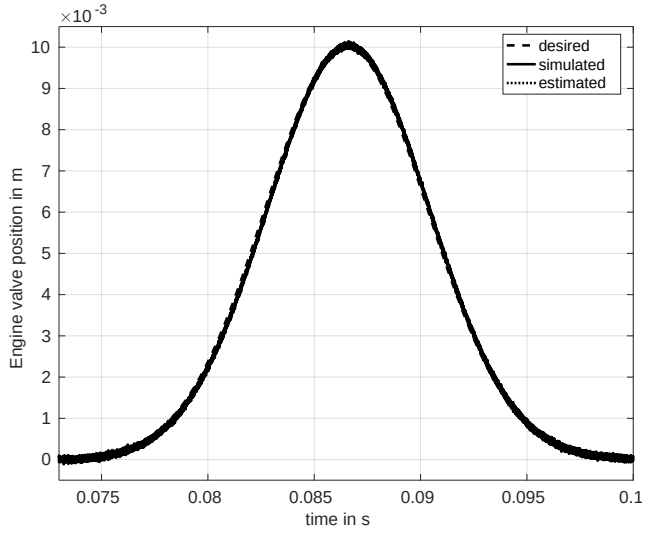
Fig. 4.12c demonstrates the accurate tracking of the engine valve position. The simulation considers an additive white noise in the measured output with a variance of $10^{-9}\,\mathrm{m}^2$. The solid line signal corresponds to the measured output, which is fed into the Kalman filter to estimate all the other states of (4.12). The dotted line signal represents the estimate provided by the Kalman Filter. In Fig. 4.12d, the corresponding velocity is shown. Unfortunately, it cannot be measured at the test stand, which prevents any use of the solid line signal. The dotted line signal is the estimate provided by the Kalman Filter, which is employed instead by the miscellaneous control algorithms. The estimation is quite good, even at the beginning, even though the disturbance is present. An accurate tracking of the desired velocity – unlike the desired position – is not very important, however, the soft landing property is achieved in the given case. Fig. 4.12b shows the valve spool position. Note that the desired and obtained signals are practically identical. The desired valve spool position is calculated by the input-output linearization – the inverse dynamics – and includes estimated feedback signals. Please note as well that this signal has been filtered with $\frac{1}{10^{-3}s+1}$. The curve's asymmetry along the time axis is caused by the presence of the disturbance force – here, the beneficial contribution of the linear feedback



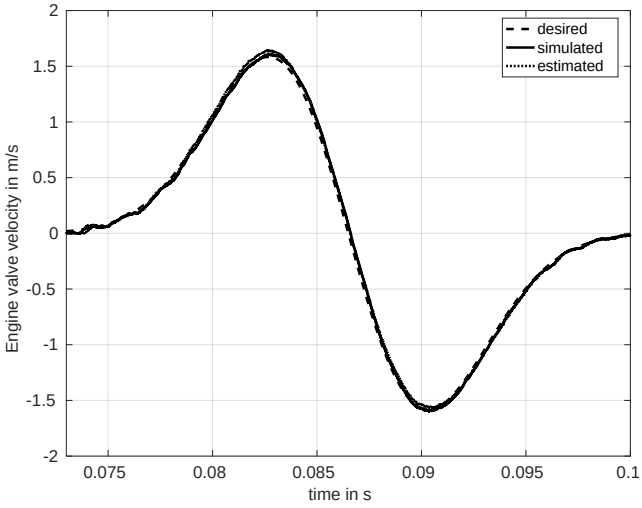
(a) Piezo input voltage at 2000 RPM.



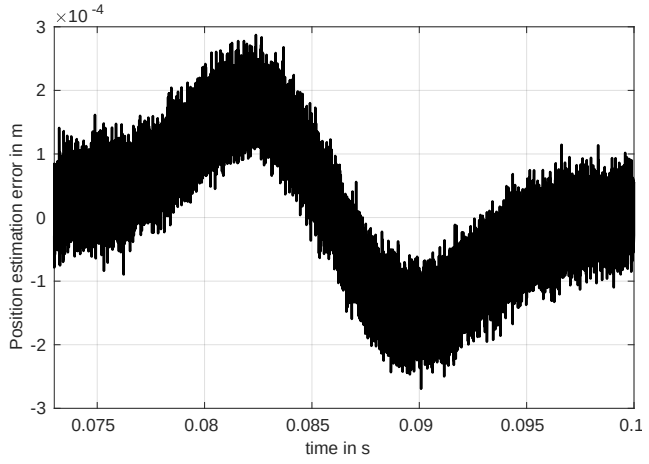
(b) Valve spool position at 2000 RPM.



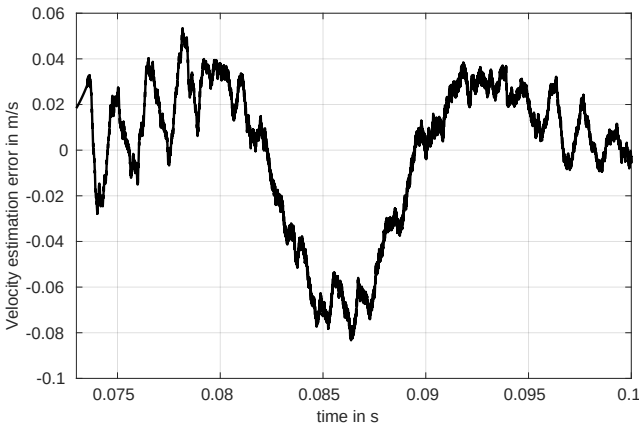
(c) Engine valve position at 2000 RPM.



(d) Engine valve velocity at 2000 RPM.



(e) Position estimation error (sim. − est.).



(f) Velocity estimation error (sim. − est.).

Figure 4.12: Simulation results

controller by means of $v(t)$ becomes visible. Fig. 4.12a depicts the piezo input voltage. These signals were filtered with $\frac{1}{(10^{-3}s+1)^3}$, too, in order to obtain a nice plot. Note that the amplitude is higher in the first half of the curve than in the second half, which is related to the fact that the simulated disturbance is much higher at the beginning than later on. The presented control and estimation architecture was implemented in Matlab/Simulink. As indicated in Fig. 4.11, the CEKF was implemented using "Matlab Function" blocks that can be used to automatically generate embeddable C and VHDL code. Using a DSP system (consisting of a 2.8 GHz quadcore DS1006 processor board and one or more 100 MHz DS5203 FGPA boards) and the corresponding Matlab/Simulink interface software by dSPACE , the model can be flashed in its entirety into dedicated real-time control hardware. Regarding the feasability of implementation, it is possible to justify that the CEKF algorithm could be executed directly on one CPU core of the DS1006 board instead of the FPGA within the chosen sampling time of 500 ns, considering the small clock time of $\frac{1}{2.8\,\mathrm{GHz}} \approx 0.36\,\mathrm{ns}$. The resulting $500\,\mathrm{ns} \times 2.8\,\mathrm{GHz} = 1400$ available clocks should be enough to execute the $175 + 127 = 302$ necessary operations (see Table 4.2) of the CEKF. This simplistic, rather naive calculation neglects optimizations and high IPC of the modern x86 CPU integrated in the DS1006, but also neglects the high scheduling overhead attributable to the real-time operational mode of the embedded system. A traditional EKF, on the other hand, needs $262 + 218 = 480$ operations (see Table 4.1), so it would take approx. 60% longer to execute. It would propbably run within 500 ns on the CPU, too (which was not tested), but the more efficient variant is obviously preferrable. Nevertheless, to exploit the parallel structure, the CEKF algorithm should be implemented in an FPGA, regardless of these considerations. For details on these practical aspects, see [64].

4.6 Outlook

An innovative actuator dedicated for a fully variable valve control in camless combustion engines is considered in this contribution. The actuator consists of piezo-mechanical as well as hydraulic parts, and this contribution extends earlier work by a nonlinear modelling of the hydraulic system part. Accurate tracking of desired valve trajectories as well as soft landing – despite disturbance forces and measurement noise – are achieved using a combined control strategy which consists of an input-output linearisation, an LQI controller and a suitable feedforward control design. Given the very small adopted sampling time, a CEKF is implemented to estimate the state variables required for the control implementation. The CEKF allows to drastically reduce the overall number of operations and thus the computational load. The necessary calculation time is one of the most important requirements for a possible real-time hardware application of the algorithm. Future work may include implementing the presented control architecture, particularly the CEKF, in a DSP system at the test stand to validate the obtained results.

5 Gain adaptation in sliding mode control using model predictive control and disturbance compensation with application to actuators

This chapter considers, as an example application, a DC electric motor that is subject to unknown, nonlinear friction effects or unknown load torques. The main focus is on the control strategy, which aims to be robust and efficient thanks to a sophisticated gain adaptation scheme. The system, modelled in state space, is controlled by a second order sliding mode control (SMC), i.e. with an additional integral term in the sliding surface, increasing the order. SMC is usually characterized by very fast switching between two different control laws of high amplitudes in the system input, so-called *control chattering*. Comparable to very fast bang-bang control, a robust controller with guaranteed stability can be achieved at the price of poor electrical efficiency (due to switching at high frequency, which is problematic in semiconductor-based discrete switches such as transistors). This disadvantage of SMCs can be mitigated or avoided by various methods, especially using the concept of *equivalent control*, unburdening the switching part to a certain extent using continuous signals to compensate for those parts of the sliding surface dynamics that are not uncertain. Additionally, the switching amplitude can be adapted online, instead of tuning it once (conservatively, for all eventualities and disturbances) – this approach is employed here. The adaptation is based on the assumption that the (closed loop, SMC-controlled) system behaves according to the SMC error dynamics. However, this is only the case if there are no disturbances or load torques. For this reason, a KF was designed that, apart from all state variables, estimates such disturbances and also the derivative of the disturbance signal. These two quantities are applied to the input voltage with suitable factors in order to implement the disturbance compensation and to achieve the prescribed system behavior (*model matching*). Once this is achieved, a linear MPC can be designed that adapts the switching amplitude so that it is always large enough to reach the control target, but small enough to minimize the *control chattering*. An interesting property of this strategy is that the model used for the MPC is not application-specific, but can be used for any system that is controllable by SMC, rendering it a "plug and play" adaptation scheme.

5.1 Literature review

5.1.1 Adaptive sliding mode control and model predictive control

While belonging to perhaps the most robust and versatile control strategies, SMC tends to suffer from high energy consumption and high-frequency oscillations in system inputs, states or even outputs, which certainly is to be avoided in tracking problems. Currently, many remedies have been proposed and successfully realized to deal with these problems. A very important one is the so-called boundary layer approach, see e.g. [89], which introduces a permissible region around the sliding surface. This layer is characterized by its thickness, inside which no switching of the control input takes place. Another frequently applied approach to chattering reduction is higher-order SMC [90], which can also be combined with a boundary layer concept [91]. Nevertheless, powerful alternatives exist towards a model-based or signal-based adaptation of the switching amplitudes. A model-based approach is shown in [92], where model reference adaptive control (MRAC) is employed to adjust the SMC gain in an application of a brushless DC motor. The signal-based approaches discussed in the literature are generally based on integral-type scheduling rules, typically case distinctions depending on a norm or an absolute value of the sliding surface, see e.g. [93] and [94]. The contribution at hand looks at an innovative model-based

adaptation. Thanks to countermeasures like these against chattering, SMC is suitable to be widely used in industry. In the context of drive applications, for example, it has even been used to *reduce* torque ripple [95].

MPC still constitutes a developing research field in the context of machines and drives, though many applications already use such control strategies, e.g., in [96] for a permament magnet synchronous machine (PMSM) or in [97] for a DC drive. The MPC approach takes into account model-based predictions and determines the control inputs by minimizing a cost function. However, compared to classical controllers like PID control, this method from the field of optimal control requires a relatively high modeling accuracy in order to yield acceptable results. SMC, on the other hand, is known for being robust against disturbances, model mismatch and parametric uncertainties.

5.1.2 Estimation and Kalman filters

Thanks to the high computing power of modern processors, micro-controllers, or even FPGAs, it has become possible to deploy intelligent and sophisticated control approaches, e.g. observer-based control, utilizing only a minimal number of sensors, see [72, 98]. Contributions like [99–102] reflect the progress in theoretical studies of KFs, especially concerning robustness and the ability to deal with unknown or inaccessible disturbances or model uncertainty. In many situations, time-varying disturbances like friction effects can be modeled as additional unknown inputs. In [103] for instance, a two-stage KF is implemented to estimate the pressure disturbance inside a cylinder of an internal combustion engine and its effect on the controlled output.

5.1.3 Actuators

As important mechatronic components, electromagnetic actuators are used in many technical applications, in particular in the automotive industry and in industrial production systems. In production systems, they play a key role in motion control and precise positioning. Mechanical, pneumatic or hydraulic components tend to be replaced by electromagnetic actuators due to their high efficiency, excellent dynamic behavior and cleanliness. An important effect that needs to be considered in the mechanical part of actuators is nonlinear friction. An extended survey of friction modeling is given in [104] including a large number of literature references. Recent contributions mark progress in terms of identification of friction phenomena and their compensation [105].

5.1.4 Main contribution

In this chapter, a combination of SMC and linear MPC is proposed to create an adaptive control method. Here, LMPC adapts the switching height of the discontinuous control part and, thereby, reduces the undesired chattering effect. The combination of SMC and LMPC allows for an exploitation of the benefits of both worlds, gaining both robustness and a degree of optimality with regard to the specific MPC cost function – at the cost of a more sophisticated control design as well as an increased implementation and computation effort. The introduction of a cost function and, therefore, of an optimality measure allows the intuitive balancing of the error convergence rate versus chattering amplitude penalties, in order to achieve a reasonable trade-off. Linear MPC is a straight-forward, easily implementable way of minimizing such a cost function. Furthermore, an augmented linear KF is employed with the primary goal to contribute to a fast convergence to the sliding surface, thus to unburden the switching control part of the SMC and to reduce undesired chattering. To achieve this goal, the KF estimates the disturbance (and also the first derivative with regard to time), which is considered as an unknown input, and is subsequently used for a compensation in the SMC law. Also, since the cost function that is minimized by the MPC includes the predicted tracking error stemming from a model-based prediction scheme, an accurate model of the SMC sliding surface dynamics is necessary to allow for an optimally small SMC gain. To ensure that the underlying model assumptions of the MPC design are met, a compensation of lumped disturbances is mandatory. Thanks to the disturbance compensation using the estimates of the KF, an accurate prediction over a finite horizon

becomes possible in compliance with the underlying assumptions at the KF design. The disturbance compensation may also be interpreted as a lowering of the necessary minimum SMC gain because part of the disturbances are compensated for by the KF estimates – and the SMC is disburdened. In order to demonstrate the properties of the proposed control method in a practice-relevant field, this work considers a DC drive system that is subject to both nonlinear friction and model uncertainty. The nonlinearity is represented by the sum of the Coulomb friction model and a quadratic term depending on the relative velocity.

To conclude, the goal of this contribution, which represents an extension of a conference paper [106], is to conceive an SMC with optimal adaptivity that can be implemented as simply as possible. Its effectiveness is demonstrated in simulations, subject to realistic conditions regarding disturbances and model uncertainty, controlling the angular velocity of a DC drive including a nonlinear friction model.

The chapter is structured as follows:

- Section 5.2 presents the physically-based model of a DC drive that is affected by a nonlinear friction torque and model uncertainty.
- The feedback control design is described in Section 5.3, where
- Subsection 5.3.1 contains details on the employed SMC techniques which involve a combination of a continuous control action and a discontinuous switching part, and where
- in Subsection 5.3.2, the height of the switching control action is adapted using MPC techniques to counteract undesired chattering.
- Moreover, an unknown lumped disturbance – accounting for nonlinear friction and model uncertainty in the equation of motion – is estimated in Subsection 5.3.3 by a KF. This estimate is employed subsequently in the error dynamics for compensation purposes and, as a result, contributes to the reduction of the necessary switching height determined by MPC.
- Finally, the benefits are shown by meaningful simulation results in Section 5.4.

5.2 System modelling

The system model is based on physical considerations and involves ordinary differential equations for the armature current $i(t)$ and the motor angular velocity $\omega(t)$

$$\frac{\mathrm{d}i(t)}{\mathrm{d}t} = \frac{1}{L}\left(u(t) - Ri(t) - K_T\omega(t)\right), \tag{5.1}$$

$$\frac{\mathrm{d}\omega(t)}{\mathrm{d}t} = \frac{1}{J}\left(K_T i(t) - T_r(\omega(t))\right). \tag{5.2}$$

Here, a nonlinear friction torque

$$T_r(\omega(t)) = \Big(K_f\omega^2(t) + T_{r0}\Big)\,\mathrm{sign}(\omega(t)) \tag{5.3}$$

is introduced, where $K_f > 0$ denotes a coefficient related to the quadratic term in the motor angular velocity, and T_{r0} characterizes the Coulomb friction part. The nonlinear friction torque $T_r(\omega) = T_r(\omega(t))$ is depicted in Fig. 5.1a. This friction model is implemented in a regularized form and used in the simulation studies to represent nonlinear friction.

Given this model description, two alternatives seem to be promising to address nonlinear friction:

1. Feedback disturbance compensation: In this solution, the friction term (5.3) is assumed as known and explicitly included in the sliding mode control design. The corresponding parameters are identified beforehand by the least-squares method. In the envisaged sliding-mode control design this would involve a time differentiation of the friction model and a compensation by means of feedback. It is clear that any changes of the friction model afterwards results in an imperfect compensation.

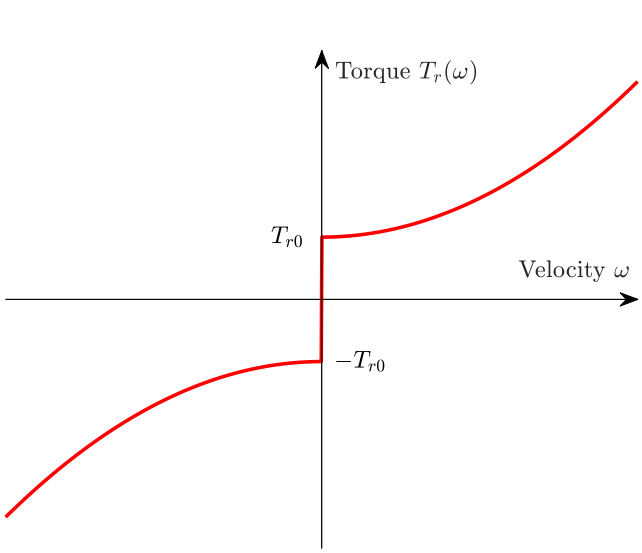
2. Estimator-based disturbance compensation: In this approach, the detailed physical model for the nonlinear friction (5.3) is not employed at all in the control design. Instead, nonlinear friction is estimated by a Kalman filter. It turns out that the approach can be generalized by considering a lumped disturbance torque $d(t) = T_r(\omega(t)) + T_u(t)$, where $T_u(t)$ represents any further external loads torques, unmodelled dynamic effects and model uncertainty. The estimate can be subsequently used for a disturbance compensation. The modified system model is then given by

$$\frac{\mathrm{d}i(t)}{\mathrm{d}t} = \frac{1}{L}\left(u(t) - Ri(t) - K_T\omega(t)\right), \tag{5.4}$$

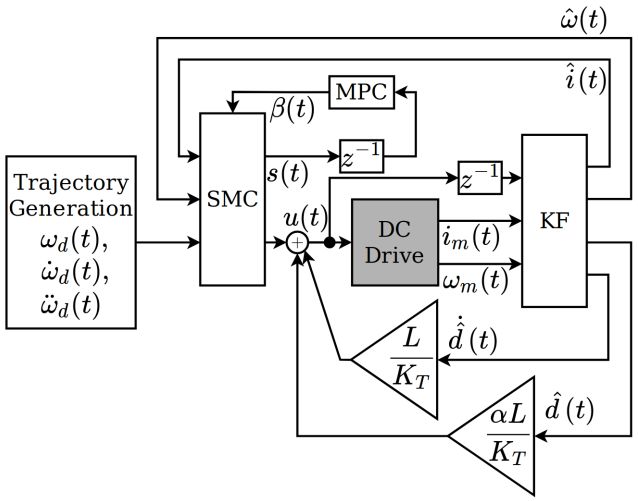
$$\frac{\mathrm{d}\omega(t)}{\mathrm{d}t} = \frac{1}{J}\left(K_T i(t) - d(t)\right). \tag{5.5}$$

In the sequel, the latter approach using a KF is followed because it promises a higher tracking accuracy. Moreover, the estimator dynamics can be specified appropriately in the KF design.

5.3 Control design



(a) Nonlinear friction characteristic $T_r(\omega)$.



(b) Implementation of the SMC in combination with a KF for state and parameter estimation and an MPC for the adaptation of the switching height. The gray block represents the DC drive.

Figure 5.1

The implementation of the overall control structure corresponds to the block diagram shown in Fig. 5.1b. The control input involves an equivalent control action, a disturbance compensation and a robustifying switching term. In this contribution, the switching height is adapted by means of a quasi-linear MPC.

5.3.1 Feedback control design using SMC

Since the lumped disturbance $d(t)$ is estimated by a KF and used for a subsequent disturbance compensation, the state-space representation (5.4) and (5.5) can be used in the derivation of an integral SMC. It is worth mentioning that the estimated disturbance compensates the largest part of the lumped disturbance and significantly increases the tracking accuracy. As a result, the SMC has to cope with model imperfections that are related to the dynamics of the estimator only – leading to a significant reduction of the necessary switching height and, thereby, reducing the undesired chattering effect. The SMC design aims at a highly accurate tracking of desired trajectories for the angular velocity with

smallest possible tracking errors $e(t) := \omega_d(t) - \omega(t)$. For that purpose, an integral sliding surface is introduced as follows

$$s(t) = \dot{e}(t) + \alpha e(t) + \eta \int_0^t e(\tau)\,\mathrm{d}\tau - e(0), \tag{5.6}$$

where $e(0)$ represents an initial error. Its presence in $s(t)$ could eliminate the reaching phase ($s(0) = 0$), see [107]. In this contribution, however, $e(0)$ is assumed to be unknown and is set to zero in the implementation. The coefficients $\alpha \in \mathbb{R}$ and $\eta \in \mathbb{R}$ have to be positive. The time derivative of the sliding surface can be easily computed and results in

$$\begin{aligned}
\dot{s}(t) &= \ddot{\omega}_d(t) - \ddot{\omega}(t) + \alpha(\dot{\omega}_d(t) - \dot{\omega}(t)) + \eta\,(\omega_d(t) - \omega(t)) \\
&= \ddot{\omega}_d(t) - \left(\frac{K_T}{J}\frac{di(t)}{dt} - \frac{1}{J}\dot{d}(t)\right) + \alpha\left[\dot{\omega}_d(t) - \left(\frac{K_T}{J}i(t) - \frac{1}{J}d(t)\right)\right] + \eta\,(\omega_d(t) - \omega(t)) \\
&= \ddot{\omega}_d(t) - \left[\frac{K_T}{J}\left(\frac{1}{L}u(t) - \frac{R}{L}i(t) - \frac{K_T}{L}\omega(t)\right) - \frac{1}{J}\dot{d}(t)\right] + \alpha\left[\dot{\omega}_d(t) - \left(\frac{K_T}{J}i(t) - \frac{1}{J}d(t)\right)\right] \\
&\quad + \eta\,(\omega_d(t) - \omega(t)) \\
&= \ddot{\omega}_d(t) - \frac{K_T}{JL}u(t) + \frac{K_T R}{JL}i(t) + \frac{K_T^2}{JL}\omega(t) + \frac{1}{J}\dot{d}(t) + \alpha\left[\dot{\omega}_d(t) - \frac{K_T}{J}i(t) + \frac{1}{J}d(t)\right] \\
&\quad + \eta\,(\omega_d(t) - \omega(t))\,.
\end{aligned} \tag{5.7}$$

It becomes obvious that the time derivative $\dot{d}(t)$ of the lumped disturbance affects the time derivative of the sliding surface $\dot{s}(t)$ – and, hence, is needed in the SMC law. This explains why it is estimated as well using a Kalman filter with a suitable disturbance model. For the derivation of the SMC law, a quadratic Lyapunov function candidate based on the integral sliding surface is considered

$$V(t) = \frac{1}{2}s(t)^2. \tag{5.8}$$

The time derivative of the Lyapunov function candidate can be easily calculated. It has to fulfill the sliding condition, which is chosen as follows in this work

$$\dot{V}(t) = s(t)\dot{s}(t) \leq s(t)\,(-\lambda s(t) - \beta\mathrm{sgn}(s(t))) = -\lambda s(t)^2 - \beta|s(t)|. \tag{5.9}$$

Now, all known terms in the time derivative $\dot{s}(t)$ are compensated for by feedback, which leads to the following expression for the equivalent control

$$u_{eq}(t) = \frac{JL}{K_T}\left[\ddot{\omega}_d(t) + \frac{K_T R}{JL}i(t) + \frac{K_T^2}{JL}\omega(t) + \alpha\left[\dot{\omega}_d(t) - \frac{K_T}{J}i(t)\right] + \eta\,(\omega_d(t) - \omega(t))\right]. \tag{5.10}$$

Then, the time derivative of the sliding surface becomes

$$\dot{s} = -\frac{K_T}{JL}[u(t) - u_{eq}(t)] + \frac{1}{J}\dot{d}(t) + \frac{\alpha}{J}d(t). \tag{5.11}$$

The unknown disturbance as well as its time derivative are estimated by a Kalman filter. A disturbance compensation based on these estimates $\hat{\dot{d}}$ and $\hat{d}$, hence, contributes to a reduction of the discontinuous switching term, i.e., to a suppression of undesired chattering. The disturbance compensation law results in

$$u_{dc}(t) = \frac{L}{K_T}\hat{\dot{d}}(t) + \frac{\alpha L}{K_T}\hat{d}(t). \tag{5.12}$$

Finally, the switching part $u_{sw}(t)$ can be derived from the sliding condition

$$\dot{s}(t) = \frac{-K_T}{JL}u_{sw}(t) \leq -\lambda s(t) - \beta\mathrm{sgn}(s(t)). \tag{5.13}$$

If the equality sign holds, the switching part is given by

$$u_{sw}(t) = \frac{JL}{K_T} \left(\lambda s(t) + \beta \text{sgn}(s(t))\right). \tag{5.14}$$

The overall SMC law comprises the sum of all three terms

$$u(t) = u_{eq}(t) + u_{dc}(t) + u_{sw}(t). \tag{5.15}$$

Outside the boundary layer – during the convergence to the sliding surface – the time derivative $\dot{s}(t)$ is governed by the nonlinear error dynamics

$$\dot{s}(t) = -\lambda s(t) - \beta \text{sgn}(s(t)). \tag{5.16}$$

5.3.2 Adaption of the switching height using MPC

The main idea of this contribution is now to use MPC techniques to determine an optimal switching height $\beta(k)$. For that purpose, the error dynamics is discretized with regard to time using the explicit Euler method with a sampling time of $T_s = 10\,\mu\text{s}$, and the switching height $\beta(k)$ is introduced as control input for the MPC

$$s(k+1) = (1 - \lambda T_s)\, s(k) - \beta(k) T_s \text{sgn}(s(k)). \tag{5.17}$$

Converge properties outside the boundary layer

Outside the boundary layer, from (5.17), a possible discrete-time state-space representation results

$$\begin{aligned} s(k+1) &= a_{k,r} s(k) + b_{k,r} \beta(k), \quad y(k) = c_k s(k), \\ &\Rightarrow a_{k,r} = 1 - \lambda T_s,\ b_{k,r} = -T_s \text{sgn}(s(k)),\ c_k = c = 1, \end{aligned} \tag{5.18}$$

where subscript r indicates the reaching phase characteristics. By repeated evaluations of the difference equation, the system behavior can be predicted as

$$\begin{aligned} \hat{y}(k+1) &= c\ a_{k,r}\ s(k) + c\ b_{k,r}\ \beta(k) \\ \hat{y}(k+2) &= c\ a_{k,r}^2\ s(k) + c\ a_{k,r}\ b_{k,r}\ \beta(k) + c\ b_{k+1,r}\ \beta(k+1), \\ \hat{y}(k+3) &= c\ a_{k,r}^3\ s(k) + c\ a_{k,r}^2\ b_{k,r}\ \beta(k) + c\ a_{k,r}\ b_{k+1,r}\ \beta(k+1) + c\ b_{k+2,r}\ \beta(k+2), \end{aligned} \tag{5.19}$$

etc. It is straightforward to show that the following vector expression holds

$$\hat{\mathbf{y}}(k) = \mathbf{g}_r s(k) + \mathbf{F}_{k,r} \mathbf{u}_k, \tag{5.20}$$

with

$$\hat{\mathbf{y}}(k) = \begin{bmatrix} \hat{y}(k+1) \\ \hat{y}(k+2) \\ ... \\ \hat{y}(k+p) \end{bmatrix}, \quad \mathbf{u}_k = \begin{bmatrix} \beta(k) \\ \beta(k+1) \\ ... \\ \beta(k+p-1) \end{bmatrix}, \tag{5.21}$$

and a prediction horizon of length p. The system matrices for use in the MPC become

$$\mathbf{g}_r = \begin{bmatrix} c\ a_{k,r} \\ c\ a_{k,r}^2 \\ ... \\ c\ a_{k,r}^p \end{bmatrix}, \quad \mathbf{F}_{k,r} = \begin{bmatrix} cb_{k,r} & \mathbf{0} & ... & \mathbf{0} \\ ca_{k,r} b_{k,r} & cb_{k+1,r} & ... & \mathbf{0} \\ ... & ... & ... & ... \\ ca_{k,r}^{p-1} b_{k,r} & ca_{k,r}^{p-2} b_{k+1,r} & ... & cb_{k+p-1,r} \end{bmatrix}, \tag{5.22}$$

where the prediction horizon p should not be chosen as too large, considering that $b_{k,r}$ might change unpredictably. For $p = 2$, these matrices simplify to

$$\mathbf{g}_r = \begin{bmatrix} 1 - \lambda T_s \\ (1 - \lambda T_s)^2 \end{bmatrix}, \ \mathbf{F}_{k,r} = -T_s \begin{bmatrix} \text{sgn}(s(k)) & 0 \\ (1 - \lambda T_s)\, \text{sgn}(s(k)) & \text{sgn}(s(k+1)) \end{bmatrix}. \tag{5.23}$$

With given $\mathrm{sgn}(s(k))$, $\mathrm{sgn}(s(k+1))$ can be obtained from the prediction step (5.18) using the second element of the input vector $\mathbf{u}_{k-1}$ computed in the previous step. Given the system according to (5.20), an optimal input $\beta(k)$ has to be calculated that minimizes the following cost function

$$J(k) = \frac{1}{2}\Big(\mathbf{y}_d(k) - \hat{\mathbf{y}}(k)\Big)^{\mathrm{T}} \mathbf{Q}\Big(\mathbf{y}_d(k) - \hat{\mathbf{y}}(k)\Big) + \frac{1}{2}\mathbf{u}_k^{\mathrm{T}}\mathbf{R}\mathbf{u}_k, \tag{5.24}$$

where $\mathbf{Q} \geq 0$ and $\mathbf{R} > \mathbf{0}$ are symmetric non-negative definite matrices. Moreoever, $\mathbf{y}_d(k)$ is the sliding surface reference trajectory for the next p time steps. In this case, its elements can simply be set to zero. The corresponding solution can be stated in closed form

$$\mathbf{u}_k = (\mathbf{F}_{k,r}^T \mathbf{Q}\mathbf{F}_{k,r} + \mathbf{R})^{-1}\mathbf{F}_{k,r}^T\mathbf{Q}(\mathbf{y}_d(k) - \mathbf{g}y(k)), \tag{5.25}$$

where $y(k) = s(k)$ holds and the sliding mode control switching gain $\beta = \beta(k)$ is now chosen as the first element of $\mathbf{u}_k$. The MPC with the predicted sliding surface is realized with matrices (5.18) that do not depend on any specific systems parameters. As a result, this approach offers an intrinsic robustness regarding the prediction.

Convergence properties inside the boundary layer

As a second measure against chattering — in addition to the disturbance compensation by means of the Kalman filter – a regularized version of the switching law is employed. This leads to the definition of a boundary layer resulting from the replacement of $\mathrm{sgn}(s(k))$ by a smoothed version given by the saturation function $\mathrm{sat}\left(\frac{s(k)}{\Phi}\right)$. Inside the boundary layer, the saturation function is linear, resulting in $\mathrm{sat}\left(\frac{s(k)}{\Phi}\right) = \frac{s(k)}{\Phi}$:

$$s(k+1) = (1 - T_s\lambda)\, s(k) - T_s\frac{s(k)}{\Phi}\beta(k). \tag{5.26}$$

This expression contains a multiplication of a state variable with the input variable and is, hence, a nonlinear term. A first-order multivariate Taylor linearization of the function $f(\mathbf{x}_T) = \frac{s(k)}{\Phi}\beta(k)$, where the vector $\mathbf{x}_T = \begin{bmatrix}\beta(k) & s(k)\end{bmatrix}^T$ denotes the independent variables, can be performed in the operating point $\mathbf{x}_T^* = \begin{bmatrix}\beta^*(k-1) & s^*(k-1)\end{bmatrix}^T$, which contains known values at the discrete point at time $k-1$. Here, the star symbol $(\cdot)^*$ denotes the operating point and allows for representing the corresponding values in the following equations. The Taylor series expansion up to linear terms becomes

$$f(\mathbf{x}_T) \approx f(\mathbf{x}_T^*) + \nabla f(\mathbf{x}_T^*)\,(\mathbf{x}_T - \mathbf{x}_T^*)\,, \tag{5.27}$$

where the operator ∇ indicates the gradient of f, a row vector. A detailed description can be stated as follows

$$\frac{s(k)}{\Phi}\beta(k) \approx \frac{s^*(k-1)}{\Phi}\beta^*(k-1) + \frac{s^*(k-1)}{\Phi}\left(\beta(k) - \beta^*(k-1)\right) + \frac{\beta^*(k-1)}{\Phi}\left(s(k) - s^*(k-1)\right) \tag{5.28}$$

$$= \frac{s^*(k-1)}{\Phi}\beta(k) + \frac{\beta^*(k-1)}{\Phi}s(k) - \frac{s^*(k-1)\beta^*(k-1)}{\Phi}. \tag{5.29}$$

By substituting this Taylor linearization into the difference equation, a linear first-order discrete-time model can be derived

$$s(k+1) = \underbrace{\left(1 - T_s\lambda - T_s\frac{\beta^*(k-1)}{\Phi}\right)}_{a_k} s(k) + \underbrace{\frac{-T_s s^*(k-1)}{\Phi}}_{b_k}\beta(k) + w^*(k-1), \tag{5.30}$$

where $w^*(k-1) = \frac{T_s}{\Phi}s^*(k-1)\beta^*(k-1)$ represents a known term consisting on past information, i.e., from the previous time step. The discrete-time system matrix and the input vector become scalars and are given by a_k and b_k, respectively. Note that these terms are time-dependent. The output equation is

scalar as well and can be stated as $y(k) = c_k\ s(k),\ c_k = c = 1$. In analogy to the approach outside the boundary layer, the design of a quasi-linear MPC is presented that is based on the discrete-time model above. The moving prediction horizon comprises two steps in the future. With the following vectors and matrices

$$\mathbf{g}_k = \begin{bmatrix} a_k \\ a_k a_{k+1} \end{bmatrix} = \begin{bmatrix} 1 - T_s\lambda - T_s\frac{\beta^*(k-1)}{\Phi} \\ \left(1 - T_s\lambda - T_s\frac{\beta^*(k-1)}{\Phi}\right)\left(1 - T_s\lambda - T_s\frac{\beta^*(k)}{\Phi}\right) \end{bmatrix}, \tag{5.31}$$

$$\mathbf{F}_k = \begin{bmatrix} cb_k & 0 \\ ca_k b_k & cb_{k+1} \end{bmatrix} = -\frac{T_s}{\Phi}\begin{bmatrix} s^*(k-1) & 0 \\ \left(1 - T_s\lambda - T_s\frac{\beta^*(k-1)}{\Phi}\right)s^*(k-1) & s^*(k) \end{bmatrix}, \tag{5.32}$$

$$\mathbf{w}_k = \begin{bmatrix} c \\ ca_k + c \end{bmatrix} = \begin{bmatrix} 1 \\ 2 - T_s\lambda - T_s\frac{\beta^*(k-1)}{\Phi} \end{bmatrix}, \tag{5.33}$$

the solution can be determined in a closed-form expression as follows

$$\mathbf{u}_k = (\mathbf{F}_k^{\mathrm{T}}\mathbf{Q}\mathbf{F}_k + \mathbf{R})^{-1}\mathbf{F}_k^{\mathrm{T}}\mathbf{Q}\left[\mathbf{y}_d(k) - \mathbf{g}_k y(k) - \mathbf{w}_k w^*(k-1)\right]. \tag{5.34}$$

Here, the output $y(k) = s(k)$ is identical to the current value of the sliding surface, whereas the time-varying SMC switching gain β is chosen again as the first element of the computed input vector $\mathbf{u}_k = \begin{bmatrix}\tilde{\beta}_k(k) & \tilde{\beta}_k(k+1)\end{bmatrix}^T$. As the input $\beta^*(k)$, which shows up in $\mathbf{g}_k$ and represents the linearization point for β, is not yet available, it is substituted by the second element of the input $\mathbf{u}_{k-1}$, which corresponds to previous time step.

5.3.3 KF for the estimation of a lumped disturbance torque

In the sequel, the combined estimation of the state variables and the external disturbance as well as its time derivative is described. The design of a corresponding KF is based on the modified system model, including a double-integrator disturbance model

$$\frac{\mathrm{d}i(t)}{\mathrm{d}t} = \frac{1}{L}\left(u(t) - Ri(t) - K_T\omega(t)\right) \tag{5.35}$$

$$\frac{\mathrm{d}\omega(t)}{\mathrm{d}t} = \frac{1}{J}\left(K_T i(t) - d(t)\right)) \tag{5.36}$$

$$\frac{\mathrm{d}d(t)}{\mathrm{d}t} = \dot{d}(t) \tag{5.37}$$

$$\frac{\mathrm{d}\dot{d}(t)}{\mathrm{d}t} = 0 \tag{5.38}$$

and aims at providing estimates for both state variables, an estimate $\hat{d}$ for the unknown lumped disturbance and its derivative $\dot{\hat{d}}$. It is worth mentioning that the chain of two integrators has no input so far. Nevertheless, this integrator chain is driven by the output error feedback as well as the process noise — the stochastic part — in the framework of the KF design. As a result, the estimator states vary during the operation of the KF and highly accurate estimates are obtained for a subsequent compensation in the control structure.

Whenever feedback control is applied, it is necessary to measure selected system outputs. Under realistic conditions, however, measurements are affected by errors like deterministic offsets and stochastic disturbances, e.g. white noise processes. In such cases, a KF can be advantageously employed and provides estimates with minimum covariances. The optimality conditions include an accurate system model and the knowledge about the noise characteristics. In the given case, the system model (5.4) and (5.5) — that contains a perfectly-known part and the unknown lumped disturbance – is both complete and correct. As confirmed by the simulation results, the quality of the estimates is high and, hence, an accurate system model is obtained due to the estimates. The typical design of a KF addresses uncorrelated process noise and measurement noise that are assumed to be Gaussian, white and with

a vanishing mean value. Despite the fact that in practice the stochastic noise processes are often not perfectly known, the KF algorithm is usually still capable of providing meaningful state and disturbance estimates. The covariances can then be considered as tuning parameters like in the LQR control design. The model defining the KF prediction step can be stated in state-space form, with the input variable $u(t)$,

$$\mathbf{x}_{KF}(t) = \left[[i(t) \quad \omega(t) \quad d(t) \quad \dot{d}(t) \right]^T, \tag{5.39}$$

$$\dot{\mathbf{x}}_{KF}(t) = \mathbf{A}_{KF}\mathbf{x}_{KF}(t) + \mathbf{b}_{KF}u(t), \tag{5.40}$$

$$\mathbf{y}_m(t) = \mathbf{C}_{KF}\mathbf{x}_{KF}(t) \tag{5.41}$$

with matrices

$$\mathbf{A}_{KF} = \begin{bmatrix} -\frac{R}{L} & -\frac{K}{L} & 0 & 0 \\ \frac{K}{J} & 0 & -\frac{1}{J} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{b}_{KF} = \begin{bmatrix} \frac{1}{L} \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{C}_{KF} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \tag{5.42}$$

The discrete-time state space model can be obtained using explicit Euler discretization with step width T_s, after which the random variables $\mathbf{w}_{KF}$ and $\mathbf{v}_{KF}$ are introduced to represent white, uncorrelated process and measurement noise, respectively, with normal probability distributions [7].

$$\mathbf{x}_{KF}(k+1) = \mathbf{A}_{KFd}\mathbf{x}_{KF}(k) + \mathbf{b}_{KFd}u(k) + \mathbf{w}_{KF}(k), \tag{5.43}$$

$$\mathbf{y}_{KF}(k) = \mathbf{C}_{KF}\mathbf{x}_{KF}(k) + \mathbf{v}_{KF}(k), \tag{5.44}$$

$$\mathbf{A}_{KFd} = \mathbf{I}_{4\times4} + T_s\mathbf{A}_{KF}, \quad \mathbf{b}_{KFd} = T_s\mathbf{b}_{KF}.$$

Based on this, the a-priori estimates are calculated in the prediction step of the Kalman filter algorithm according to

$$\hat{\mathbf{x}}^-_{KF}(k+1) = \mathbf{A}_{KFd}\hat{\mathbf{x}}^+_{KF}(k) + \mathbf{b}_{KFd}u(k). \tag{5.45}$$

For the first step, initial values $\hat{\mathbf{x}}^+_{KF}(0)$ can be either specified by the user or simply set to zero. The same applies to the initial uncertainty $\mathbf{P}^+(0)$ in the following equation. The a-priori estimate of the covariance matrix is

$$\mathbf{P}^-(k+1) = \mathbf{A}_{KFd}\mathbf{P}^+(k)\mathbf{A}^T_{KFd} + \mathbf{Q}_{KF}, \tag{5.46}$$

where $\mathbf{Q}_{KF}$ represents a $\mathbb{R}^{4\times4}$ matrix quantifying the covariance matrix of the process noise $\mathbf{w}_{KF}$,

$$\mathbf{Q}_{KF}(k) = \mathrm{E}\left(\mathbf{w}_{KF}(k)\mathbf{w}_{KF}(k)^T\right) = \begin{bmatrix} q_i & 0 & 0 & 0 \\ 0 & q_\omega & 0 & 0 \\ 0 & 0 & q_d & 0 \\ 0 & 0 & 0 & q_{\dot{d}} \end{bmatrix}. \tag{5.47}$$

Here, $\mathbf{Q}_{KF}$ is assumed to be constant, diagonal and positive and is treated as a tuning parameter matrix. Parameter q_d is set to zero since equation (5.37) is a certain relationship, while $q_{\dot{d}}$ is assigned a large value since equation (5.38) is not. Parameters q_i and q_ω reflect the modeling uncertainty concerning Eqs. (5.36), (5.35). The Kalman gain can now be calculated as

$$\mathbf{K}(k+1) = \mathbf{P}^-(k+1)\mathbf{C}^T_{KF}(\mathbf{C}_{KF}\mathbf{P}^-(k+1)\mathbf{C}^T_{KF} + \mathbf{R}_{KF})^{-1}, \tag{5.48}$$

with the measurement matrix $\mathbf{C}_{KF}$ according to the measured outputs $\mathbf{y}_m(t) = \mathbf{C}\mathbf{x}_{KF}(t)$. Here, the measurement covariance matrix $\mathbf{R}_{KF}$ is related to the measurement noise $\mathbf{v}_{KF}$ and has properties similar to $\mathbf{Q}_{KF}$

$$\mathbf{R}_{KF} = \mathrm{E}\left(\mathbf{v}_{KF}(k)\mathbf{v}_{KF}(k)^T\right) = \begin{bmatrix} r_i & 0 \\ 0 & r_\omega \end{bmatrix}. \tag{5.49}$$

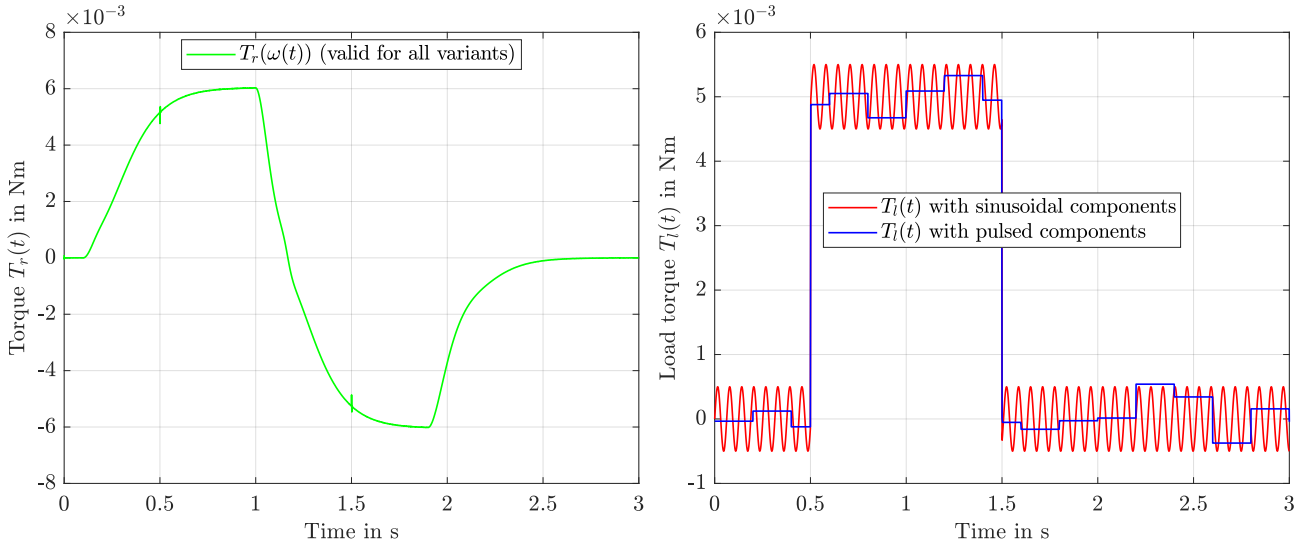
In the correction step of the Kalman filter algorithm, the a-posteriori estimates for covariance and states are calculated as follows

$$\mathbf{P}^{+}(k+1) = (\mathrm{I}_{4\times4} - \mathbf{K}(k+1)\mathbf{C}_{KF})\,\mathbf{P}^{-}(k+1), \tag{5.50}$$

$$\hat{\mathbf{x}}^{+}_{KF}(k+1) = \hat{\mathbf{x}}^{-}_{KF}(k+1) + \mathbf{K}(k+1)\left(\mathbf{y}_m(k+1) - \mathbf{C}_{KF}\hat{\mathbf{x}}^{-}_{KF}(k+1)\right), \tag{5.51}$$

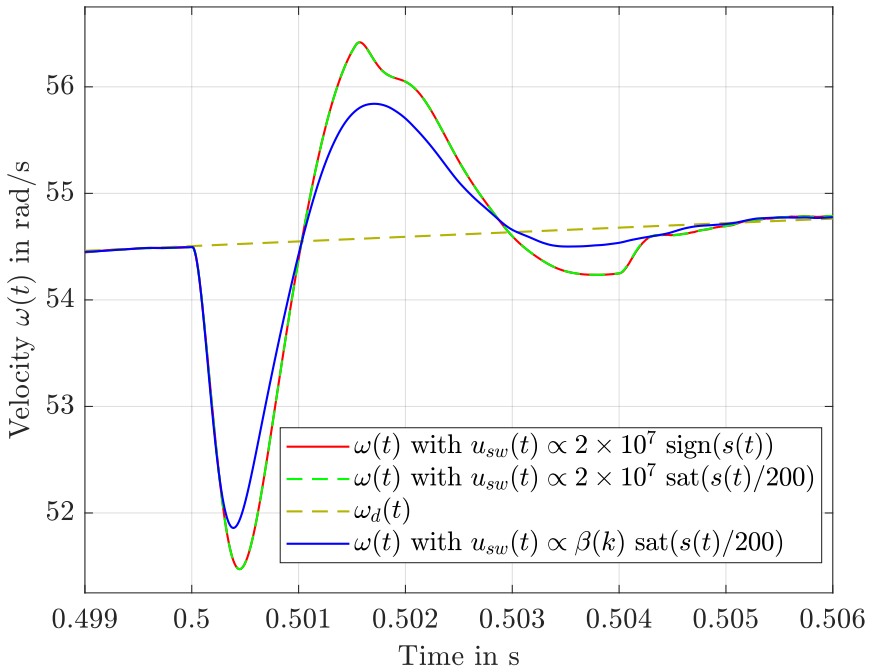
with the measured current and velocity $\mathbf{y}_m = \begin{bmatrix} i_m & \omega_m \end{bmatrix}^T$. The estimated states as well as the estimated disturbance and its time derivative are used in the SMC control law.

5.4 Simulations



(a) Friction torque T_r.

(b) load torque T_l.



(c) Detailed view of the velocity tracking after load torque steps.

Figure 5.2: Friction and load torques and their influence on the velocity tracking.

In this section, simulation results are presented. They were obtained using a sampling time of $T_s = 10\,\mu s$. The velocity profile to be tracked was generated from step-like signals using a second-order low-pass command-shaping filter $1/(0.01s^2 + 0.2s + 1)$.

5.4.1 Simulation settings and scenarios

The covariance matrices and initial conditions of the Kalman filter were chosen according to

$$\mathbf{R}_{KF} = \mathrm{diag}\,(0.001, 500)\,, \quad \mathbf{Q}_{KF} = \mathrm{diag}\,(0.001, 0.001, 0, 0.5)\,, \quad \mathbf{P}^{+}(0) = \mathrm{diag}\,(10^3, 10^3, 0, 10^3)\,, \quad \hat{\mathbf{x}}^{+}_{KF}(0) = \mathbf{0}, \tag{5.52}$$

where the third and fourth element on the diagonal of $\mathbf{Q}_{KF}$ relate to the double integrator disturbance model. The proposed control strategy, i.e., an adaption of the SMC switching height β by means of MPC, is compared with two other, more classical variants. These three variants are as follows:

1. SMC with $\beta = \text{const.}$ and switching control law (5.14),
2. SMC with $\beta = \text{const.}$ and $u_{sw} = \frac{JL}{K_T}\left(\lambda s(t) + \beta \mathrm{sat}(s(t)/\Phi)\right)$,
3. SMC with adaptive $\beta = \beta(k)$ and $u_{sw} = \frac{JL}{K_T}\left(\lambda s(t) + \beta(k) \mathrm{sat}(s(t)/\Phi)\right)$, i.e., the proposed strategy.

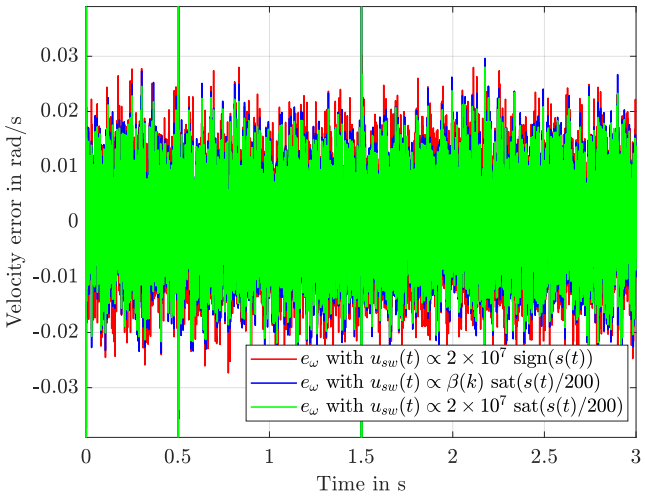
The SMC design parameter λ affecting the linear term in the sliding condition was set to zero in all variants. This corresponds to the classical choice of the sliding condition. To allow for a fair comparison, the switching height $\beta = \text{const.} = 2 \times 10^7$ and the boundary layer thickness $\Phi = 200$ were iteratively tuned to achieve good tracking properties while maintaining robustness and only small chattering in the presence of disturbances. These disturbances were implemented as load torques, see Fig. 5.2b. Two alternative load variants were tested: The red signal shows a sinusoidal signal with steps at 0.5 s and 1.5 s; all other figures were created using this load profile. It must be pointed out that also with pulsed load torques, i.e., the blue signal, the results were equally good. It becomes obvious that such strong discontinuities can be managed properly by an integrator disturbance model upon which the KF design is based — even though they represent the worst case for the observer part of the combined control system. Step-like changes are estimated accordingly complying with the KF estimation error dynamics.

5.4.2 Results

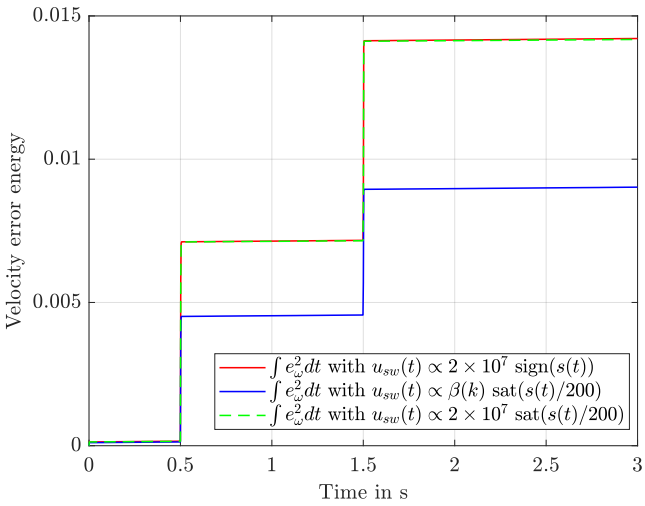
The resulting velocity tracking is demonstrated in Figs. 5.3a-5.3b, where the tracking error and error energy are depicted, and Figs. 5.3c-5.3f, which show the angular velocity itself. While all the variants manage to track the desired velocity profile almost perfectly (Fig. 5.3c), variant 1 shows a little more ripple (Fig. 5.3d) than variants 2 and 3 (Figs. 5.3e-5.3f). Additionally, variant 3 shows smaller deviations in the case of load torque steps (Fig. 5.2c), thanks to a momentarily larger switching height β. This effect results in a significantly smaller error energy, see Fig. 5.3b. The value of β can be seen in Fig. 5.4a, where spikes at 0.5 s and 1.5 s are the cause for the phenomenon mentioned above. The rest of the time, β takes relatively small values but shows a high variability. Thanks to a smaller switching height β, a significant reduction in control input chattering is achieved. This becomes visible when comparing Fig. 5.4f to Fig. 5.4e or especially to 5.4d. However, the major influence stems from the equivalent control part, Eq. (5.10), and the disturbance compensation, Eq. (5.12), which are shown in Fig. 5.4c. The disturbance estimates used for compensation are provided by the KF, using only measurements of current and velocity. These estimates are shown in Figs. 5.5a-5.5b. The currents resulting from those input voltages are displayed in Figs. 5.5c-5.5d and indicate only a negligible difference between the three variants. Finally, the sliding surface $s(t)$ is depicted in Figs. 5.5e-5.5f, where the influence of the external load torque becomes obvious. The SMC variants employing a boundary layer clearly achieve superior behaviour in comparison with the classical switching variant, which has a strong chattering impact in the sliding surface.

5.4.3 Discussion

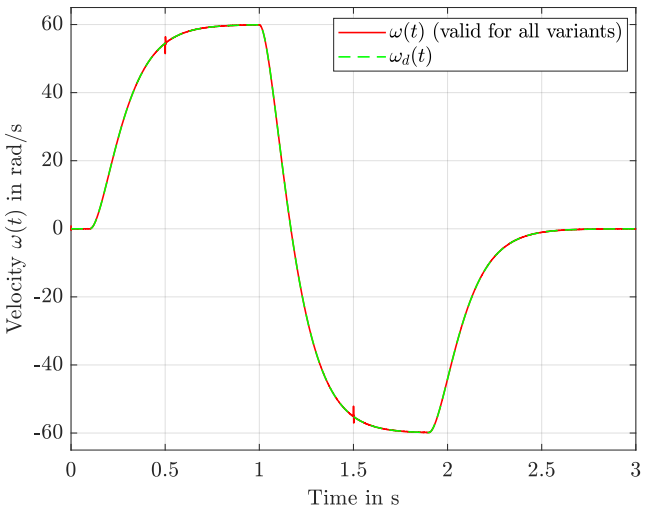
Note that in contrast to the authors' previous work [106], upon which this contribution is based, the sliding function $s(t)$ remains in the close vicinity of zero. This positive effect can be attributed to the employed disturbance compensation provided by the KF. Some slight but negligible influence of the load torque in $s(t)$ becomes visible, which is the price of a significantly reduced chattering. In [106], due to the violation of the SMC sliding condition mentioned there, $s(t)$ was partially non-zero in the presence of large disturbances, with no tendency to converge to zero – despite the fact that a nearly perfect velocity tracking was achieved. This phenomenon can be explained by a model mismatch concerning the dynamics of the sliding surface in closed loop, due to a missing disturbance compensation, which can yield a mismatched MPC. This problem has been solved in the given contribution by means of the disturbance compensation. To further point out the benefits of the proposed approach – the combination of a disturbance estimation by a KF, finite horizon MPC and SMC gain adaptation – it is benchmarked in closed-loop simulation studies against two other widespread methods: A disturbance observer (DOB) according to [108] and a time-delay estimation (TDE), see [93]. Both are implemented to provide estimates for $d(t)$ and $\dot{d}(t)$, using the same (noisy) measurements that are available to the KF as well: current $i(t)$ and velocity $\omega(t)$. The resulting estimates can be compared in Fig. 5.6. Both of these alternatives lead to similar but slightly increased error energies (the unit is omitted here) in comparison with the KF approach: 0.009024 (KF), 0.009076 (TDE) and 0.009383 (DOB). Despite a negligible difference in tracking performance, the use of either DOB or TDE in closed-loop control shows significantly larger chattering amplitudes as compared to the KF variant: u_{sw} is in the range of $\pm 40\,\mathrm{mV}$ for DOB and TDE, where a range of $\pm 20\,\mathrm{mV}$ holds for the KF (apart from spikes of about $\pm 1\,\mathrm{V}$ at $0.5\,\mathrm{s}$ and $1.5\,\mathrm{s}$, respectively). Since TDE is based on the feedback of the acceleration $\dot{\omega}$, which needs to be determined via numerical differentiation of a noisy measurement signal, the performance degradation as compared to the KF becomes evident. In fact, to achieve reasonably useful results, the necessary numerical derivative of the velocity $\omega(t)$ had to be low-pass filtered using a cut-off frequency of $5000\,\mathrm{rad\,s^{-1}}$. Lower frequencies resulted in considerably worse tracking, while higher corner frequencies led to even more noise than depicted in Fig. 5.6e. The KF offers a superior behaviour in the presence of noise because it explicitly considers noise processes. Furthermore, since the KF constitutes a disturbance estimator with an integrated de-noising state observer, its state estimates for current and velocity are used to supply the SMC with state feedback, anyway. This renders the KF a perfect solution in cases like these, where state and disturbance estimates are needed. A formal proof, however, regarding the compliance with the SMC sliding condition as stated in Eq. (5.9) using suitable choices of the MPC weighting matrices $\mathbf{Q}$ and $\mathbf{R}$, still remains an open problem. This contribution presents an adaptive tuning of the switching gain of an SMC that is achieved by means of an MPC scheme designed on the basis of the sliding surface error dynamics. To properly address disturbances and the impact of external load torque in the control approach, they are compensated for using estimates provided by a KF. The overall design is benchmarked in simulations considering a frequently used application of high practical relevance — a DC drive. The simulations clearly show that the MPC-based adaptation of the switching height represents an effective means of counteracting a drawback of classical SMC — chattering caused by a conservative choice of the switching height, which is unnecessarily high most of the time and should be reduced when permissible.



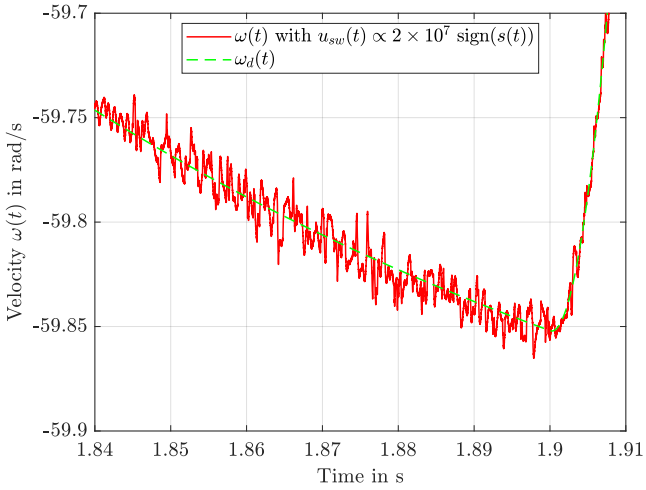
(a) Simulated values for the tracking errors $e(t)$ for all the variants.



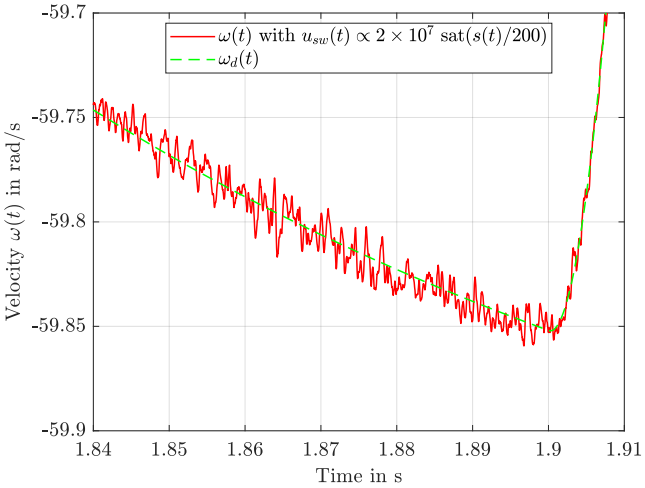
(b) Integrated squared error for all the variants



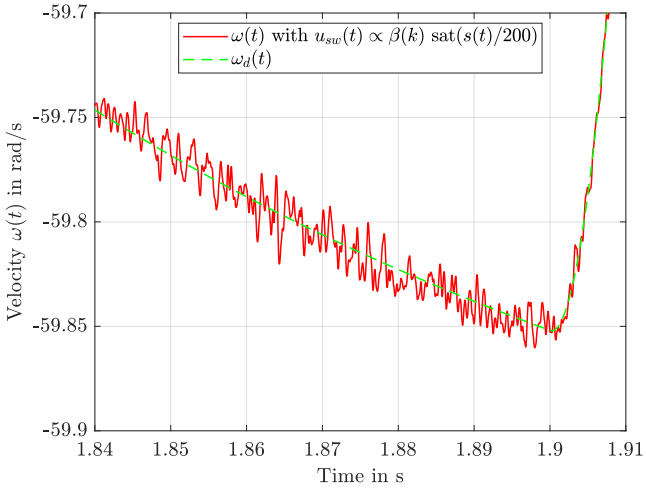
(c) Comparison of desired and simulated angular velocities for all variants.



(d) Comparison of desired and simulated angular velocities, detailed view for constant switching amplitudes.

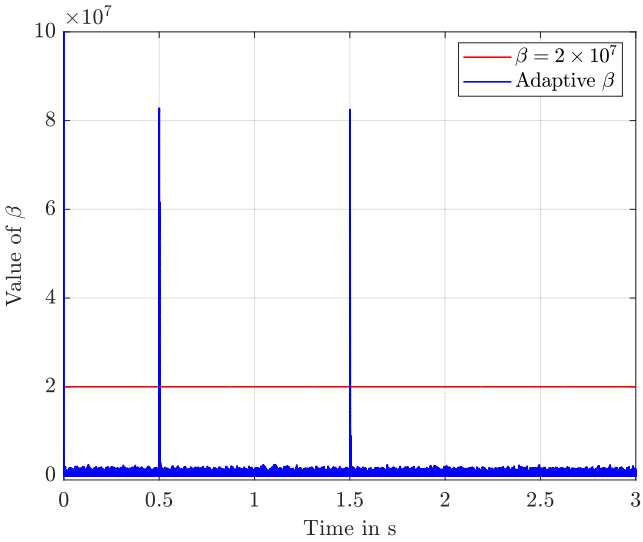


(e) Comparison of desired and simulated angular velocities, detailed view for the classical boundary layer approach.

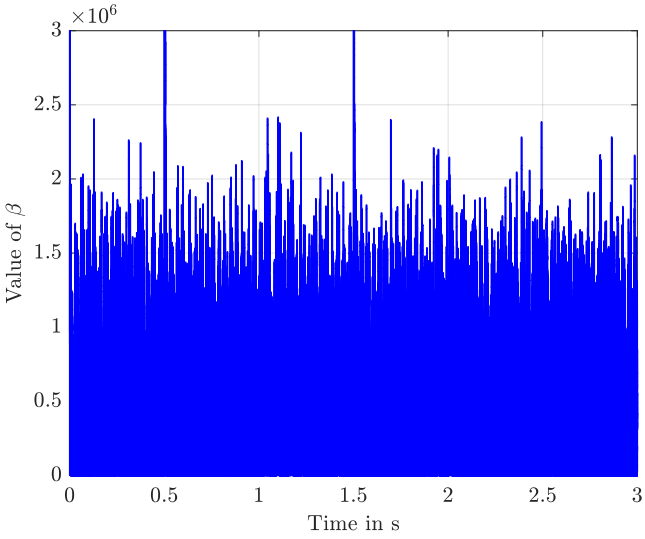


(f) Comparison of desired and simulated angular velocities, detailed view for the proposed gain adaptation scheme.

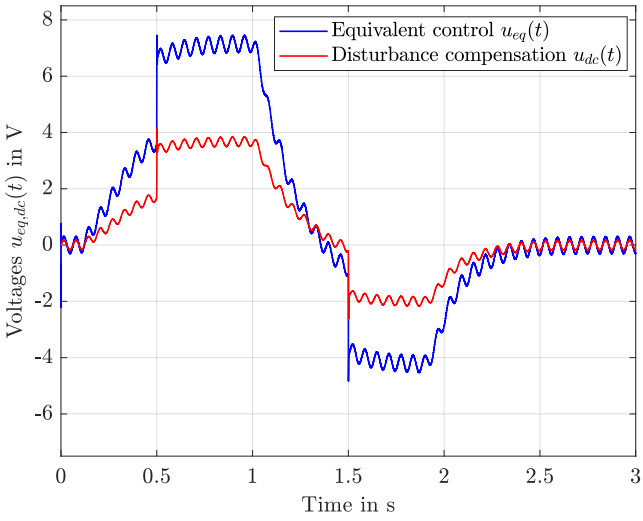
Figure 5.3: Simulation results showing the velocity tracking quality.



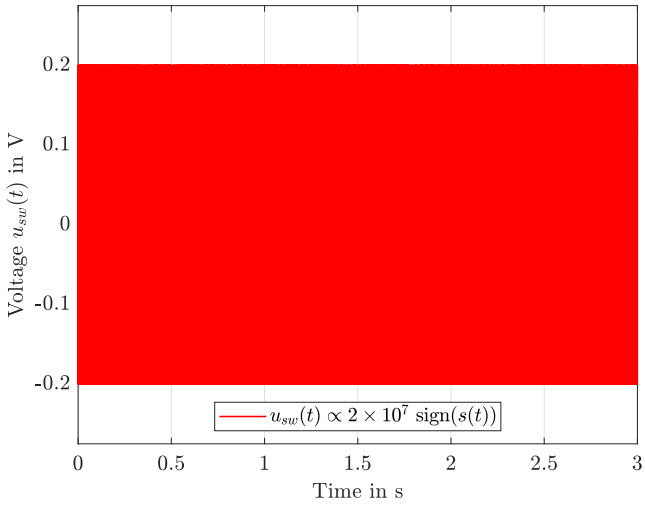
(a) Simulated values for the switching height β.



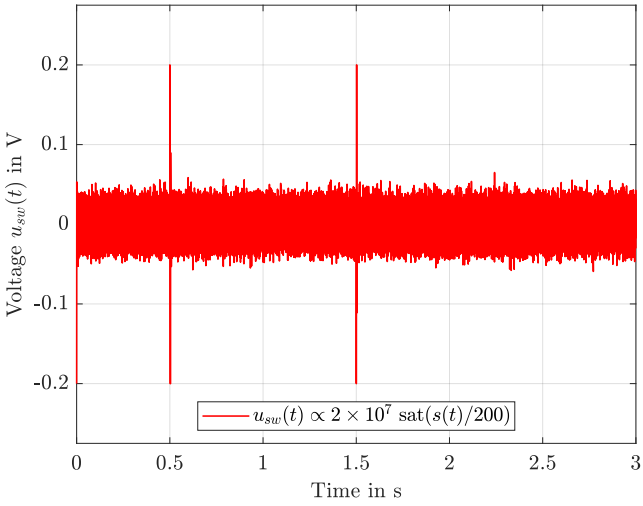
(b) Simulated values for the switching height β (detailed view).



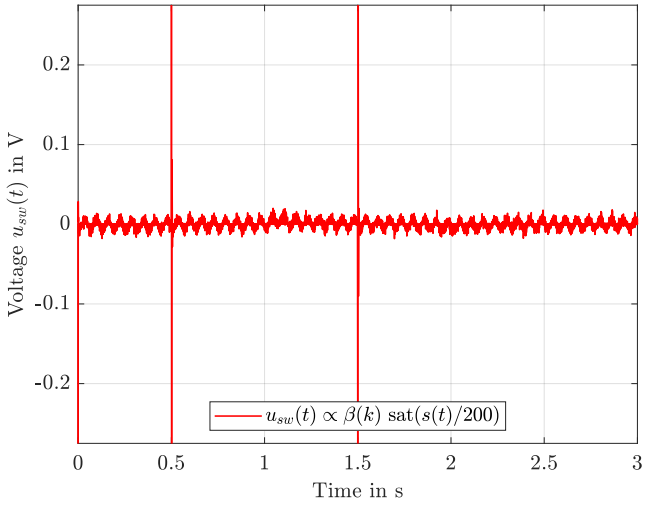
(c) Contribution of the equivalent control law and the disturbance compensation to the input $u(t)$.



(d) Switching component $u_{sw}(t)$ of the input for constant switching amplitudes.

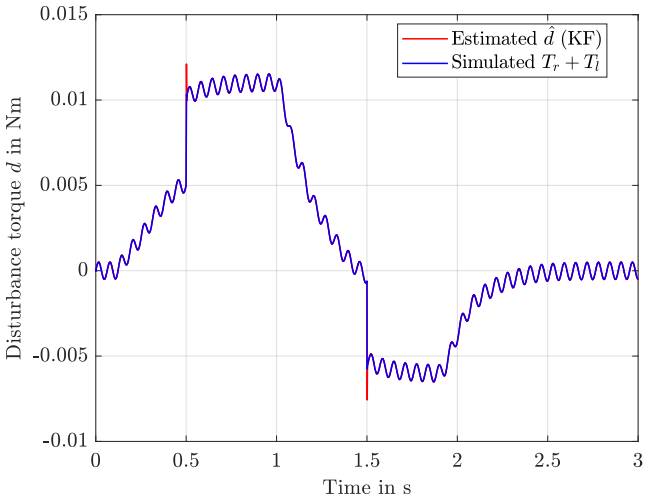


(e) Switching component $u_{sw}(t)$ of the input for the classical boundary layer approach.

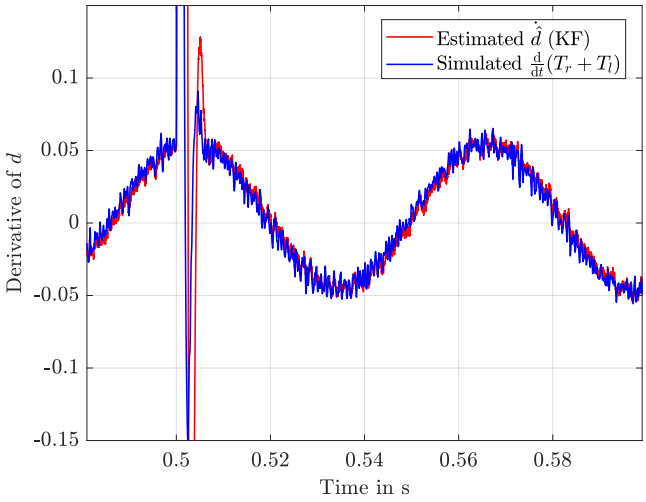


(f) Switching component $u_{sw}(t)$ of the input for adaptive switching amplitudes.

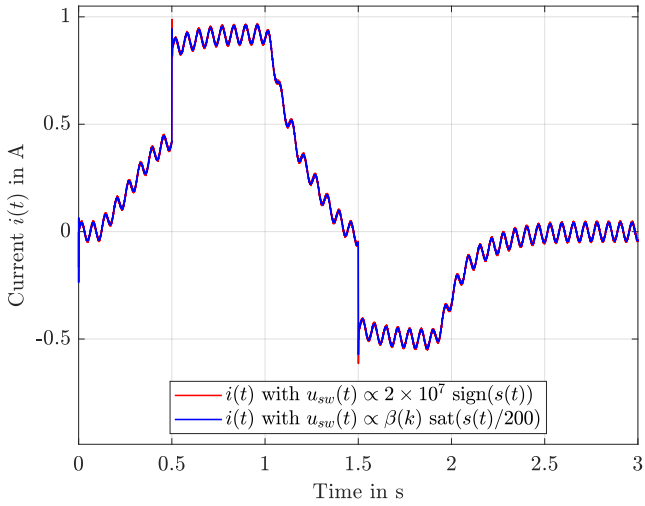
Figure 5.4: Simulation results involving the switching amplitude β and the various control laws for the input $u(t)$.



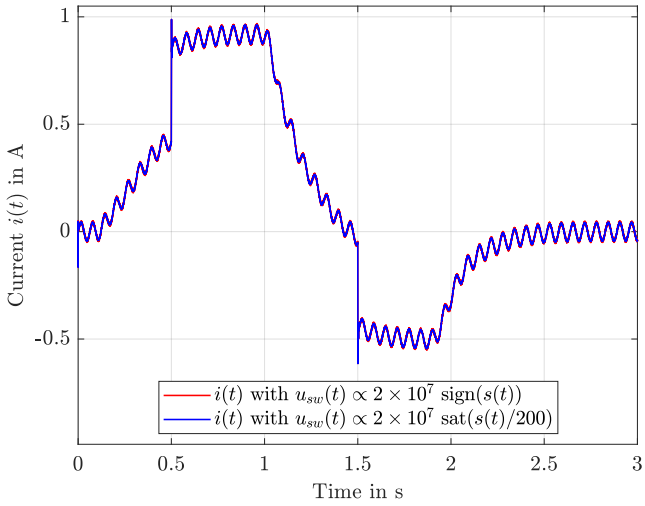
(a) $d(t)$ and its estimates provided by the KF.



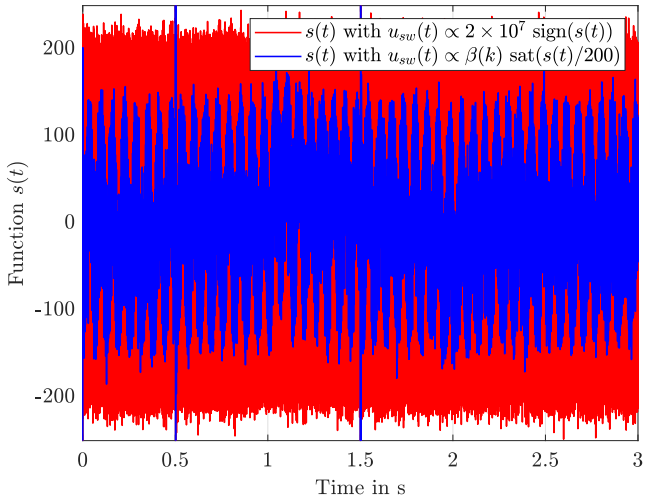
(b) $\dot{d}(t)$ and its estimates provided by the KF.



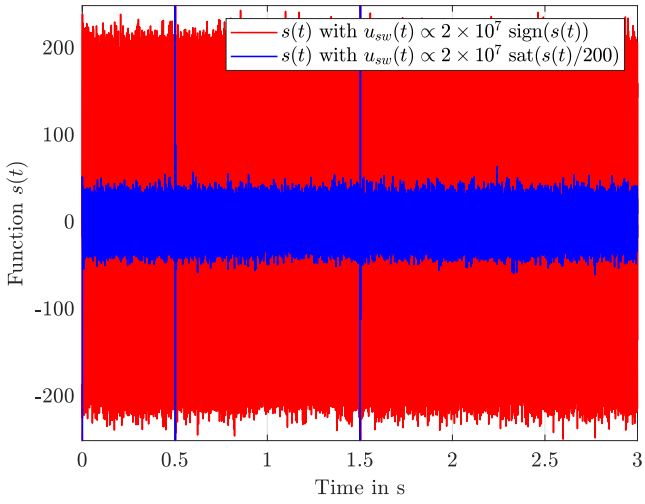
(c) Comparison of simulated currents i for constant switching amplitudes and the proposed adaptation scheme.



(d) Comparison of simulated currents i for constant switching amplitudes and the classical boundary layer approach.

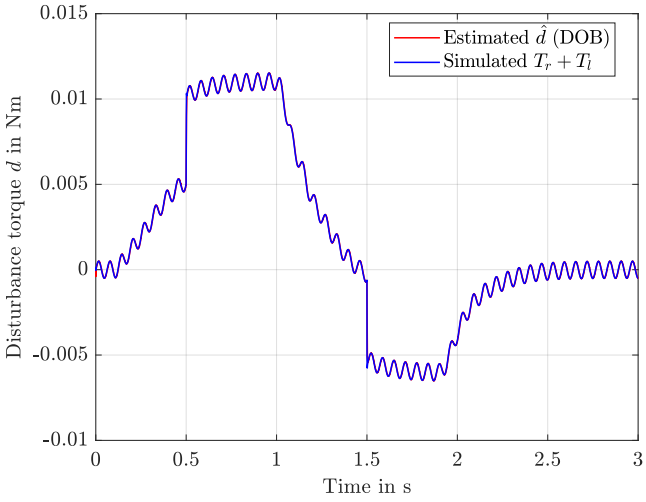


(e) Simulated values for the sliding surface $s(t)$ for constant switching amplitudes and the proposed adaptation scheme.

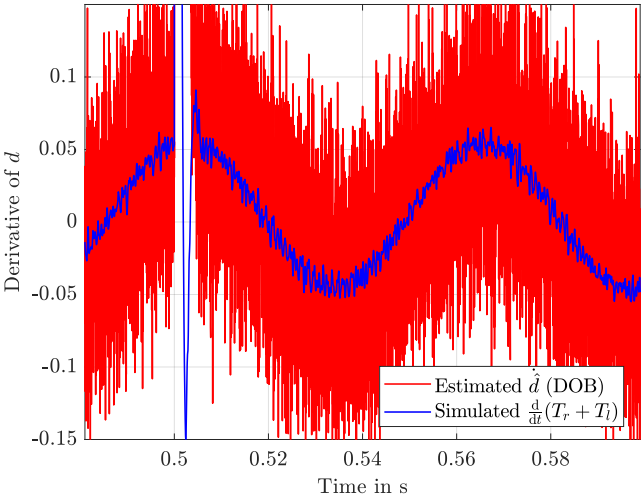


(f) Simulated values for the sliding surface $s(t)$ for constant switching amplitudes and the classical boundary layer approach.

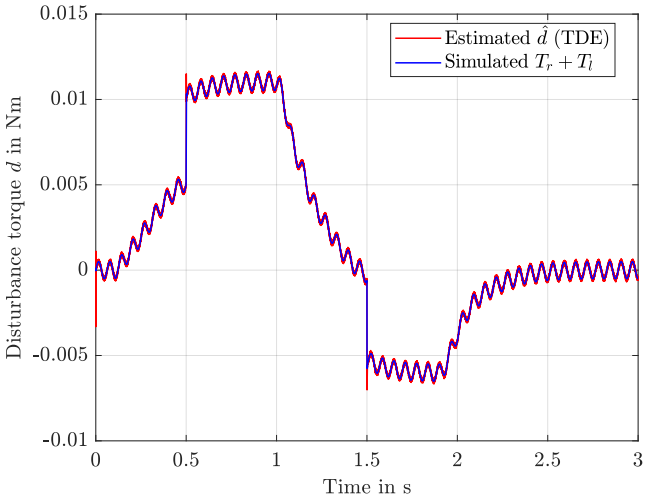
Figure 5.5: Simulation results for the disturbance and its derivative, current $i(t)$ and sliding surface $s(t)$.



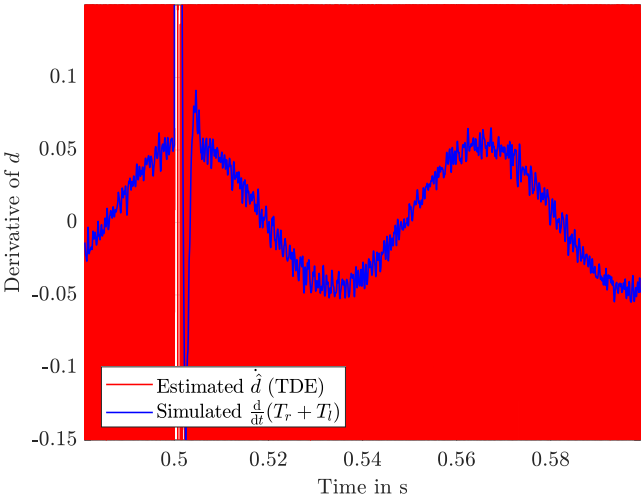
(a) Disturbance estimates for d using a disturbance observer (DOB).



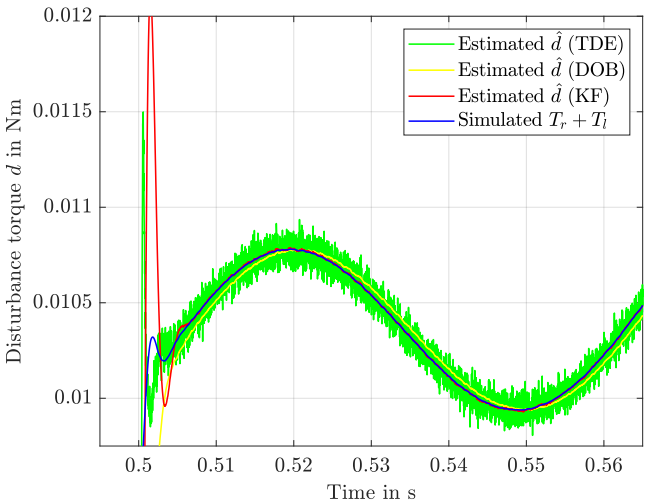
(b) Disturbance estimates for $\dot{d}$ using a disturbance observer (DOB).



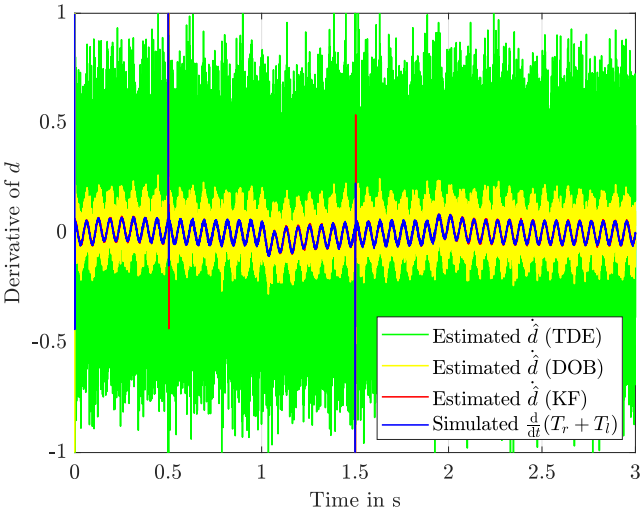
(c) Disturbance estimates for d using time-delay estimation (TDE).



(d) Disturbance estimates for $\dot{d}$ using time-delay estimation (TDE).



(e) Disturbance estimates for d using KF, DOB and TDE in comparison.



(f) Disturbance estimates for $\dot{d}$ using KF, DOB and TDE in comparison.

Figure 5.6: Comparison of different disturbance observer schemes.

6 Direct Speed Control of PMSMs using Constrained Optimal SMC and Disturbance Compensation via EKF

This chapter, like the last one, pursues the goal of direct (as opposed to cascaded) speed control of electric motors. Here, however, three-phase synchronous motors with permanent magnets in the rotor (PMSM) are considered instead of DC motors. PMSMs do not require a commutator or brushed sliding contacts, making them mechanically durable and low-maintenance; however, the three-phase rotating field must be generated externally. For this purpose, knowledge of the rotor angle is required at all times, otherwise it is possible that the motor rotates in the wrong direction, does not move at all or shows other error patterns. If the angle is not measured with a sensor (resolver or absolute encoder, using a coded encoder disk or the like), it must be estimated, which was the foundation of the field of "sensorless control". The estimation of this angle, based on the measured stator currents/voltages, is the main task of the EKF that was designed for this purpose (and, apart from the paper upon which this chapter is based, has not been described in the literature so far). Furthermore, the EKF, analogous to chapter 5, also estimates the load torque (or a lumped, unmatched disturbance) and its derivative, for the purpose of disturbance compensation by the controller. The most important innovation of this project, however, is the derivation of the *equivalent control* laws of the SMC, which is based on analytic, classical minimization of the system inputs $u_{d,q}$ subject to equality constraints (using Lagrange multipliers). The developed SMC, which also has not been described in the literature yet, works with two surfaces instead of one, where one is dedicated to the velocity error and one to the deviation of the so-called maximum torque per Ampere (MTPA) trajectory. The MTPA approach allocates the desired currents $i_{d,q}(t)$ (in the rotor fixed, i.e. rotating coordinate system with the abstracted axes d and q) depending on a given desired motor torque. If the motor has a rotor with permanent magnets glued to the inner surface of the rotor (so-called *surface PMSM*, SPMSM), the inductances of the two axes are approximately equal and only the Lorentz force (and, thus, only i_q) contributes to the torque generation. However, if the magnets are buried in the rotor material (*interior PMSM*, IPMSM), reluctance force effects occur. These can be exploited in addition to the Lorentz force, which is achieved with negative i_d currents (field weakening operation). In the latter case, there is a certain redundancy/ambiguity or degree of freedom in deciding which of the two system inputs $u_{d,q}$ is used, at least if speed control is the only objective and no other criteria are considered. This ambiguity can be resolved by means of optimization, including a second constraint (apart from velocity tracking), namely the MTPA condition. This minimization turns out to have a unique pair of solutions for u_d and u_q which is shown to be the optimum.

Classically, PMSM (and other AC motors) are controlled in cascaded architectures, i.e. with one external control loop for position and/or velocity control (manipulated variable: torque) and one or two internal control loops for the current $i_{d,q}$ (manipulated variables: voltages $u_{d,q}$). However, this approach only works well as long as the inner, electrical subsystem reacts much faster than the outer, mechanical subsystem. This is no longer the case for many modern PMSMs (especially small, high-powered ones), which is why direct speed control is becoming increasingly important. Approaches to achieve direct speed control via SMC have been present in the literature for several years. However, they still require a reference for the i_d current, so they are mostly only suitable for SPMSMs (where i_d can be forced to zero). The presented SMC solves the problem of direct speed control regardless of the PMSM type, because, thanks to the second sliding surface, no current references have to be generated.

6.1 Literature survey

The power density of PMSMs is currently still increasing, so the choice of control strategy becomes more and more important so as to ensure further efficiency gains when operating these electrical machines. During the last few years, a great deal of effort has been invested in the context of optimal control strategies for PMSMs, and also in the field of SMC [95, 109]. Such techniques are often combined with sensor-reduced control strategies, as recently in [72, 98]. However, few contributions consider how to integrate optimization criteria with SMC strategies, as in [4].

6.1.1 Optimality in the context of PMSM control strategies

In order to obtain maximum motoric torque using minimum d, q stator currents, the MTPA control strategy can be employed, the foundations of which are laid down in [110]. A more recent publication is [111], in which a nonlinear MPC scheme is adopted to minimize the MTPA condition within a direct torque control. This approach offers only suboptimal solutions at each sample time, because of the numerical structure of the control law, and does not guarantee the stability of the optimizing closed-loop control strategy, which is a common simplification in the literature (see, for example, [112]). Another related paper by the same authors of [111], which combines the above-mentioned MTPA criterion in the MPC cost function with one that considers voltage limits, can be found in [113]. Ultimately, the controller switches between multiple cost functions during the online optimization, depending on the current operating point. However, the limit of this approach in [111] and [113] is the separation of the optimization procedure, which implies switching criteria. This renders the control system stability very difficult to investigate, since different cost functions are used in the control law and must be selected during the operation of the machine. Moreover, when using a separation procedure, global optimality is not guaranteed. The contribution [114] is based on the same concept as that seen in [111] and [113], and shows the same disadvantages, but instead deals with direct speed control.

In its most straightforward manifestation, MTPA is used to generate current references that can be used within current control loops. As an alternative to such cascaded control architectures, direct speed control can be used for PMSMs, dissolving the loop separation and directly determining the input voltage according to the speed reference. This work proposes such a direct speed control scheme, adopting a monolithic (not cascaded) control strategy. This choice can be justified as cascaded control architectures can cause problems if the dynamics of the electrical and the mechanical subsystems are approximately equally fast. The inner loop dynamics cannot be neglected from the point of view of the outer loop controller in the case of fast PMSMs that are optimized for very high power densities; this is the main field of application for these motors. This limits the possible dynamics of cascaded controllers, particularly in high-power applications. Cascaded structures have a limited bandwidth, in order to avoid large overshoots and possible ringing, due to an artificial separation principle in their design. This chapter considers a PMSM with comparably fast electrical and mechanical dynamics, proposing a monolithic speed control scheme, as in Fig. 6.1. A short overview of the advantages and drawbacks of cascaded control versus direct speed control is given in [111].

The paper [115] describes a speed control scheme based on input-output linearization, a load torque observer and a boundary-layer integral SMC. Here, the d-axis current and the rotor's mechanical angular velocity are chosen as controlled outputs, the system is linearized with respect to these, and two integral sliding surfaces are designed that consider the error of the respective output. However, the control system cannot be considered as a *direct* speed control, because it still requires a reference generator for the currents (and not only for the velocity) and is, therefore, a cascaded control scheme. This approach does not allow optimization of the inputs u_d and u_q since they are already fixed by separation, with which the control is conceived.

6.1.2 Kalman filters as state and disturbance observers

Sensorless control strategies (i.e. control strategies that use a reduced or minimal number of sensors for feedback control) are very attractive because of the cost and complexity reductions in relation to the control system and motor. (*Sensorless* control has been established as an umbrella term in the literature, even when the specific approach is not truly "sensorless" but rather "sensor-reduced".) Said advantages are not only present in the case of electromagnetic drives, but also in that of any controlled system in which the reduction of measured signals represents a relevant specification [3]. Recent literature on the control of electromagnetic drives considers sensorless control as a primary factor, not only for reasons related to costs reduction and reliability, but also those regarding the quality of control and signals and electromagnetic compatibility, and yields a robust and more precise tracking [116]. SMC is a state-based control strategy that often requires the availability of the full state vector, but more importantly, it requires state feedback with very low noise power. In this sense, sensorless control, with the aim of reducing the number of measured states, and denoising the signals, is a challenging problem whose resolution can improve the whole controlled system. A reduction in the number of sensors used is achieved by using an observer as a virtual sensor. Observers combine a-priori knowledge about the system (in form of a mathematical model used for prediction) with actual feedback of the system's states and/or outputs. It is known that an accurate knowledge of the model and its parameters is necessary in order to realize an effective sensorless control. In particular, it must be understood that observers suffer from a lack of precision in the presence of uncertainties in the model. In [79, 80] the problem of PMSM model parameter estimation is solved using different types of Kalman filters.

Various contributions have recently been made to this field; for example, [117] and [118] compare the performance of EKFs and sliding-mode observers in the presence of measurement noise. However, the estimation of the mechanical rotor angle [119] as well as possible disturbances [120] are still challenging problems. Disturbance estimation and compensation for PMSMs using an EKF was recently demonstrated in [121], where it aids an SMC with velocity-tracking control. Here, however, the observer is based on the system dynamics in rotor-fixed d, q coordinates, therefore requiring the direct measurement of the rotor angle. More formally, through a sensitivity analysis, the contributions [122] and [123] have recently investigated the influence of model uncertainty on the observed position and robustness of a proportional-integral observer in current control used for PMSM drives.

6.1.3 Contribution of this work

In this chapter, an EKF is used for sensorless control, providing estimates for all states. The approach is based on that reported in [124] (that is, it considers the phase-current dynamics within a two-component stator-fixed α, β reference frame). The approach taken in [124] was amended by considering the mechanical dynamics of the system, introducing the *mechanical* rotor velocity as an additional state, while for the angle the *electrical* rotor angle is considered. The approach in [124] (which considers the use of a model with an infinitely large moment of inertia in the mechanical subsystem) is typical in the literature, see e.g. [125], [126]. In comparison, the proposed approach obtains a more precise result through the use of a mechanical model of the system, instead of adopting the conservative hypothesis that the derivative of the electrical angular velocity is equal to zero. This combined EKF allows the electrical rotor angle (which is always required in the control of PMSMs) to be precisely estimated, as well as the mechanical disturbance torque and its derivative. Since an estimate of the angular acceleration is also required, which is in fact a very common problem in applied control, another Kalman filter is introduced that combines the three most powerful techniques of approximating the acceleration: numerical differentiation of the velocity, model-based approximation, and state-observer based approximation.

The first and main innovation by this contribution is that it proposes a direct speed control law that is usable for both IPMSMs and SPMSMs. Additionally, it offers analytical stabilizing solutions for the defined convex optimization problem in the context of SMC. The control law minimizes the input voltages and achieves velocity tracking while optimizing the currents at the same time. The analytical solution to the defined problem allows a globally stabilizing optimal control law that does

not require computationally expensive online optimization to be implemented. More technically, in the context of the derivation of the equivalent control laws, a constrained convex optimization problem is defined and solved, eliminating redundancy in the choice of control inputs by introducing a constrained cost function, consisting of the two input voltages, that can be minimized through the use of the classic Lagrange approach. The two optimization constrains are characterized by $\dot{s} = 0$ (for two sliding surfaces, as will be shown in the sequel), which is the well-known sliding mode differential equation. The first sliding surface defines the desired speed error dynamics and thus guarantees asymptotic tracking of the desired velocity dynamics. At the same time, the second constraint, which states the "MTPA criterion dynamics", achieves current optimality in terms of MTPA. Since the unique analytical solution is found offline, no expensive online optimization is required.

The chapter is organized in the following way. Section 6.2 describes the physical system and shows the mathematical model, as well as providing an observability analysis. In Section 6.3 the EKF algorithm is detailed, followed by a description of a secondary Kalman filter, dedicated to the estimation of the angular acceleration. In Section 6.5, an optimal SMC with analytical solution is proposed and formalized as a theorem. Section 6.6 discusses the simulation results.

6.2 Description of the physical system

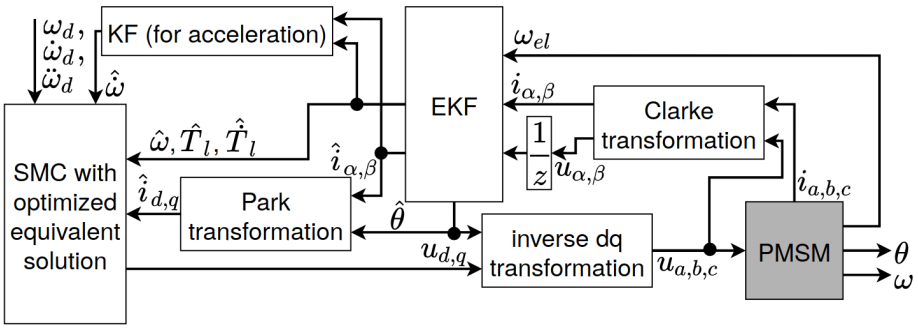


Figure 6.1: Schematics of the combined estimation and control architecture.

To aid advanced controller design in PMSM, it is very important that an appropriate model of the motor be obtained. A good model should not only be an accurate representation of system dynamics, it should also facilitate the application of the existing control techniques. Among a variety of models presented in the literature since the introduction of PMSM, the two-axis d-q model, obtained using the d-q transformation (which is a combination of Clarke's and Park's transformations) is the most widely used in variable-speed PMSM control applications, [127] and [128]. The d-q transformation is a coordinate transformation that converts the three-phase stationary variables into variables in a rotating coordinate system. In d-q transformation, the rotating coordinate is defined relative to a stationary reference angle. It is worth noting that the d-q transformation allows the complexity of the model to be reduced. This is because the three-phase system can be described using just two coordinates, whose dynamics no longer show trigonometric expressions as they are hidden by the transformation. Relation (6.1) from [129] gives the variant of the d-q transformation used in this contribution, in which $\theta(t)$ is the electrical rotor angle, i.e. the mechanical rotor angle multiplied by the number of pole pairs p, and where $\mathbf{M}^{-1}$ represents the corresponding inverse transformation. Note that, while there are several

different formulations of this transformation described in the literature, here we use the following:

$$\begin{aligned}
\begin{bmatrix} x_d(t) & x_q(t) & x_0(t) \end{bmatrix}^T &= \mathbf{M} \begin{bmatrix} x_a(t) & x_b(t) & x_c(t) \end{bmatrix}^T, \\
\mathbf{M} &= \begin{bmatrix} \frac{2\cos(\theta(t))}{3} & \frac{2\cos(\theta(t)-2\pi/3)}{3} & \frac{2\cos(\theta(t)+2\pi/3)}{3} \\ \frac{-2\sin(\theta(t))}{3} & \frac{-2\sin(\theta(t)-2\pi/3)}{3} & \frac{-2\sin(\theta(t)+2\pi/3)}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix}, \\
\mathbf{M}^{-1} &= \begin{bmatrix} \cos(\theta(t)) & -\sin(\theta(t)) & 1 \\ \cos(\theta(t)-2\pi/3) & -\sin(\theta(t)-2\pi/3) & 1 \\ \cos(\theta(t)+2\pi/3) & -\sin(\theta(t)+2\pi/3) & 1 \end{bmatrix}.
\end{aligned} \tag{6.1}$$

6.2.1 PMSM dynamics

The flux linkages in a PMSM and their derivatives (assuming constant inductances) are [129]

$$\psi_d(t) = L_d i_d(t) + K_f, \qquad \frac{\mathrm{d}\psi_d(t)}{\mathrm{d}t} = L_d \frac{\mathrm{d}i_d(t)}{\mathrm{d}t}, \tag{6.2}$$

$$\psi_q(t) = L_q i_q(t), \qquad \frac{\mathrm{d}\psi_q(t)}{\mathrm{d}t} = L_q \frac{\mathrm{d}i_q(t)}{\mathrm{d}t}. \tag{6.3}$$

From this, the following electrical model can be derived:

$$\frac{\mathrm{d}i_d(t)}{\mathrm{d}t} = -\frac{R}{L_d} i_d(t) + p\omega(t)\frac{L_q}{L_d} i_q(t) + \frac{u_d(t)}{L_d} \tag{6.4}$$

$$\frac{\mathrm{d}i_q(t)}{\mathrm{d}t} = -\frac{R}{L_q} i_q(t) - p\omega(t)\frac{L_d}{L_q} i_d(t) + \frac{u_q(t)}{L_q} - \frac{K_f p}{L_q}\omega(t), \tag{6.5}$$

where it is worth noting that the parameters R, L_d, L_q, K_f and p are the stator resistance, d-axis and q-axis inductances, the permanent magnet flux linkage in the air gap, and the number of magnetic pole pairs, respectively.

Considering the electrical rotor angle θ and the mechanical angular velocity of the rotor ω, the mechanical model is

$$\frac{\mathrm{d}\theta(t)}{\mathrm{d}t} = p\omega(t), \tag{6.6}$$

$$\frac{\mathrm{d}\omega(t)}{\mathrm{d}t} = \frac{T_m(t)}{J} - \frac{F}{J}\omega(t) - \frac{T_l(t)}{J}, \tag{6.7}$$

where J represents the inertia of the motor, F is a velocity-proportional friction coefficient, $T_l(t)$ represents a possible external load torque and/or unknown lumped disturbances, and $\theta(t) \in (-\pi, \pi]$ is cyclic. From the electrical model, through the superposition of the three-phase fields, the following equation for the torque can be derived:

$$T_m(t) = \frac{3}{2} p\Big(i_q(t)\psi_d(t) - i_d(t)\psi_q(t)\Big), \tag{6.8}$$

where the term $i_q(t)\psi_d(t) - i_d(t)\psi_q(t)$ defines the cross coupling within the PMSM dynamics. This leads to the effect whereby a variation of the current $i_d(t)$ has an impact on the current $i_q(t)$ in the q-axis and vice versa. By introducing the inductances, the torque can be expressed alternatively in the form

$$T_m = \frac{3}{2} p\Big((L_d - L_q) i_d(t) i_q(t) + K_f i_q(t)\Big). \tag{6.9}$$

Remark 6.1 *Please note that power electronics components were not modelled in this contribution. It is assumed that the actual three-phase input voltages $u_{a,b,c}$ can be controlled.*

Using the following substitutions (from the Clarke and Park transformations)

$$i_d(t) = i_\alpha(t)\cos(\theta(t)) + i_\beta(t)\sin(\theta(t)), \tag{6.10}$$
$$i_q(t) = -i_\alpha(t)\sin(\theta(t)) + i_\beta(t)\cos(\theta(t)), \tag{6.11}$$
$$\frac{\mathrm{d}i_d(t)}{\mathrm{d}t} = \frac{\mathrm{d}i_\alpha(t)}{\mathrm{d}t}\cos(\theta(t)) - i_\alpha(t)\omega(t)\sin(\theta(t)) + \frac{\mathrm{d}i_\beta(t)}{\mathrm{d}t}\sin(\theta(t)) + i_\beta(t)\omega(t)\cos(\theta(t)), \tag{6.12}$$
$$\frac{\mathrm{d}i_q(t)}{\mathrm{d}t} = -\frac{\mathrm{d}i_\alpha(t)}{\mathrm{d}t}\sin(\theta(t)) - i_\alpha(t)\omega(t)\cos(\theta(t)) + \frac{\mathrm{d}i_\beta(t)}{\mathrm{d}t}\cos(\theta(t)) - i_\beta(t)\omega(t)\sin(\theta(t)), \tag{6.13}$$

together with some simplifying manipulations, the PMSM dynamics can be expressed using stator-fixed α, β-coordinates:

$$\frac{\mathrm{d}i_\alpha}{\mathrm{d}t} = \frac{1}{2L_dL_q}\Big[(u_\alpha - i_\alpha R)(L_d+L_q) + i_\beta p\omega(L_d-L_q)^2 - (L_d-L_q)(u_\alpha - i_\alpha R + i_\beta p\omega(L_d+L_q))\cos(2\theta)$$
$$-(L_d-L_q)(u_\beta - i_\beta R - i_\alpha p\omega(L_d+L_q))\sin(2\theta) + 2K_f L_d p\omega\sin(\theta)\Big], \tag{6.14}$$
$$\frac{\mathrm{d}i_\beta}{\mathrm{d}t} = \frac{1}{2L_dL_q}\Big[(u_\beta - i_\beta R)(L_d+L_q) - i_\alpha p\omega(L_d-L_q)^2 + (L_d-L_q)(u_\beta - i_\beta R - i_\alpha p\omega(L_d+L_q))\cos(2\theta)$$
$$-(L_d-L_q)(u_\alpha - i_\alpha R + i_\beta p\omega(L_d+L_q))\sin(2\theta) - 2K_f L_d p\omega\cos(\theta)\Big], \tag{6.15}$$
$$\frac{\mathrm{d}\omega}{\mathrm{d}t} = -\frac{F\omega}{J} - \frac{T_l}{J} + \frac{3p}{2J}(i_\beta\cos(\theta) - i_\alpha\sin(\theta))(K_f + (L_d-L_q)(i_\beta\sin(\theta) + i_\alpha\cos(\theta))), \tag{6.16}$$
$$\frac{\mathrm{d}\theta}{\mathrm{d}t} = p\omega. \tag{6.17}$$

Remark 6.2 *The given PMSM dynamics in d,q- and α,β-coordinates are equivalent. In simulation studies, both sets of differential equations can be used to model the PMSM, producing identical results in terms of the actual three-phase quantities. The d,q-variant is easier to implement, due to its more compact equations, and has the additional advantage of yielding constant currents in steady-state (in the given reference frame, that is); it thereby facilitates easier debugging and/or other analysis methods like fault detection. In this contribution, however, the α,β-variant was implemented for the simulation of the actual PMSM system, because the observer was designed to use this coordinate system and requires model implementation in any case. The control design, on the other hand,* is *based on d,q-coordinates, complying with the traditional approach in PMSM control.*

6.2.2 Observability analysis

Regarding PMSM observability in cases where the mechanical rotor angle is *not* measured, and only the currents in the stator-fixed α,β reference frame are known, the property of local weak observability [9] can be investigated, yielding a sufficient (but not necessary) condition for observability at a point $\mathbf{x}_0$. For a nonlinear system of order n, with m outputs, inputs $\mathbf{u}$, states $\mathbf{x}$, dynamics $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x},\mathbf{u})$, and outputs field $\mathbf{y}(t) = \mathbf{h}(\mathbf{x})$, the condition is

$$\mathrm{rank}\mathbf{O}(\mathbf{x},\mathbf{u}) = n \quad \text{at } \mathbf{x} = \mathbf{x}_0, \tag{6.18}$$

with the nonlinear observability matrix $\mathbf{O} \in \mathbb{R}^{nm\times n}$, where $\mathcal{L}_\mathbf{f}^k\mathbf{h}$ is the kth Lie derivative of output field $\mathbf{h}$ with respect to the dynamics field $\mathbf{f}$

$$\mathbf{O} = \begin{bmatrix}\mathbf{O}_1\\ \mathbf{O}_2\\ \mathbf{O}_3\\ \dots\\ \mathbf{O}_n\end{bmatrix} = \frac{\partial}{\partial\mathbf{x}^T}\begin{bmatrix}\mathbf{h}\\ \mathcal{L}_\mathbf{f}\mathbf{h}\\ \mathcal{L}_\mathbf{f}^2\mathbf{h}\\ \dots\\ \mathcal{L}_\mathbf{f}^{n-1}\mathbf{h}\end{bmatrix} = \begin{bmatrix}\frac{\partial\mathbf{h}}{\partial\mathbf{x}^T}\\ \frac{\partial\mathbf{O}_1\mathbf{f}}{\partial\mathbf{x}^T}\\ \frac{\partial\mathbf{O}_2\mathbf{f}}{\partial\mathbf{x}^T}\\ \dots\\ \frac{\partial\mathbf{O}_{n-1}\mathbf{f}}{\partial\mathbf{x}^T}\end{bmatrix}. \tag{6.19}$$

Symbolically calculating this observability matrix is tedious, but it can be done automatically using computer algebra systems. It is sufficient to find one determinant (i.e. a closed-form expression involving states/inputs as variables) that is non-zero for the value ranges of all emergent variables that correspond to realistic working conditions. Some determinants are investigated in-depth and reported in the literature (see [130–132]), showing that the PMSM system in stator-fixed α, β-coordinates is observable (with respect to the states i_α, i_β, ω, θ) using only measurements of the currents $i_{\alpha,\beta}$ (and knowledge of the inputs, of course), as long as the velocity ω is not too small or $L_d \neq L_q$.

Since the approach presented in this chapter requires not only the above-mentioned states, but also the load torque T_l and its derivative, a third measurement is introduced: the electrical angular velocity $\omega_{el} = p\omega$. This allows for the convergence of the state estimates, even if they are used for feedback control and even when starting from standstill, and also for unknown initial rotor angles. Since most modern three-phase PMSMs integrate three digital Hall effect sensors in the stator to facilitate basic field commutation, ω_{el} is easy to calculate from the digital sensor signals using time measurements (or using edge detection and counting, in case of high speeds or large number of poles), which is accurate enough to serve as an additional informaton source for the observer.

As a suboptimal state-observer algorithm that features powerful sensor-fusion capabilities, an EKF that estimates states and disturbances is designed in the following section. For this purpose, the model given in (6.14-6.17) is augmented through the use of a double-integrator disturbance model [4]

$$\frac{\mathrm{d}T_l}{\mathrm{d}t} = \dot{T}_l, \qquad \frac{\mathrm{d}\dot{T}_l}{\mathrm{d}t} = 0. \tag{6.20}$$

6.3 EKF for state and disturbance estimation

A popular EKF variant was introduced in [78, 133] and is still used in more-recent contributions [124]. In this variant, the EKF is used to estimate the currents $i_{\alpha,\beta}$ in the stationary α, β reference frame, as well as the mechanical rotor velocity and the electrical rotor angle. Usually, PMSM control is based on currents $i_{d,q}$ within the rotary d, q reference frame, and the corresponding coordinate transformation is calculated *outside of the EKF* using the EKF-estimated electrical rotor angle.

We propose to extend the EKF reported in [124] through the use of a model of the mechanical subsystem, using measurements of the stator currents $i_{\alpha,\beta}$ and the electrical rotor velocity ω_{el} to obtain denoised estimates of the currents, mechanical rotor velocity ω, and electrical rotor angle θ, as well as the lumped disturbance or load torque T_l and its derivative. A nonlinear state-space model with states $\mathbf{z}$, inputs $\mathbf{u}$ and outputs $\mathbf{y}$ can be derived as

$$\begin{aligned} &\mathbf{z} = \begin{bmatrix} i_\alpha & i_\beta & \omega & \theta & T_l & \dot{T}_l \end{bmatrix}^T, \quad \mathbf{u} = \begin{bmatrix} u_\alpha & u_\beta \end{bmatrix}^T, \\ &\dot{\mathbf{z}} = \mathbf{f}(\mathbf{z}, \mathbf{u}), \quad \mathbf{y} = \mathbf{C}_{KF}\mathbf{z}, \quad \mathbf{C}_{KF} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & p & 0 & 0 & 0 \end{bmatrix}. \end{aligned} \tag{6.21}$$

The dynamics for the six states are modelled as in Eqs. (6.14-6.17). Note that this system includes additional augmented states T_l, $\dot{T}_l$ that represent the load torque (a lumped unmatched disturbance) and its derivative, with the modelled dynamics given in Eq. (6.20). It is known that Runge-Kutta methods are generally better than the explicit Euler method in terms of truncation errors and resulting computation time (due to the ability to use lager step sizes T_s); however, this benefit is balanced by an increased implementation effort. In the presented case, it became necessary to use a third-order Runge-Kutta (RK3) scheme for the EKF, since sampling times smaller than the chosen one ($T_s = 0.01$ ms), as they would be required when using Euler instead, would have slowed down the implemented

control and estimation system too much. Thus, the a-priori state estimates are

$$\hat{\mathbf{z}}^-(k) = \hat{\mathbf{z}}^+(k-1) + T_s\left(\frac{1}{6}\mathbf{f}_A + \frac{4}{6}\mathbf{f}_B + \frac{1}{6}\mathbf{f}_C\right) \tag{6.22}$$

$$\mathbf{f}_A = \mathbf{f}\left(\hat{\mathbf{z}}^+(k-1), \mathbf{u}(k-1)\right) \tag{6.23}$$

$$\mathbf{f}_B = \mathbf{f}\left(\hat{\mathbf{z}}^+(k-1) + \frac{T_s}{2}\mathbf{f}_A, \mathbf{u}(k-1)\right) \tag{6.24}$$

$$\mathbf{f}_C = \mathbf{f}\left(\hat{\mathbf{z}}^+(k-1) - T_s\mathbf{f}_A + 2T_s\mathbf{f}_B, \mathbf{u}(k-1)\right). \tag{6.25}$$

For the covariance prediction, the continuous-time Jacobian of the system is calculated symbolically

$$\mathbf{J}_\mathbf{f}(\mathbf{z},\mathbf{u}) = \frac{\partial \mathbf{f}}{\partial \mathbf{z}^T} = \begin{bmatrix} \frac{\partial \frac{\mathrm{d}i_\alpha(t)}{\mathrm{d}t}}{\partial \mathbf{z}^T} \\ \frac{\partial \frac{\mathrm{d}i_\beta(t)}{\mathrm{d}t}}{\partial \mathbf{z}^T} \\ \frac{\partial \frac{\mathrm{d}\omega(t)}{\mathrm{d}t}}{\partial \mathbf{z}^T} \\ \frac{\partial \frac{\mathrm{d}\theta(t)}{\mathrm{d}t}}{\partial \mathbf{z}^T} \\ \frac{\partial \frac{\mathrm{d}T_l(t)}{\mathrm{d}t}}{\partial \mathbf{z}^T} \\ \frac{\partial \frac{\mathrm{d}\dot{T}_l(t)}{\mathrm{d}t}}{\partial \mathbf{z}^T} \end{bmatrix} = \begin{bmatrix} & & \mathbf{j}_\alpha & & & \\ & & \mathbf{j}_\beta & & & \\ & & \mathbf{j}_\omega & & & \\ 0 & 0 & p & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{6.26}$$

with symbols $\Delta L = L_d - L_q$,

$$\mathbf{j}_\alpha^T = \begin{bmatrix} \frac{\Delta L(p\omega(L_d+L_q)\sin(2\theta)+R\cos(2\theta))-R(L_d+L_q)}{2L_dL_q} \\ \frac{\Delta L(-p\omega(L_d+L_q)\cos(2\theta)+p\omega\Delta L+R\sin(2\theta))}{2L_dL_q} \\ \begin{bmatrix} \frac{j_{\alpha,\omega}}{2L_dL_q} & \frac{j_{\alpha,\theta}}{L_dL_q} & 0 & 0 \end{bmatrix}^T \end{bmatrix}, \tag{6.27}$$

$$j_{\alpha,\omega} = p\Delta L(i_\alpha(L_d+L_q)\sin(2\theta) + i_\beta\Delta L) + i_\beta p\left({L_q}^2 - {L_d}^2\right)\cos(2\theta) + 2K_f pL_d\sin(\theta), \tag{6.28}$$

$$\begin{aligned} j_{\alpha,\theta} = {} & \Delta L\sin(2\theta)\left(-i_\alpha R + i_\beta p\omega(L_d+L_q) + u_\alpha\right) + \Delta L\cos(2\theta)(i_\alpha p\omega(L_d+L_q) + i_\beta R - u_\beta) \\ & + K_f L_d p\Delta L\omega\cos(\theta), \end{aligned} \tag{6.29}$$

$$\mathbf{j}_\beta^T = \begin{bmatrix} \frac{\Delta L(R\sin(2\theta)-p\omega((L_d+L_q)\cos(2\theta)+L_d-L_q))}{2L_dL_q} \\ -\frac{\Delta L(p\omega(L_d+L_q)\sin(2\theta)+R\cos(2\theta))+R(L_d+L_q)}{2L_dL_q} \\ \begin{bmatrix} \frac{j_{\beta,\omega}}{2L_dL_q} & \frac{j_{\beta,\theta}}{L_dL_q} & 0 & 0 \end{bmatrix}^T \end{bmatrix}, \tag{6.30}$$

$$j_{\beta,\omega} = -p\Delta L(L_d+L_q)(i_\alpha\cos(2\theta) + i_\beta\sin(2\theta)) - 2K_f pL_d\cos(\theta) - pi_\alpha\Delta L^2, \tag{6.31}$$

$$\begin{aligned} j_{\beta,\theta} = {} & -\Delta L\cos(2\theta)(-i_\alpha R + i_\beta p\omega(L_d+L_q) + u_\alpha) + \Delta L\sin(2\theta)(i_\alpha p\omega(L_d+L_q) + i_\beta R - u_\beta) \\ & + K_f L_d p\omega\sin(\theta), \end{aligned} \tag{6.32}$$

$$\mathbf{j}_\omega^T = \begin{bmatrix} \frac{3p(\Delta L(i_\beta\cos(2\theta)-i_\alpha\sin(2\theta))-K_f\sin(\theta))}{2J} \\ \frac{3p(\Delta L(i_\alpha\cos(2\theta)+i_\beta\sin(2\theta))+K_f\cos(\theta))}{2J} \\ \begin{bmatrix} \frac{-F}{J} & \frac{j_{\omega,\theta}}{2J} & \frac{-1}{J} & 0 \end{bmatrix}^T \end{bmatrix}, \tag{6.33}$$

$$j_{\omega,\theta} = -3p\Big[i_\alpha\cos(\theta)(4i_\beta\Delta L\sin(\theta) + K_f) + (i_\alpha - i_\beta)(i_\alpha + i_\beta)\Delta L\cos(2\theta) + i_\beta K_f\sin(\theta)\Big]. \tag{6.34}$$

The Jacobian is also discretized using an RK3 scheme:

$$\mathbf{J}_{\mathbf{f}d}(k) = \mathrm{I}_{6\times 6} + T_s\left(\frac{1}{6}\mathbf{J}_A + \frac{4}{6}\mathbf{J}_B + \frac{1}{6}\mathbf{J}_C\right) \tag{6.35}$$

$$\mathbf{J}_A = \mathbf{J}_{\mathbf{f}}\left(\hat{\mathbf{z}}^+(k-1), \mathbf{u}(k-1)\right)\ , \tag{6.36}$$

$$\mathbf{J}_B = \mathbf{J}_{\mathbf{f}}\left(\hat{\mathbf{z}}^+(k-1) + \frac{T_s}{2}\mathbf{f}_A, \mathbf{u}(k-1)\right)\ , \tag{6.37}$$

$$\mathbf{J}_C = \mathbf{J}_{\mathbf{f}}\left(\hat{\mathbf{z}}^+(k-1) - T_s\mathbf{f}_A + 2T_s\mathbf{f}_B, \mathbf{u}(k-1)\right). \tag{6.38}$$

The a-priori estimation of the covariance is as follows:

$$\mathbf{P}^-(k) = \mathbf{J}_{\mathbf{f}d}(k)\mathbf{P}^+(k-1)\mathbf{J}_{\mathbf{f}d}(k)^T + \mathbf{Q}_{KF}, \tag{6.39}$$

whereupon the Kalman gain is

$$\mathbf{K}(k) = \mathbf{P}^-(k)\mathbf{C}_{KF}^T\left(\mathbf{C}_{KF}\mathbf{P}^-(k)\mathbf{C}_{KF}^T + \mathbf{R}_{KF}\right)^{-1}. \tag{6.40}$$

A-posteriori estimates of states and covariance are obtained in the update step, using measurements of $i_{\alpha,\beta}$ and ω_{el}

$$\hat{\mathbf{z}}^+(k) = \hat{\mathbf{z}}^-(k) + \mathbf{K}(k)\left(\begin{bmatrix} i_\alpha(k) \\ i_\beta(k) \\ \omega_{el}(k) \end{bmatrix} - \mathbf{C}_{KF}\hat{\mathbf{z}}^-(k)\right), \tag{6.41}$$

$$\mathbf{P}^+(k) = \left(\mathbf{I}_{6\times 6} - \mathbf{K}(k)\mathbf{C}_{KF}\right)\mathbf{P}^-(k). \tag{6.42}$$

The matrices $\mathbf{Q}_{KF}$ and $\mathbf{R}_{KF}$ can be considered as tuning parameter matrices and correspond to the process and measurement uncertainty, respectively. The EKF algorithm is implemented in discrete-time and is used for all state feedback and disturbance compensation purposes, with one notable exception being the mechanical angular acceleration $\dot{\omega}$. Since the main EKF already estimates the velocity ω as well as the load torque, it is difficult to include yet another augmented state that will allow another quantity to be estimated at the torque level. For this reason, an additional Kalman filter is designed, with the sole purpose of estimating $\dot{\omega}$. This Kalman filter results to be linear, since no nonlinear dynamics have to be considered, and will be detailed in the following section.

6.4 KF-based estimation of the acceleration

Another Kalman filter for the estimation of the angular acceleration is then derived, since this quantity is required for the computation of the sliding surface in (6.52). In the following, the abbreviation KF always denotes the secondary observer that is dedicated to the estimation of $\dot{\omega}$, while the abbreviation EKF always refers to the main state observer discussed in the last section. The working principle of the secondary observer is to combine the numerical differentiation of the EKF-estimated velocity $\hat{\omega}$ and a model-based approximation of $\dot{\omega}$ in an optimal way (such that the covariance is minimized). By combining these two basic approaches to approximate acceleration with the powerful Kalman filter algorithm, a very fast, yet denoised, estimate of the angular acceleration can be obtained. Since Kalman filters always require an underlying system model in order to compute the state prediction, an integrator disturbance model with no inputs is again used. The two modelled states are $\mathbf{z}_\omega = \begin{bmatrix} z_{1\omega} & z_{2\omega} \end{bmatrix}^T$, with $z_{1\omega} \hat{=} \omega$ and $z_{2\omega} \hat{=} \dot{\omega}$. Using an explicit Euler discretization (which works sufficiently well in this case,) the state prediction equations are obtained as

$$\hat{z}_{1\omega}^-(k) = \hat{z}_{1\omega}^+(k-1) + T_s\hat{z}_{2\omega}^+(k-1), \tag{6.43}$$

$$\hat{z}_{2\omega}^-(k) = \hat{z}_{2\omega}^+(k-1). \tag{6.44}$$

The "measured" outputs are both states. For $z_{1\omega,m}$, the actual velocity estimate $\hat{\omega}$ generated by the EKF is used. As $z_{2\omega,m}$, an acceleration "measurement" based on the mechanical model (6.7) is used, with EKF-estimated states

$$z_{2\omega,m} = \frac{3p}{2J}\left(\Delta L \hat{i}_d \hat{i}_q + K_f \hat{i}_q\right) - \frac{F}{J}\hat{\omega} - \frac{1}{J}\hat{T}_l. \tag{6.45}$$

Hence, the output map $\mathbf{C}_{\omega KF}$ is the 2×2 identity matrix, while the discrete system matrix is simply

$$\mathbf{A}_{\omega KF} = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}. \tag{6.46}$$

The filter equations for this KF are structurally almost the same as for the EKF, where a state Jacobian $\mathbf{J}$ is used instead of the discrete system matrix $\mathbf{A}_{\omega KF}$. However, the equations can be simplified substantially, thanks to $\mathbf{C}_{\omega KF} = \mathbf{I}_{2\times 2}$:

$$\mathbf{P}_\omega^-(k) = \mathbf{A}_{\omega KF}\mathbf{P}_\omega^+(k-1)\mathbf{A}_{\omega KF}^T + \mathbf{Q}_{\omega KF}, \tag{6.47}$$

$$\mathbf{K}_\omega(k) = \mathbf{P}_\omega^-(k)\left(\mathbf{P}_\omega^-(k) + \mathbf{R}_{\omega KF}\right)^{-1} \tag{6.48}$$

$$\hat{\mathbf{z}}_\omega^+(k) = \hat{\mathbf{z}}_\omega^-(k) + \mathbf{K}_\omega(k)\left(\begin{bmatrix} z_{1\omega,m}(k) \\ z_{2\omega,m}(k) \end{bmatrix} - \hat{\mathbf{z}}_\omega^-(k)\right), \tag{6.49}$$

$$\mathbf{P}_\omega^+(k) = \left(\mathbf{I}_{2\times 2} - \mathbf{K}_\omega(k)\mathbf{C}_{\omega KF}\right)\mathbf{P}_\omega^-(k). \tag{6.50}$$

6.5 An optimal sliding mode control

This section is devoted to the derivation of an optimal SMC for direct speed control. Two sliding mode surfaces are defined. The first is based on the angular velocity error, while the second one is based on the dynamics of a criterion that represents the optimal MTPA trajectory. This criterion is the solution of a power optimization problem and is given by

$$c_{MTPA}(t) = i_d(t) + \frac{\Delta L}{K_f}\left(i_d(t)^2 - i_q(t)^2\right), \tag{6.51}$$

in which $\Delta L = L_d - L_q$. The case $c_{MTPA}(t) = 0$ represents the locus of $i_d(t)$ and $i_q(t)$ in which the optimal working condition in terms of MTPA is obtained [111].

Remark 6.3 *Enforcing $c_{MTPA}(t) = 0$ does* not *limit the reachability of the desired velocity. If the required torque (also considering possible load torques or disturbances) can be supplied by the PMSM in the first place, then $c_{MTPA}(t) = 0$ directly corresponds to the allocation of $i_{d,q}$ to the minimal stator current. Hence, the two sliding surfaces are compatible under all circumstances.*

Innovatively, the input voltages $u_{d,q}$ are minimized in order to avoid or to alleviate saturation effects while guaranteeing both velocity tracking and optimal currents according to the MTPA condition. A theorem is proven that shows optimal and efficient velocity tracking; this is obtained by minimizing the squared input voltages $u_{d,q}$ with additional equality constraints, i.e. $\dot{s}_{1,2}(t) = 0$.

Problem 6.1 *Find a stabilizing and optimizing pair of control laws $u_d(t)$, $u_q(t)$ that guarantees an asymptotic velocity tracking of the mechanical angular rotor velocity $\omega(t)$ for the system defined in (6.4-6.9), and at the same time, guarantees to stay on the MTPA trajectory to improve the efficiency of the motor.*

To solve problem 6.1, let us propose the following theorem, which constructively introduces a suitable cost function and shows how to find its unique solution, based on classic Lagrangian optimization.

Theorem 6.1 *Given the system (6.4-6.9), let*

$$s_1(t) = \dot{e}(t) + \alpha e(t) + \eta \int_0^t e(\tau)\mathrm{d}\tau \tag{6.52}$$

be a sliding surface with velocity error $e(t) = \omega_d(t) - \omega(t)$, where $\alpha > 0$ and $\eta > 0$ are tuning parameters. The second sliding surface, in which $\gamma > 0$, is

$$s_2(t) = c_{MTPA}(t) + \gamma \int_0^t c_{MTPA}(\tau)\mathrm{d}\tau. \tag{6.53}$$

A Lyapunov function candidate is introduced as

$$V(t) = \underbrace{\frac{1}{2}s_1(t)^2}_{V_1(t)} + \underbrace{\frac{1}{2}s_2(t)^2}_{V_2(t)}. \tag{6.54}$$

There exists a pair of globally stabilizing control laws $u_d(t)$, $u_q(t)$ which guarantees prescribed closed-loop error dynamics with the chosen decay rates $\beta_{1,2} > 0$

$$\dot{s}_1(t) = -\beta_1 \mathrm{sgn}\left(s_1(t)\right), \tag{6.55}$$
$$\dot{s}_2(t) = -\beta_2 \mathrm{sgn}\left(s_2(t)\right), \tag{6.56}$$

as well as yielding

$$\frac{\mathrm{d}V(t)}{\mathrm{d}t} < 0, \tag{6.57}$$

i.e. global asymptotic convergence of both the velocity error $e(t)$ and the MTPA error $c_{MTPA}(t)$, by combining a pair of equivalent controls $u_{d_{eq}}$ and $u_{q_{eq}}$, which are the global minimizing solution of the Lagrangian function

$$\mathcal{C}(u_d(t), u_q(t), \lambda_1, \lambda_2) = u_d^2(t) + u_q^2(t) + \lambda_1 \dot{s}_1(t) + \lambda_2 \dot{s}_2(t), \tag{6.58}$$

with corrective parts according to

$$u_d(t) = u_{d_{eq}}(t) + u_{d_{sw}}(s_1, s_2), \tag{6.59}$$
$$u_q(t) = u_{q_{eq}}(t) + u_{q_{sw}}(s_1, s_2), \tag{6.60}$$

which are detailed in the proof.

Proof 6.1 *If the Lagrangian (6.58) is considered, in which*

$$\dot{s}_1(t) = \ddot{e}(t) + \alpha\dot{e}(t) + \eta e(t) = 0, \tag{6.61}$$
$$\dot{s}_2(t) = \dot{c}_{MTPA}(t) + \gamma c_{MTPA}(t) = 0 \tag{6.62}$$

state the constraints required to calculate the unique optimal equivalent solution *of the defined SMC problem, then*

$$\nabla\mathcal{C}(\tilde{u}_d(t), \tilde{u}_q(t), \lambda_1, \lambda_2) = \mathbf{0} \tag{6.63}$$

represents the condition used to symbolically calculate the unique pair of minimizing solutions $\tilde{u}_d(t)$ and $\tilde{u}_q(t)$, which can be done automatically using computer algebra software. These algebraic expressions are used as the equivalent controls $u_{d_{eq}} = \tilde{u}_d$ and $u_{q_{eq}} = \tilde{u}_q$ and are complemented by corrective components $u_{d_{sw}}$ and $u_{q_{sw}}$ (to overcome imperfect cancellations through the equivalent part of the control law) according to the following solution, which globally stabilizes the control system:

$$u_d(t) = u_{d_{eq}}(t) + \underbrace{\xi_{d1}(t)\beta_1 \mathrm{sgn}\left(s_1(t)\right) + \xi_{d2}(t)\beta_2 \mathrm{sgn}\left(s_2(t)\right)}_{u_{d_{sw}}}, \tag{6.64}$$

$$u_q(t) = u_{q_{eq}}(t) + \underbrace{\xi_{q1}(t)\beta_1 \mathrm{sgn}\left(s_1(t)\right) + \xi_{q2}(t)\beta_2 \mathrm{sgn}\left(s_2(t)\right)}_{u_{q_{sw}}}. \tag{6.65}$$

The expressions $\xi_{d1,2}$ and $\xi_{q1,2}$ are used to achieve the prescribed error dynamics (6.55-6.56) and were determined using computer algebra software, as well:

$$\xi_{d1}(t) = \frac{-3K_f p(i_d(t)\Delta L + K_f)JL_d}{N(t)}, \tag{6.66}$$

$$\xi_{d2}(t) = \frac{4i_q(t)J^2L_d\Delta L}{N(t)}, \tag{6.67}$$

$$\xi_{q1}(t) = \frac{3JL_q i_q(t)K_f p\Delta L}{N(t)}, \tag{6.68}$$

$$\xi_{q2}(t) = \frac{2J^2L_q(2i_d(t)\Delta L + K_f)}{N(t)}, \tag{6.69}$$

$$N(t) = 3Jp\Big(2\big(i_d(t)^2 + i_q(t)^2\big)\Delta L^2 + 3i_d(t)K_f\Delta L + {K_f}^2\Big). \tag{6.70}$$

Eq. (6.63) yields the following expressions (after eliminating λ_1, λ_2), in which multiplicative expressions spanning multiple rows are enclosed within square brackets, expressions in round brackets always end on the same line, and the symbol $\Delta L = L_d - L_q$:

$$\begin{aligned} u_{d_{eq}}(t) = \frac{1}{N(t)}\Bigg[&\Delta L\Big[2Fi_q(t)L_d\left(3i_q(t)K_f p - 2T_l(t) - 2\alpha J\omega(t)\right) \\ &-4F^2 i_q(t)L_d\omega(t) + 3J\gamma K_f L_d p\left(i_q(t)^2 - 2i_d(t)^2\right) + 9Ji_d(t)^2K_f pR - 9Ji_d(t)i_q(t)K_f L_q p^2\omega(t) + 4i_q(t)J^2L_d\ddot{\omega}_d(t) \\ &+ 4Ji_q(t)L_d\left(\dot{T}_l(t) - J\eta\omega(t)\right) + 4\eta i_q(t)J^2L_d\omega_d(t) + 2J\alpha i_q(t)L_d\left(2T_l(t) + 2J\dot{\omega}_d - 3i_q(t)K_f p\right)\Big] \\ &- 3JK_f^2 p\left(\gamma i_d(t)L_d - i_d(t)R + i_q(t)L_q p\omega(t)\right) + 3\Delta L^2 p\Big[2Fi_d(t)i_q(t)^2L_d + J\gamma L_d i_d(t)\left(i_q(t)^2 - i_d(t)^2\right) \\ &- 2J\alpha i_d(t)i_q(t)^2L_d + 2J\left(i_q(t)^2 + i_d(t)^2\right)\left(i_d(t)R - i_q(t)L_q p\omega(t)\right)\Big]\Bigg], \end{aligned} \tag{6.71}$$

$$\begin{aligned} u_{q_{eq}}(t) = \tfrac{1}{N(t)}\Bigg[&K_f\Big[FL_q\left(3i_q(t)K_f p - 2T_l(t) + 2\alpha J\omega(t)\right) + 3JK_f p\left(i_q(t)R + \omega(t)\left(K_f + i_d(t)L_d\right)p\right) \\ &- 2F^2L_q\omega(t) + J\alpha L_q\left(2T_l(t) + 2J\dot{\omega}_d(t) - 3i_q(t)K_f p\right) + 2JL_q\left(\dot{T}_l(t) + J\left(\ddot{\omega}_d(t) + \eta\omega_d(t) - \eta\omega(t)\right)\right)\Big] \\ &+ \Delta Li_d(t)\Big[FL_q\left(9i_q(t)K_f p - 4T_l(t) + 4\alpha J\omega(t)\right) - 4F^2L_q\omega(t) + 3J\gamma i_q(t)K_f L_q p \\ &+ 9JK_f p\left(i_q(t)R + \omega(t)\left(K_f + i_d(t)L_d\right)p\right) + J\alpha L_q\left(4\left(T_l(t) + J\dot{\omega}_d(t)\right) - 9i_q(t)K_f p\right) \\ &+ 4JL_q\left(\dot{T}_l(t) + J\left(\ddot{\omega}_d(t) + \eta\omega_d(t) - \eta\omega(t)\right)\right)\Big] + 3\Delta L^2 p\Big[i_d(t)^2 i_q(t)L_q\left(2F - 2J\alpha\right) \\ &+ 2J\left(i_d(t)^2 + i_q(t)^2\right)\left(i_q(t)R + \omega(t)\left(K_f + i_d(t)L_d\right)p\right) + J\gamma L_q\left(i_d(t) - i_q(t)\right)i_q(t)\left(i_d(t) + i_q(t)\right)\Big]\Bigg]. \end{aligned} \tag{6.72}$$

Considering realistic assumptions about the system parameters

$$J > 0, \tag{6.73}$$

$$p > 0, \tag{6.74}$$

$$K_f > 0, \tag{6.75}$$

$$L_d > 0, \tag{6.76}$$

$$L_q > L_d, \tag{6.77}$$

it is possible to show that the control is non-singular at all times: Analyzing $N(t) > 0$ in the unfavourable case $i_q(t) = 0$ yields two boundary conditions for $i_d(t)$

$$i_d(t) < -\frac{K_f}{2\Delta L}, \text{ or} \tag{6.78}$$

$$i_d(t) > -\frac{K_f}{\Delta L}. \tag{6.79}$$

Both boundaries are positive for machines with $L_q > L_d$, as per assumption. Since the goal is $i_d(t) \leq 0$ (i.e. meeting the MTPA condition $c_{MTPA}(t) = 0$), the smaller value is considered as a boundary. Hence, the sufficient condition for non-singularity of the control laws results as

$$i_d(t) < -\frac{K_f}{2\Delta L}. \tag{6.80}$$

It is not a necessary condition – except at start-up – because $i_q(t) > 0$ helps to further increase $N(t)$. Apart from non-singularity of the control laws, controllability (both of the physical system, which is guaranteed anyway, and for the implemented control system) is another necessary condition in the field of sliding mode methods. The only potentially problematic case for the controllability arises if $i_{d,q} = 0$, i.e. before start-up or on no-load standstill, yielding $\xi_{d2} = 0$ and $\xi_{q1} = 0$. However, thanks to the equivalent parts of the control laws and thanks to $\xi_{d1}(t) < 0$ and $\xi_{q2}(t) > 0$ for all t, the overall system results to be controllable.

□

Remark 6.4 *For realistic values of K_f and ΔL, condition (6.80) is always met.*

6.6 Simulation study

In the simulation study, the system depicted in Fig. 6.1 is implemented in Matlab/Simulink R2020a Update 3, running on a standard desktop computer with *Intel Skylake* CPU on Linux. As mentioned above, the PMSM dynamics are simulated in the stator-fixed α, β coordinate system.

6.6.1 Simulation setup and tuning of controllers and observers

The desired velocity profile is a smooth *A to B* signal, generated using the sigmoid function tanh:

$$\omega_d(t) = \omega_{max} \frac{1 + \tanh\left(m_\omega(t - t_{on})\right)}{2}, \tag{6.81}$$

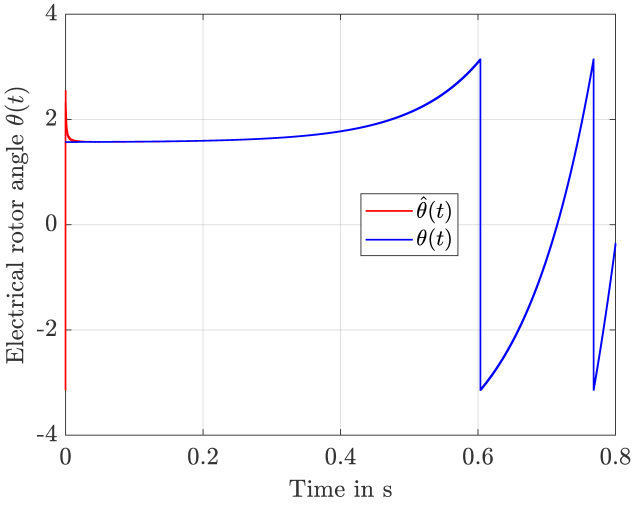
$$\dot{\omega}_d(t) = \frac{1}{2} m_\omega \omega_{max} \operatorname{sech}\left(m_\omega(t - t_{on})\right)^2, \tag{6.82}$$

$$\ddot{\omega}_d(t) = m_\omega^2 \omega_{max} \tanh\left(m_\omega(t - t_{on})\right) \operatorname{sech}\left(m_\omega(t - t_{on})\right)^2. \tag{6.83}$$

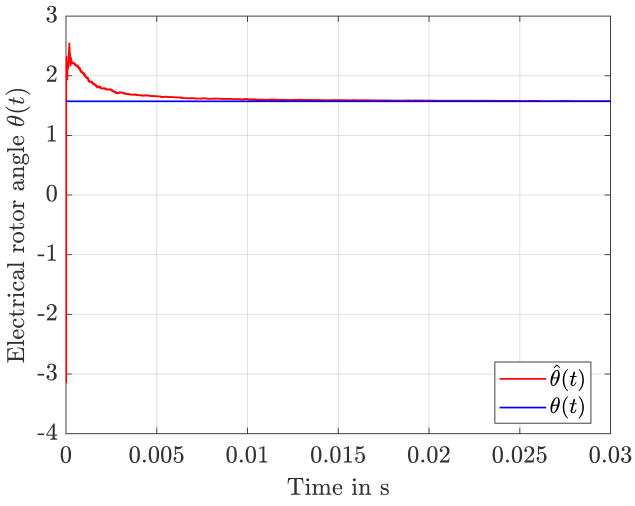
The target velocity is $\omega_{max} = 2000$ RPM ≈ 209.44 rad/s, time parameter $t_{on} = 1$ s, and slope $m_\omega = 5$. The fundamental sampling time for the whole system was chosen as $T_s = 0.01$ ms due to the fast dynamics (small inductances) of the electrical subsystems. All simulation results discussed in this section have been obtained with the control architecture shown in Fig. 6.1 and consider an IPMSM ($L_d < L_q$).

Step-like disturbances/load torques represent one of the worst cases for electrical machine control. Thus, in the simulation study, a repeating pulsed disturbance signal is synthesized, yielding a continuous step-like load torque profile that is applied to the PMSM and can be seen in Fig. 6.4c.

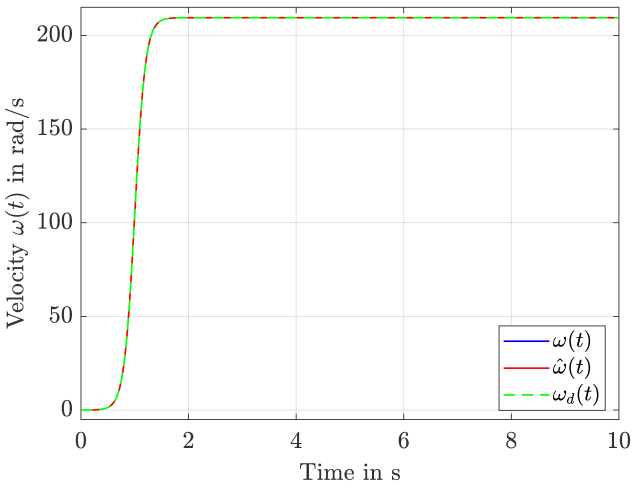
In the EKF that estimates the PMSM states, the electrical rotor angle θ is erroneously initialized as $\hat{\theta}(0) = -\pi$. This represents the worst possible case, because the true $\theta(0) = \frac{\pi}{2}$, yielding an initial error of 90° (due to the cyclic nature of $\theta(t) \in [-\pi, \pi)$ this is also equivalent to an error of 270°). The case $|\hat{\theta}(0) - \theta(0)| = 90°$ can be problematic because the estimated angle $\hat{\theta}$ is used in the Park



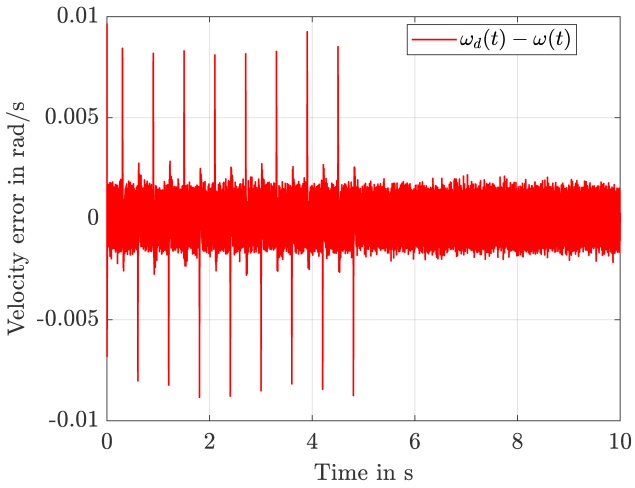
(a) Electrical rotor angle θ and its estimation from unknown initial conditions, $\theta(0) - \hat{\theta}(0) = 90°$.



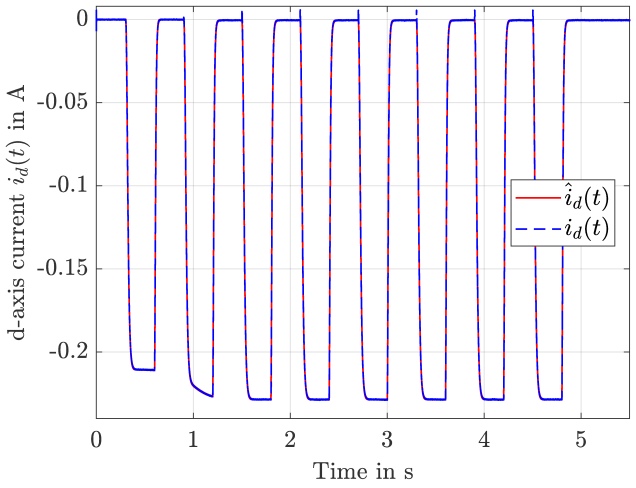
(b) Detailed view of the transient of the angle estimation.



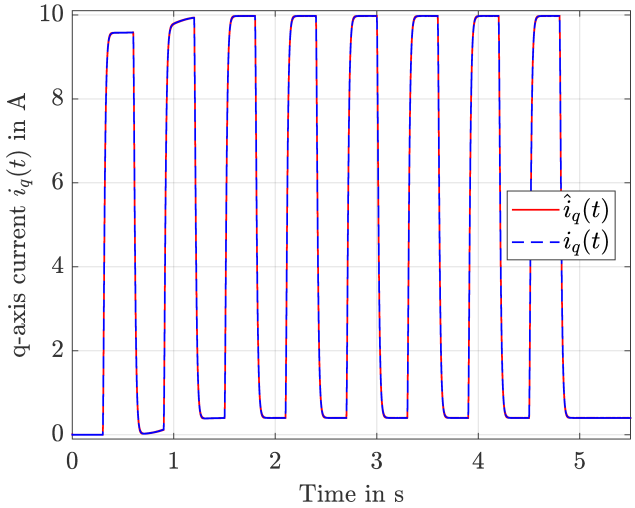
(c) Tracking of a given desired mechanical velocity signal for the rotor and estimated velocity.



(d) Velocity tracking error.

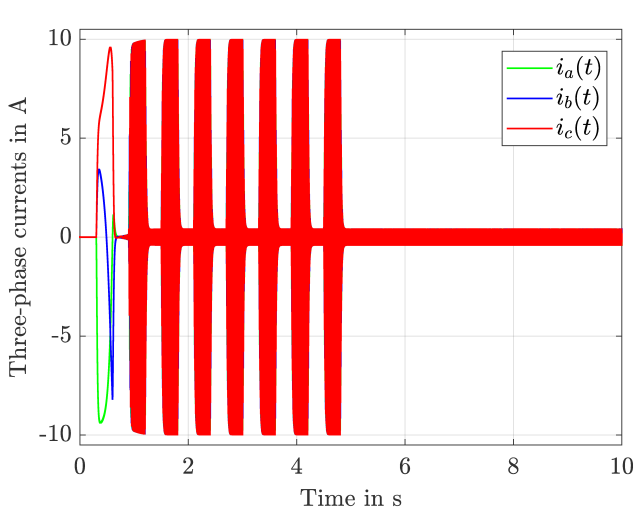


(e) Current i_d in the rotor-fixed reference system and its estimation, computed using $\hat{\theta}$.

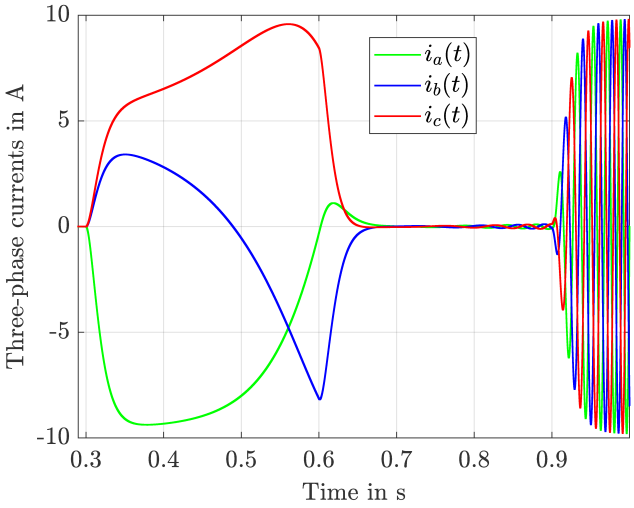


(f) Current i_q in the rotor-fixed reference system and its estimation, computed using $\hat{\theta}$.

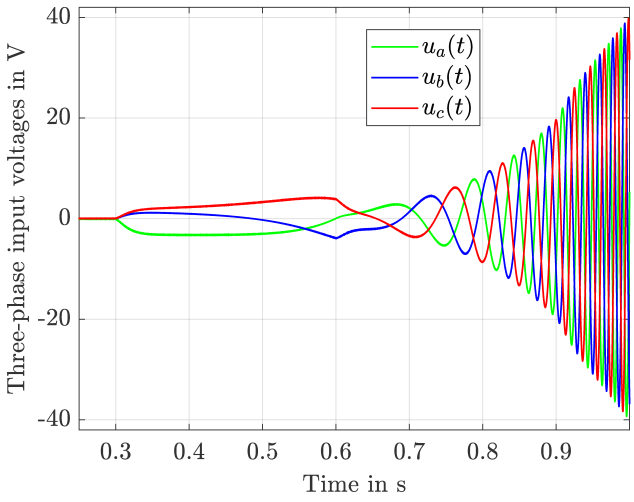
Figure 6.2: Simulation results.



(a) Reaction of the currents to step-like disturbances.



(b) Detailed view during transient and upon step-like loads.



(c) Three-phase input voltages from stand-still.

Figure 6.3: Three-phase currents and voltages.

transformation of the estimated currents, which are used for feedback control, and also in the inverse d-q transformation employed to obtain the three-phase input voltages. In some less-sophisticated sensorless control strategies, an angle error of $\pm 90°$ can cause the motor to rotate in the wrong direction, not to move at all, or to show other faulty behavior that sometimes cannot be perceived by the controller, due to confused d- and q-axes.

All PMSM states are initialized as zero in the simulation. The measured quantities, i.e. the three-phase currents and the electrical rotor angular velocity ω_{el}, contain synthetic additive uncorrelated white noise with realistic variances (comparable to common sensor systems). Only EKF-estimated states are used for the feedback control, allowing for reduced noise at a minimum delay. The chosen parameters of the main EKF are

$$\begin{aligned}
\hat{\mathbf{z}}^{+}(0) &= \begin{bmatrix} 0 & 0 & 0 & -\pi & 0 & 0 \end{bmatrix}^{T}, \\
\mathbf{P}^{+}(0) &= \operatorname{diag}\left(10, 10, 10^{3}, 5 \times 10^{6}, 1, 100\right), \\
\mathbf{Q}_{KF} &= \operatorname{diag}\left(10^{-3}, 10^{-3}, 10^{-3}, 0, 0, 100\right), \\
\mathbf{R}_{KF} &= \mathbf{I}_{3\times 3}.
\end{aligned} \tag{6.84}$$

In the implementation of the SMC, another Kalman filter (see section 6.4) is used to estimate the

mechanical angular acceleration of the rotor, $\hat{\dot{\omega}}$, which is used in the computation of s_1. This KF is tuned as

$$\hat{\mathbf{z}}_{\omega}^{+}(0) = \begin{bmatrix} 0 & 0 \end{bmatrix}^T, \qquad \mathbf{P}_{\omega}^{+}(0) = \mathrm{diag}\,(0,0)\,,$$
$$\mathbf{Q}_{\omega KF} = \mathrm{diag}\,(1,100)\,, \qquad \mathbf{R}_{\omega KF} = \mathrm{diag}\left(1, 10^4\right). \tag{6.85}$$

The equivalent control part of the SMC is tuned indirectly, using parameters α, η and γ of the sliding surfaces. These are chosen using an auxiliary parameter $\nu = 10^4$, which (when only the equivalent controls are used, without the corrective parts) can be interpreted similarly to an eigenvalue of the closed-loop system. Parameter ν should be chosen no larger than $\frac{1}{10T_s}$. In practice, when using real motors, smaller values should suffice. In the presented case,

$$\alpha = 2\nu, \quad \eta = \nu^2 \tag{6.86}$$

are used for the experiment. Regarding the MTPA criterion, γ can be set to the same value as α in order to obtain similarly fast error dynamics. The switching amplitudes for the corrective part of the controls are chosen as $\beta_1 = 10^7$ and $\beta_2 = 10^4$. In order to reduce chattering effects, boundary layers around the sliding surfaces $s_{1,2}$ are introduced, with *thicknesses* $\Phi_{1,2}$. These describe the region where one may stay without further switching. This is achieved by replacing the switching terms $\mathrm{sgn}(x)$ by

$$\mathrm{sat}\left(\frac{x}{\Phi}\right), \tag{6.87}$$

where sat is the -1, 1 saturation function. In the tuning of the thickness, the resulting noise levels of the computed $s_{1,2}(t)$ should be considered, as such noise causes dramatically increased chattering levels in the inputs. This, in turn, depends on the measurement noise, motor parameters, the tuning of the state observer and the acceleration observer, and other possible influences. Such effects can be alleviated using a larger boundary-layer thickness. In this case, thanks to the denoising KF that estimates the acceleration, it turns out that a thickness of $\Phi_1 = 100$ is large enough for s_1, while $\Phi_2 = 0.1$.

6.6.2 Discussion of the results

The estimation performance of the electrical rotor angle θ can be assessed in Fig. 6.2a. Even though the initial estimate is 90° off, the EKF quickly converges to the correct angle after approximately 10 ms, when the motor slowly begins starting up, see Fig. 6.2b. The tracking of the desired velocity profile, along with the estimated mechanical rotor angular velocity, is depicted in Fig. 6.2c. Even small deviations from the desired velocity trajectory can be compensated for by the controller. The EKF, thanks to feeding back the measured electrical rotor angular velocity, is able to closely follow the true state evolution. The sophisticated control strategy derived in the previous section, combined with such high-quality estimates, leads to a satisfactory velocity tracking error, as shown in Fig. 6.2d. The small remaining deviations (in the instants of time when the step-like load torque changes) are of the same order of magnitude as the peak simulated measurement noise of ω_{el} divided by the number of pole pairs, p. The repeating spikes are due to the disturbance torque signal described in the previous subsection. It also influences the resulting d- and q-axis currents, which are shown in Figs. 6.2e-f together with their estimates. These are computed *outside* of the EKF using its estimated states $\hat{i}_{\alpha,\beta}$ and Park's transformation with estimated angle $\hat{\theta}$ (see also Fig. 6.1).

The controller computes the control inputs u_d and u_q; their individual components according to Eqs. (6.64-6.65) are shown in Figs. 6.4a-b. Note that, due to the enforced MTPA, flux weakening is applied, which yields negative d-axis voltages and currents. This effect is particularly pronounced in the switching component of u_d. Thanks to the employed boundary-layer approach with a permissible region $\Phi_{1,2}$ around $s_{1,2}(t)$, the switching part $u_{q_{sw}}$ is merely a low-voltage component of the total input voltage and shows only mild chattering effects.

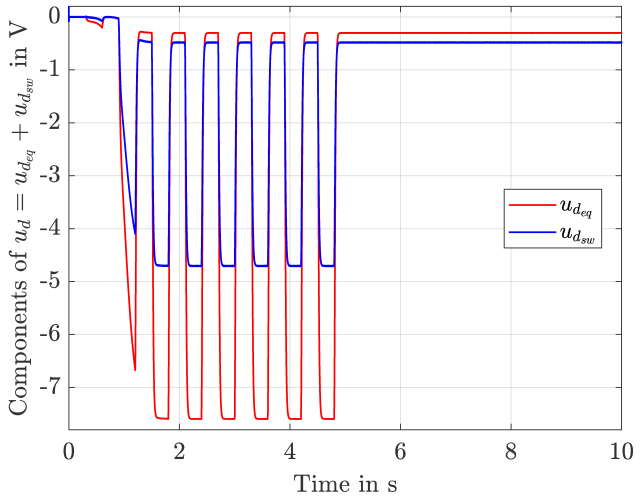
In many of the above-mentioned diagrams, the step-like nature of the simulated disturbance/load torque is visible, which is shown in Figs. 6.4c-d, together with its derivative and the corresponding

estimates by the EKF. Again, thanks to the feeding back of ω_{el}, the disturbance torque can be estimated by the EKF effortlessly. The estimated derivative of the disturbance shows a small delay, but the EKF is able to attenuate any visible noise. In case of truly discontinuous step disturbances (instead of the simulated continuous signal), this estimate would appear more like a Dirac pulse with a magnitude and width that are determined mainly by the corresponding element of the EKF tuning parameter matrix $\mathbf{Q}_{KF}$. However, *real* discontinuous step disturbances do not happen in real-world mechanical systems; there are always slip effects and similar tendencies to smooth out the actual load torque. Thus, this issue is not too problematic.

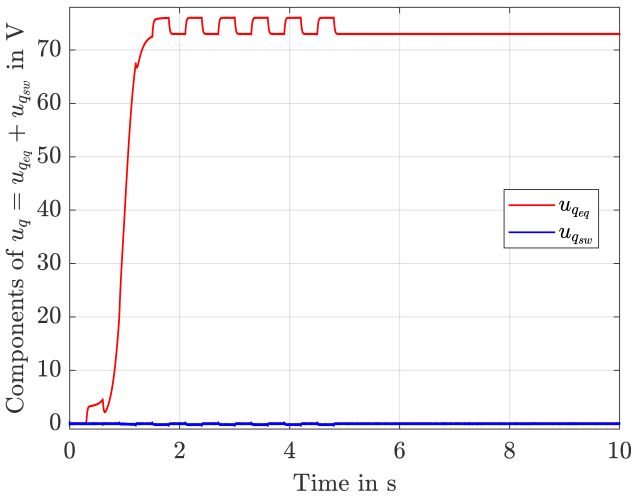
In Fig. 6.4e, the value of the components of sliding surface s_1 can be observed. Apart from some very short time instances of rapid load change, its value (being the sum of the depicted signals) stays within the boundary layer. The same is true for surface s_2, shown in Fig. 6.4f, where the remaining offset of the integral term of the surface (again, due to the chosen permissible boundary-layer thickness) can safely be ignored, as c_{MTPA} vanishes.

Since the voltages u_d and u_q are rather abstract quantities with no simple physical interpretation, the actual three-phase input voltages that are applied to the PMSM during the start-up phase are shown in Fig. 6.3c, where a sinusoidal progression can be observed thanks to the accurately estimated electrical rotor angle $\hat{\theta}$, which is used in the inverse d-q transformation step in the control scheme. The (mostly) sinusoidal three-phase currents are shown in Fig. 6.3a, and in detail in Fig. 6.3b, during a time interval in which the transient takes place and the step-like load torque becomes active.

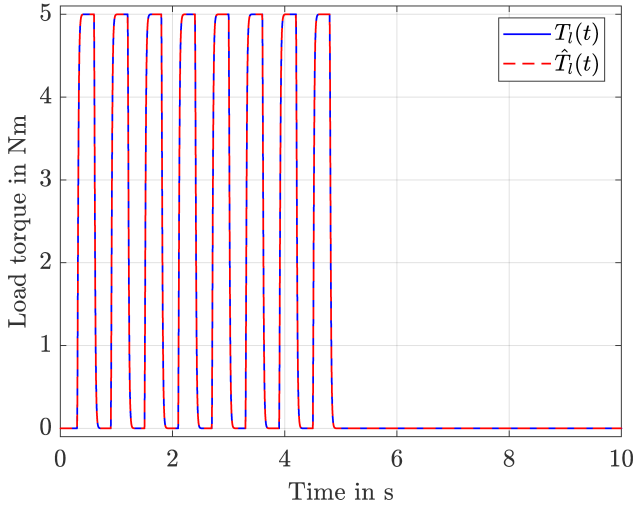
To conclude, this chapter presents a novel direct speed sliding mode control for permanent-magnet synchronous motors. The control laws consist of an optimized equivalent solution together with a corrective switching part that can be further refined using the well-known boundary-layer approach for chattering reduction. A convex cost function of the inputs is defined, containing two sliding surfaces' error dynamics as equality constraints. The first sliding surface's error dynamics realizes a velocity-tracking constraint. At the same time, thanks to the second sliding surface, the *maximum torque per ampere* condition is reached. The equivalent solution to the sliding mode control is minimal with respect to the input power, yielding a more efficient torque generation. In this context, the extended Kalman filter plays a crucial role because it is required in order to provide estimates of the load torque as well as its derivative, in addition to all system states. The control strategy is amended by a second Kalman filter, which is dedicated to the estimation of the angular acceleration. Computer simulations show that this approach to combined PMSM control and estimation is promising.



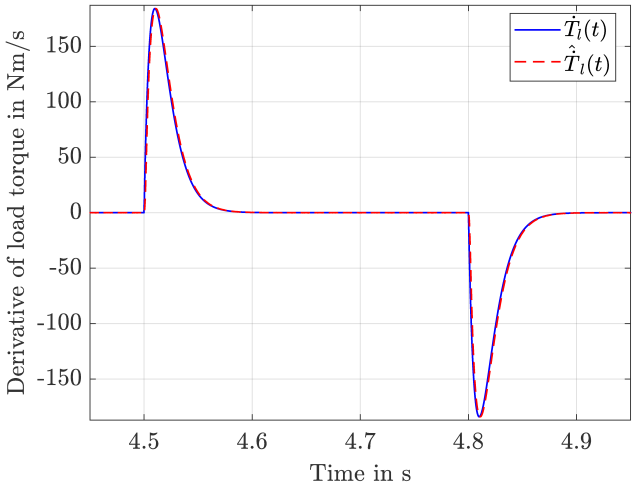
(a) Input voltage u_d in the rotor-fixed reference frame that is transformed back to the stator-fixed system using $\hat{\theta}$.



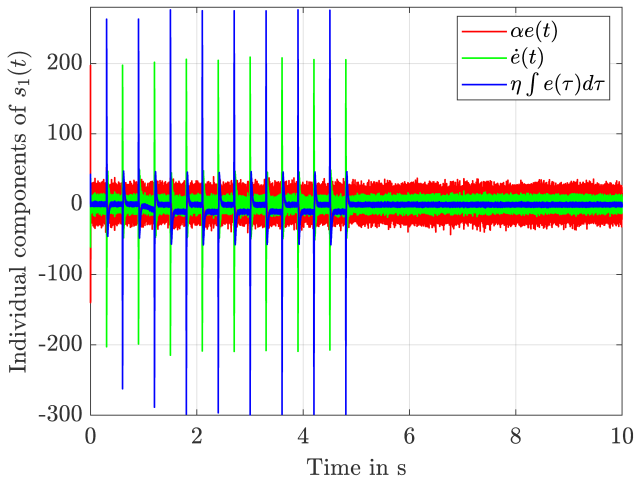
(b) Input voltage u_q in the rotor-fixed reference frame that is transformed back to the stator-fixed system using $\hat{\theta}$.



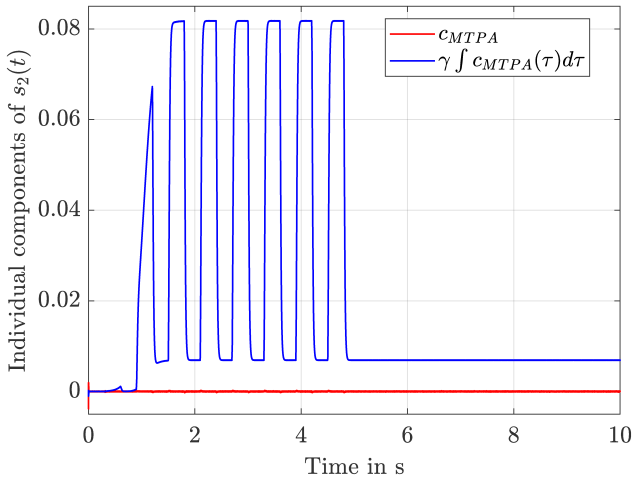
(c) Lumped torque disturbance or load and its estimate.



(d) Derivative of the load torque and its estimate.



(e) Components of the sliding surface $s_1(t)$.



(f) Components of the sliding surface $s_2(t)$. Note that $|s_2| < \Phi_2$; the remaining offset is tolerated (boundary layer approach).

Figure 6.4: Simulation results.

7 Conclusions and outlook

Minimum-variance state estimation and disturbance compensation, being two relatively recent approaches, are shown to significantly improve the performance of nonlinear systems subject to perturbation and uncertainty. In particular, state and disturbance estimators based on the Kalman filter family are used due to their (sub-)optimality and inherent robustness, even against implementation errors. In addition to control goals, these estimators are well-suited for fault detection and diagnosis purposes in safety-critical systems thanks to model-based state predictions. Regarding the control methods, the robustness property is perhaps strongest in sliding mode control, which (usually) works using very conservative tuning, i.e. high-gain control. Employing advanced, model-based control strategies is demonstrated to significantly mitigate the disadvantages that come with high-gain controllers, like control chattering, thanks to less (or smaller) discontinuities in the control.

Within the scope of this dissertation, the following innovations were contrived and disseminated.

Mathematical foundations and state of the art

A preparative chapter is dedicated to the fundamentals of observation, state estimation, and minimum variance observers, as well as common practices in applied control. After a brief introduction to the state space from the field of system theory and to nonlinear, continuous-time, observer-oriented modelling using differential equations, the concept of observability and the choice of measured system outputs is discussed. The derivation of the Kalman filter algorithm is then presented, using two intuitive approaches from the literature that are *not* based on analytic minimization of the covariance matrix, but on geometric considerations and on the iterative evolution of the discretized dynamics of the covariance matrix (algebraic Riccati equation) in a quadratic form. Particular emphasis is put on control applications of the estimation techniques, i.e. disturbance compensation and full state feedback.

Li-Ion state of charge estimation

However, as initially explained, control applications are not the only target of the methodological framework introduced in this thesis. Supervision of systems that potentially can cause harm is another field where observers tend to shine. Thus, the physical system first presented in this thesis is a Lithium ion battery cell, where the goal is to estimate the state of charge offset-free and without the knowledge of the initial true state of charge. Since most observers presented in the literature require initial conditions that are known with relatively high certainty, a robustified, self-adaptive EKF with redundant bases in the decomposition of the system output equation is used, yielding improved state of charge estimation. While the augmentation of actual, meaningful system parameters into the state vector that is to be estimated is a (more or less) established procedure in observer design, doing so with the coefficients of willfully redundant bases is much less orthodox. This seemingly problematic digital representation turns out to aid estimate convergence in a region of reduced observability.

Time delay estimation based on a fractional variable-order model

Another case where the model structure is intentionally different than the real system is shown in chapter 3, where the time delay phenomenon is approximately modelled using a first-order system. Theoretically, the time delay can be seen as a system of order zero, since it has no internal dynamics, or of order infinity (when representing the time delay using a Padé approximant or a Taylor series). Obviously,

first-order models by themselves cannot possibly be a representation of the time delay. To improve this inaccurate first-order model, methods from the field of fractional-order calculus can be advantageously employed. Since the fractional derivative requires infinite memory for correct implementation, yet another approximation (back to system order one) is introduced.

Combining this complicated reduced-order model with measurements allows offset-free delay estimates that are even able to follow variable time delays. Especially within control systems, the ability to react to changing time delays at any point in time is an important requirement if such phenomena are to be expected.

Piezo-hydraulic actuator state estimation and control

One of such systems where time-delay phenomena occur is a novel multi-stage actuator, intended to be used in camless combustion engines so open and close the intake and exhaust valves. Chapter 4 presents the model of this actuator, which consists of a stack of Piezo elements, a hydraulic displacement amplifier, a proportional valve, and a double-acting hydraulic cylinder (driven by a hydraulic pump and controlled using the proportional valve). Since the Piezo stack cannot achieve the necessary servo valve spool displacements of up to 10 mm, but only 0.1 mm, a transmission ratio of 100 is required. The necessary rubber sealings of the hydraulic chamber that constitutes the displacement amplifier show strong static friction effects due to the high pressure within the chamber. Periodic intake/exhaust valve trajectories, in combination with a model-based feedforward control action, then lead to a phenomenon that can be modelled as a time delay between the desired and the obtained intake/exhaust valve position signals, which is treated in chapter 3. Thus, chapter 4 considers the rest of the control architecture with the exception of the time delay phenomenon. However, due to continuing development, yet another new incarnation of the described actuator is treated here, where *two* mirrored, antagonistic Piezo stacks are used instead of one (replacing the return spring and yielding better tracking accuracy). The feedback control scheme for this actuator consists of a feedforward control, based on approximate inversion of the inverted linearized system, an input-output linearization scheme (based on Isidori theory) that effectively cancels out the nonlinear part of the dynamics, as well as LQI (LRQ + Integral) control of the resulting virtual IT2 system (PT1 and two integrals), yielding position tracking of the intake and exhaust valves. Since this involved control method requires all state signals to be available, a novel type of EKF for state estimation is proposed. It is based on a cascaded structure involving the linear and the nonlinear subsystem of the actuator, both of which are estimated using a dedicated KF. The Cascaded EKF (CEKF) has much better computational efficiency than a monolithic, full-order observer and is shown to be real-time capable despite the very fast system dynamics that are considered.

DC motor nonlinear adaptive control through disturbance estimation

Kalman filters, being the de-facto standard for state estimation, can also be used to estimate disturbances acting on the system, like unknown load torques (or unknown inputs in general). This is particularly neat if a Kalman filter is used anyway, for the state estimation task. Augmenting an additional state comes with a cost, especially if the size n of the state vector is large already, because the computation time of the (unoptimized) KF scales roughly with n^2. In such cases, the fundamental sampling time must be large enough to allow for the computation to complete (with the corresponding real-time overhead).

If the controller can be improved by compensating known disturbances and if the digital hardware is capable of the additional computational load, (E)KFs can be advantageously employed, as was done in chapter 5. Here, a DC motor subject to nonlinear friction effects (that are assumed to be unknown) and unknown load torques is controlled using SMC. In order to avoid the common chattering problem that occurs in SMC, four independent mitigations are used: equivalent control (feedback cancellation of undesired known residual dynamics), second-order SMC (using an integral sliding surface, which also increased robustness against disturbances), regularized SMC (replacing the cause of the chattering,

which is the sgn function, by a smooth approximation), and a new gain adaptation algorithm. The adaptivity of the switching amplitude, being the main contribution of the chapter, is enabled by compensating disturbances that are estimated by an EKF. The observer is a vital part of the combined control strategy as it is a prerequisite for model matching, i.e. static and dynamic system behavior that is consistent with the model, because the disturbances are not known beforehand. The adaptation itself is no longer based on a model of the physical system, but on a model of the closed-loop sliding surface (error) dynamics. This allows to use the algorithm, which is based on linear MPC, in a plug & play manner for any system that can be controlled with SMC, not only DC motors.

Future work: Advanced sensorless PMSM control with SMC, EKF and DC

Figure 7.1: Hybrid motorcycle (work in progress), based on a *BMW F 800 GS Adventure* with a PMSM.

Passing over to the technically more complicated PMSM motors, the thesis culminates in chapter 6, where SMC and EKF are again combined to obtain velocity tracking – without rotor angle measurement, i.e. sensor-reduced control. The architecture of the control scheme is intended for small motors with very high power densities, where the mechanical subsystem is about as fast as the electrical subsystem. Such characteristics are problematic if the motor is controlled using the traditional cascaded control approach, where an inner current control PI(D) loop is combined with an outer speed/position control PI(D) loop. Hence, direct speed control is used, based on a novel two-surface SMC with MTPA-compliant current tracking and an optimal equivalent control law (and powered by EKF for state and disturbance estimation/compensation). The SMC, however, is not yet adaptive but used constant gains.

In the future, more work on *adaptive* PMSM control is planned, in the context of a research project regarding a motorcycle with parallel-hybrid drivetrain (combustion engine and PMSM, see Fig. 7.1) that is currently starting at the *Flensburg University of Applied Sciences*.

To demonstrate the effectiveness of the combination of observer-based disturbance compensation and advanced control strategies, several new or improved control and estimation methods based on "meta-methods" like Kalman filters or sliding mode control are derived and investigated, each one dedicated to an example system with high practical relevance. The improved control methods allow for reduced actuator wearout and energy consumption through control chattering suppression and/or for better, faster tracking performance. The improved observers, on the other hand, mainly enhace the accuracy of the state and disturbance estimates – almost always by augmenting the observer base model with additional unknown/uncertain quantities that appear in one of the model equations; that is, by *localizing* the main uncertainty present within a model equation into the chosen variable.

8 Bibliography

[1] B. Haus and P. Mercorelli. Polynomial augmented extended Kalman filter to estimate the state of charge of Lithium-ion batteries. *IEEE Trans. Veh. Technol.*, 69(2):1452–1463, 2020.

[2] B. Haus and P. Mercorelli. An extended Kalman filter for time delays inspired by a fractional order model. In *Non-Integer Order Calculus and its Applications*, volume 496 of *Lecture Notes in Electrical Engineering*, pages 151–163. Springer International Publishing, Cham, 2019.

[3] B. Haus, H. Aschemann, and P. Mercorelli. Tracking control of a piezo-hydraulic actuator using input-output linearization and a cascaded extended Kalman filter structure. *J. Franklin Inst.*, 355(18):9298–9320, 2018.

[4] B. Haus, P. Mercorelli, and H. Aschemann. Gain adaptation in sliding mode control using model predictive control and disturbance compensation with application to actuators. *Information*, 10(5):182, 2019.

[5] A. Mironova, B. Haus, A. Zedler, and P. Mercorelli. Extended Kalman filter for temperature estimation and control of Peltier cells in a novel industrial milling process. *IEEE Trans. Ind. Appl.*, 56(2):1670–1678, 2020.

[6] M. Schimmack, B. Haus, and P. Mercorelli. An extended Kalman filter as an observer in a control structure for health monitoring of a metal-polymer hybrid soft actuator. *IEEE/ASME Trans. Mechatronics*, 23(3):1477–1487, 2018.

[7] G. Welch and G. Bishop. An introduction to the Kalman filter. *Department of Computer Science, University of North Carolina at Chapel Hill, NC*, 2006.

[8] E. D. Sontag. *Mathematical Control Theory*. Springer New York, New York, NY, 1998.

[9] R. Hermann and A. Krener. Nonlinear controllability and observability. *IEEE Trans. Autom. Control*, 22(5):728–740, 1977.

[10] J. Lückel and P. C. Müller. Analyse von Steuerbarkeits-, Beobachtbarkeits- und Störbarkeitsstrukturen linearer zeitinvarianter Systeme [Analysis of controllability-, observability- and disturbability-structures in linear time invariant systems]. *at - Automatisierungstechnik*, 23(1-12), 1975.

[11] A. J. Krener and K. Ide. Measures of unobservability. In *Proc. 48th IEEE Conf. Decis. Control (CDC)*, pages 6401–6406, 2009.

[12] C. Letellier, L. Aguirre, and J. Maquet. How the choice of the observable may influence the analysis of nonlinear dynamical systems. *Comm. Nonlinear Sci. Numer. Simulat.*, 11(5):555 – 576, 2006.

[13] G. Lefebvre, J. Gauthier, A. Hijazi, X. Lin-Shi, and V. Le Digarcher. Observability-index-based control strategy for induction machine sensorless drive at low speed. *IEEE Trans. Ind. Electron.*, 64(3):1929–1938, 2017.

[14] A. Mironova, B. Haus, and P. Mercorelli. Combination of a reduced order state observer and an extended Kalman filter for Peltier cells. In *Proc. 19th Int. Carpathian Control Conf. (ICCC)*, pages 211–216, 2018.

[15] C. Bohn. *Recursive Parameter Estimation for Nonlinear Continuous-Time Systems through Sensitivity-Model-Based Adaptive Filters*. PhD thesis, Ruhr-Universität Bochum, Bochum, 2000.

[16] B. P. Gibbs. *Advanced Kalman filtering, least-squares and modeling: a practical handbook*. Wiley, Hoboken, NJ, 2011.

[17] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *J. Basic Eng.*, 82(1):35–45, 1960.

[18] C. Bohn. *Regelungstechnik III: Optimale und Robuste Regelung [Control engineering III: Optimal and robust control]*. Institut für elektrische Informationstechnik (IEI), TU Clausthal, Clausthal-Zellerfeld, Germany, 2011.

[19] B. Haus, P. Mercorelli, and N. Werner. Gaussian trajectories in motion control for camless engines. In *Proc. 21st Int. Conf. Process Control (PC)*, pages 76–80, 2017.

[20] C. A. Hart. Boeing 737 MAX Flight Control System. Observations, Findings, and Recommendations. Technical report, U.S. Federal Aviation Administration (FAA), Washington, DC, 2019.

[21] M. Schimmack, B. Haus, P. Leuffert, and P. Mercorelli. An extended Kalman filter for temperature monitoring of a metal-polymer hybrid fibre based heater structure. In *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatron. (AIM)*, pages 376–381, 2017.

[22] R. Xiong, Q. Yu, W. Shen, C. Lin, and F. Sun. A Sensor Fault Diagnosis Method for a Lithium-Ion Battery Pack in Electric Vehicles. *IEEE Trans. Power Electron.*, 34(10):9709–9718, 2019.

[23] V. Prajapati, H. Hess, E. J. William, V. Gupta, M. Huff, M. Manic, F. Rufus, A. Thakker, and J. Govar. A literature review of state of-charge estimation techniques applicable to Lithium poly-carbon monoflouride (LI/CFx) battery. In *India Int. Conf. Power Electron. (IICPE2010)*, pages 1–8, 2011.

[24] S. Rodrigues, N. Munichandraiah, and A. Shukla. A review of state of charge indication of batteries by means of A.C. Impedance measurements . *J. Power Sources*, 87(1-2):12–20, 2000.

[25] G. L. Plett. Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 2. Modeling and identification. *J. Power Sources*, 134(2):262–276, 2004.

[26] A. H. Ranjbar, A. Banaei, A. Khoobroo, and B. Fahimi. Online estimation of state of charge in Li-ion batteries using impulse response concept. *IEEE Trans. Smart Grid*, 3(1):360–367, 2012.

[27] B. Hariprakash, S. K. Martha, A. Jaikumar, and A. K. Shukla. Online monitoring of lead-acid batteries by galvanostatic non-destructive technique. *J. Power Sources*, 137(1):128–133, 2004.

[28] B. Jun, S. Sai, G. Wei, and W. Lu. State of charge estimation of Li-ion batteries in an electric vehicle based on a radial-basis-function neural network. *Chin. Phys. B*, 21(11):118801, 2012.

[29] B. S. Bhangu, P. Bentley, D. A. Stone, and C. M. Bingham. State-of-charge and state-of-health prediction of lead-acid batteries for hybrid electric vehicles using non-linear observers. In *Proc. Eur. Conf. Power Electron. Appl.*, 2005.

[30] K. Kutluay, Y. Cadirci, Y. S. Ozkazanc, and I. Cadirci. A new online state-of-charge estimation and monitoring system for sealed lead-acid batteries in telecommunication power supplies. *IEEE Trans. Ind. Electron.*, 52(5):1315–1327, 2005.

[31] M. Chen and G. A. Rincon-Mora. Accurate electrical battery model capable of predicting runtime and I-V performance. *IEEE Trans. Energy Convers.*, 21(2):504–511, 2006.

[32] H. He, R. Xiong, and J. Fan. Evaluation of Lithium-Ion Battery Equivalent Circuit Models for State of Charge Estimation by an Experimental Approach. *Energies*, 4(4):582–598, 2011.

[33] P. Shen, M. Ouyang, L. Lu, J. Li, and X. Feng. The co-estimation of state of charge, state of health, and state of function for lithium-ion batteries in electric vehicles. *IEEE Trans. Veh. Technol.*, 67(1):92–103, 2018.

[34] J. Kim and B. H. Cho. State-of-Charge Estimation and State-of-Health Prediction of a Li-Ion Degraded Battery Based on an EKF Combined With a Per-Unit System. *IEEE Trans. Veh. Technol.*, 60(9):4249–4260, 2011.

[35] J. Xu, C. C. Mi, B. Cao, J. Deng, Z. Chen, and S. Li. The state of charge estimation of lithium-ion batteries based on a proportional-integral observer. *IEEE Trans. Veh. Technol.*, 63(4):1614–1621, 2014.

[36] Z. Chen, Y. Fu, and C. C. Mi. State of charge estimation of lithium-ion batteries in electric drive vehicles using extended kalman filtering. *IEEE Trans. Veh. Technol.*, 62(3):1020–1030, 2013.

[37] X. Hu, H. Yuan, C. Zou, Z. Li, and L. Zhang. Co-estimation of state of charge and state of health for lithium-ion batteries based on fractional-order calculus. *IEEE Trans. Veh. Technol.*, 67(11):10319–10329, 2018.

[38] Q. Zhu, L. Li, X. Hu, N. Xiong, and G. Hu. H_∞-Based Nonlinear Observer Design for State of Charge Estimation of Lithium-Ion Battery With Polynomial Parameters. *IEEE Trans. Veh. Technol.*, 66(12):10853–10865, 2017.

[39] Q. Yu, R. Xiong, C. Lin, W. Shen, and J. Deng. Lithium-Ion Battery Parameters and State-of-Charge Joint Estimation Based on H-Infinity and Unscented Kalman Filters. *IEEE Trans. Veh. Technol.*, 66(10):8693–8701, 2017.

[40] W. Li, L. Liang, W. Liu, and X. Wu. State of charge estimation of lithium-ion batteries using a discrete-time nonlinear observer. *IEEE Trans. Ind. Electron.*, 64(11):8557–8565, 2017.

[41] F. Guo, G. Hu, S. Xiang, P. Zhou, R. Hong, and N. Xiong. A multi-scale parameter adaptive method for state of charge and parameter estimation of Lithium-ion batteries using dual Kalman filters. *Energy*, 178:79 – 88, 2019.

[42] R. Zhang, B. Xia, B. Li, L. Cao, Y. Lai, W. Zheng, H. Wang, W. Wang, and M. Wang. A study on the open circuit voltage and state of charge characterization of high capacity lithium-ion battery under different temperature. *Energies*, 11(9), 2018.

[43] H. He, R. Xiong, X. Zhang, F. Sun, and J. Fan. State-of-Charge Estimation of the Lithium-Ion Battery Using an Adaptive Extended Kalman Filter Based on an Improved Thevenin Model. *IEEE Trans. Veh. Technol.*, 60(4):1461–1469, 2011.

[44] A. P. Sage and G. W. Husa. Adaptive filtering with unknown prior statistics. In *Joint Auto. Contr. Conf.*, pages 760 – 769, 1969.

[45] H. J. Bergveld, W. S. Kruijt, and P. H. L. Notten. *Battery Management Systems: Design by Modelling*. Philips Research 1. Springer Netherlands, 1 edition, 2002.

[46] S. Zhao, S. R. Duncan, and D. A. Howey. Observability analysis and state estimation of lithium-ion batteries in the presence of sensor biases. *IEEE Trans. Control Syst. Technol.*, 25(1):326–333, 2017.

[47] C. Zhang, J. Jiang, L. Zhang, S. Liu, L. Wang, and P. C. Loh. A Generalized SOC-OCV Model for Lithium-Ion Batteries and the SOC Estimation for LNMCO Battery. *Energies*, 9(11), 2016.

[48] B. Adcock and D. Huybrechs. Frames and numerical approximation. *SIAM Review*, 61(3):443–473, 2019.

[49] M. Gholizadeh and F. R. Salmasi. Estimation of State of Charge, Unknown Nonlinearities, and State of Health of a Lithium-Ion Battery Based on a Comprehensive Unobservable Model. *IEEE Trans. Ind. Electron.*, 61(3):1335–1344, 2014.

[50] K. B. Oldham and J. Spanier. *The Fractional Calculus*. Academic Press, New York, 1974.

[51] K. S. Miller and B. Ross. *An Introduction to the Fractional Calculus and Fractional Differential Equations*. Wiley & Sons, New York, 1993.

[52] S. G. Samko, A. A. Kilbas, and O. I. Marichev. *Fractional Integrals and Derivatives: Theory and Applications*. Gordon and Breach Science, Amsterdam, 1993.

[53] I. Podlubny. *Fractional Differential Equations*. Academic Press, San Diego, 1999.

[54] R. Hilfer. *Applications of Fractional Calculus in Physics*. World Scientific, Singapore, 2000.

[55] P. Mercorelli. A discrete-time fractional order PI controller for a three phase synchronous motor using an optimal loop shaping. In A. Babiarz, A. Czornik, J. Klamka, and M. Niezabitowski, editors, *Theory and Applications of Non-integer Order Systems. Lecture Notes in Electrical Engineering*, pages 477–487. Springer International Publishing, Cham, 2017.

[56] S. Victor, P. Melchior, J. Lévine, and A. Oustaloup. Flatness for linear fractional systems with application to a thermal system. *Automatica*, 57:213 – 221, 2015.

[57] S. Ibrir and M. Bettayeb. New sufficient conditions for observer-based control of fractional-order uncertain systems. *Automatica*, 59:216 – 223, 2015.

[58] Y. H. Lan, W. J. Li, Y. Zhou, and Y. P. Luo. Non-fragile observer design for fractional-order one-sided lipschitz nonlinear systems. *Int. J. Autom. Comput.*, 10(4):2757–2764, 2013.

[59] X. J. Wen, Z. M. Wu, and J. G. Lu. Stability analysis of a class of nonlinear fractional-order systems. *IEEE Trans. Circuits Syst., II, Exp. Briefs*, 55(11):1178–1182, 2008.

[60] J. G. Lu and Y. Q. Chen. Robust stability and stabilization of fractional-order interval systems with the fractional order α : The $0 \ll \alpha \ll 1$ case. *IEEE Trans. Autom. Control*, 55(1):152–158, 2010.

[61] C. Li, J. Wang, and J. Lu. Observer-based robust stabilisation of a class of non-linear fractional-order uncertain systems: an linear matrix inequalitie approach. *IET Control Theory Appl.*, 6(18):2757–2764, 2012.

[62] B. Haus, P. Mercorelli, and N. Werner. Friction hysteresis compensation using phase correction of periodic trajectories. In *Proc. 22nd Int. Conf. on Methods and Models in Automation and Robotics (MMAR)*, pages 325–330, 2017.

[63] J. A. Tenreiro Machado. Analysis and design of fractional-order digital control systems. *Syst. Anal. Model. Sim.*, 27(2-3):107–122, 1997.

[64] N. Werner. *Entwicklung einer servohydraulischen vollvariablen Ventilpositionsregelung mit Piezoaktuator für einen Verbrennungsmotor*. PhD thesis, Fakultät für Maschinenbau und Schiffstechnik der Universität Rostock, 2015.

[65] R. Prabel and H. Aschemann. Sliding Mode Control for a System of two Coupled Hydraulic Cylinders. In *Proc. 20th Int. Conf. on Methods and Models in Automation and Robotics (MMAR)*, Międzyzdroje (Poland), 2015.

[66] R. Prabel and H. Aschemann. Nonlinear Adaptive Backstepping Control of Two Coupled Hydraulic Servo Cylinders. In *Proc. Am. Control Conf. (ACC)*, Portland (USA), 2014.

[67] R. Prabel and H. Aschemann. Comparison of Nonlinear Flatness-Based Control of two Coupled Hydraulic Servo Cylinders. In *Proc. 19th IFAC World Congress*, Capetown, (South Africa), 2014.

[68] J. Yao, Z. Jiao, and D. Ma. High dynamic adaptive robust control of load emulator with output feedback signal. *J. Franklin Inst.*, 351(8):4415 – 4433, 2014.

[69] P. Mercorelli. Robust feedback linearization using an adaptive PD regulator for a sensorless control of a throttle valve. *Mechatronics*, 19(8):1334 – 1345, 2009.

[70] M. Jelali and A. Kroll. *Hydraulic Servo-systems. Modelling, Identification and Control.* Springer-Verlag London, 2003.

[71] B. Haus, H. Aschemann, P. Mercorelli, and N. Werner. Nonlinear Modelling and Sliding Mode Control of a Piezo-Hydraulic Valve System. In *Proc. 21st Int. Conf. on Methods and Models in Automation and Robotics (MMAR)*, pages 442–447, 2016.

[72] P. Mercorelli. A two-stage sliding-mode high-gain observer to reduce uncertainties and disturbances effects for sensorless control in automotive applications. *IEEE Trans. Ind. Electron.*, 62(9):5929–5940, 2015.

[73] K. Demirbaş. Online state estimation for discrete nonlinear dynamic systems with nonlinear noise and interference. *J. Franklin Inst.*, 352(1):216 – 235, 2015.

[74] M. Hilairet, F. Auger, and E. Berthelot. Speed and rotor flux estimation of induction machines using a two-stage extended Kalman filter. *Automatica*, 45:1819–1827, 2009.

[75] N. Salvatore, A. Caponio, F. Neri, S. Stasi, and G. L. Cascella. Optimization of Delayed-State Kalman-Filter-Based Algorithm via Differential Evolution for Sensorless Control of Induction Motors. *IEEE Trans. Ind. Electron.*, 57(1):385–394, 2010.

[76] S. Bolognani, L. Tubiana, and M. Zigliotto. Extended Kalman Filter Tuning in Sensorless PMSM Sensorless Drive. *IEEE Trans. Ind. Appl.*, 39(6):1741–1747, 2003.

[77] S. Bolognani, M. Zigliotto, and M. Zordan. Extended-range PMSM sensorless speed drive based on stochastic filtering. *IEEE Trans. Power Electron.*, 16:110–117, 2001.

[78] S. Bolognani, R. Oboe, and M. Zigliotto. Sensorless full-digital PMSM drive with EKF estimation of speed and rotor position. *IEEE Trans. Ind. Electron.*, 46:1–8, 1999.

[79] P. Mercorelli. A Two-Stage Augmented Extended Kalman Filter as an Observer for Sensorless Valve Control in Camless Internal Combustion Engines. *IEEE Trans. Ind. Electron.*, 59(11):4236–4247, 2012.

[80] P. Mercorelli. A Hysteresis Hybrid Extended Kalman Filter as an Observer for Sensorless Valve Control in Camless Internal Combustion Engines. *IEEE Trans. Ind. Appl.*, 48(6):1940–1949, 2012.

[81] T. Tuovinen, M. Hinkkanen, L. Harnefors, and J. Luomi. Comparison of a Reduced-Order Observer and a Full-Order Observer for Sensorless Synchronous Motor Drives. *IEEE Trans. Ind. Appl.*, 48(6):1959–1967, 2012.

[82] M. Hilairet, F. Auger, and C. Darengosse. Two efficient Kalman filters for flux and velocity estimation of induction motors. In *Proc. 31st IEEE Ann. Power Electron. Spec. Conf.*, volume 2, pages 891–896, Galway, 2000.

[83] A. Akrad, M. Hilairet, and D. Diallo. A sensorless PMSM drive using a two stage extended Kalman estimator. In *Proc. 34th Ann. Conf. IEEE Ind. Electron. Soc. (IECON)*, pages 2776 – 2781, Orlando, 2008.

[84] B. Haus, P. Mercorelli, and N. Werner. Mirrored Piezo Servo Hydraulic Actuators for Use in Camless Combustion Engines and its Control with Mirrored Inputs and MPC. In *Proc. 14th Eur. Control Conf. (ECC)*, pages 1680–1686, 2015.

[85] Alberto Isidori. *Nonlinear Control Systems*. Springer London, London, 1995.

[86] T. Andersen, M. Hansen, H. Pedersen, and F. Conrad. Feedback linearisation applied on a hydraulic servo system. In *Proc. 6th JFPS Int. Symp. Fluid Power*, pages 167–172, Tsukuba, Japan, 2005.

[87] B. M. Åkesson, J. B. Jørgensen, N. K. Poulsen, and S. B. Jørgensen. A generalized autocovariance least-squares method for kalman filter tuning. *J. Process Control*, 18(7-8):769 – 779, 2008.

[88] B. J. Odelson, M. R. Rajamani, and J. B. Rawlings. A new autocovariance least-squares method for estimating noise covariances. *Automatica*, 42(2):303 – 308, 2006.

[89] P. Kachroo and M. Tomizuka. Chattering reduction and error convergence in the sliding-mode control of a class of nonlinear systems. *IEEE Trans. Autom. Control*, 41(7):1063–1068, 1996.

[90] G. Bartolini, A. Ferrara, and E. Usai. Chattering avoidance by second-order sliding mode control. *IEEE Trans. Autom. Control*, 43(2):241–246, 1998.

[91] H. Aschemann, B. Haus, and P. Mercorelli. Second-order sliding mode control with state and disturbance estimation for a permanent magnet linear motor. In *23rd Int. Conf. on Methods Models in Automation Robotics (MMAR)*, pages 345–350, 2018.

[92] C. Xia, G. Jiang, W. Chen, and T. Shi. Switching-gain adaptation current control for brushless DC motors. *IEEE Trans. Ind. Electron.*, 63(4):2044–2052, 2016.

[93] S. Roy, I. N. Kar, J. Lee, and M. Jin. Adaptive-robust time-delay control for a class of uncertain Euler-Lagrange systems. *IEEE Trans. Ind. Electron.*, 64(9):7109–7119, 2017.

[94] F. Plestan, Y. Shtessel, V. Brégeault, and A. Poznyak. New methodologies for adaptive sliding mode control. *Int. J. Control*, 83(9):1907–1919, 2010.

[95] J. Liu, H. Li, and Y. Deng. Torque ripple minimization of PMSM based on robust ILC via adaptive sliding mode control. *IEEE Trans. Power Electron.*, 33(4):3655–3671, 2018.

[96] H.-T. Moon, H.-S. Kim, and M.-J. Youn. A discrete-time predictive current control for PMSM. *IEEE Trans. Power Electron.*, 18(1):464–472, 2003.

[97] P. Mercorelli. A sufficient asymptotic stability condition in generalised model predictive control to avoid input saturation. In K. Ntalianis and A. Croitoru, editors, *Applied Physics, System Science and Computers II*. Springer International Publishing, 2018.

[98] P. Mercorelli. A motion-sensorless control for intake valves in combustion engines. *IEEE Trans. Ind. Electron.*, 64(4):3402–3412, 2017.

[99] C.-S. Hsieh. Robust two-stage Kalman filters for systems with unknown inputs. *IEEE Trans. Autom. Control*, 45(12):2374–2378, 2000.

[100] C.-S. Hsieh and F.-C. Chen. Optimal solution of the two-stage Kalman estimator. *IEEE Trans. Autom. Control*, 44(1):194–199, 1999.

[101] C.-S. Hsieh. General two-stage extended Kalman filters. *IEEE Trans. Autom. Control*, 48(2):289–293, 2003.

[102] D. Haessig and B. Friedland. Separate-bias estimation with reduced-order Kalman filters. *IEEE Trans. Autom. Control*, 43(7):983–987, 1998.

[103] L. Chen, P. Mercorelli, and S. Liu. A Kalman estimator for detecting repetitive disturbances. In *Proc. Am. Control Conf. (ACC)*, pages 1631–1636 vol. 3, 2005.

[104] B. Armstrong-Hélouvry, P. Dupont, and C. Canudas de Wit. A survey of models, analysis tools and compensation methods for the control of machines with friction. *Automatica*, 30(7):1083 – 1138, 1994.

[105] H. Aschemann, B. Haus, and P. Mercorelli. Sliding mode control and observer-based disturbance compensation for a permanent magnet linear motor. In *Proc. Am. Control Conf. (ACC)*, pages 4141–4146, 2018.

[106] B. Haus, J. H. Röhl, P. Mercorelli, and H. Aschemann. Model predictive control for switching gain adaptation in a sliding mode controller of a dc drive with nonlinear friction. In *Proc. 22nd Int. Conf. on System Theory, Control and Computing (ICSTCC)*, pages 765–770, 2018.

[107] J.-J. E. Slotine and W. Li. *Applied nonlinear control*. Pearson, Upper Saddle River, NJ, 1991.

[108] W.-H. Chen. Disturbance observer based control for nonlinear systems. *IEEE/ASME Trans. Mechatron.*, 9(4):706–710, 2004.

[109] K. Jezernik, J. Korelič, and R. Horvat. PMSM sliding mode FPGA-based control for torque ripple reduction. *IEEE Trans. Power Electron.*, 28(7):3549–3556, 2013.

[110] T. M. Jahns, G. B. Kliman, and T. W. Neumann. Interior permanent-magnet synchronous motors for adjustable-speed drives. *IEEE Trans. Ind. Appl.*, IA-22(4):738–747, 1986.

[111] M. Preindl and S. Bolognani. Model predictive direct torque control with finite control set for PMSM drive systems, part 1: Maximum torque per ampere operation. *IEEE Trans. Ind. Informat.*, 9(4):1912–1921, 2013.

[112] V. Smidl, S. Janous, L. Adam, and Z. Peroutka. Direct speed control of a PMSM drive using SDRE and convex constrained optimization. *IEEE Trans. Ind. Electron.*, 65(1):532–542, 2018.

[113] M. Preindl and S. Bolognani. Model predictive direct torque control with finite control set for PMSM drive systems, part 2: Field weakening operation. *IEEE Trans. Ind. Informat.*, 9(2):648–657, 2013.

[114] M. Preindl and S. Bolognani. Model predictive direct speed control with finite control set of PMSM drive systems. *IEEE Trans. Power Electron.*, 28(2):1007–1015, 2013.

[115] I.-C. Baik, K.-H. Kim, and M.-J. Youn. Robust nonlinear speed control of PM synchronous motor using boundary layer integral sliding mode control technique. *IEEE Trans. Control Syst. Technol.*, 8(1):47–54, 2000.

[116] P. Mercorelli. An adaptive and optimized switching observer for sensorless control of an electromagnetic valve actuator in camless internal combustion engines. *Asian J. Control*, 16(4):959–973, 2014.

[117] G. Rinaldi, P. P. Menon, C. Edwards, and A. Ferrara. Sliding mode based dynamic state estimation for synchronous generators in power systems. *IEEE Control Syst. Lett.*, 2(4):785–790, 2018.

[118] E. Ghahremani and I. Kamwa. Dynamic state estimation in power system by applying the extended Kalman filter with unknown inputs to phasor measurements. *IEEE Trans. Power Syst.*, 26(4):2556–2566, 2011.

[119] R. Ortega, L. Praly, A. Astolfi, J. Lee, and K. Nam. Estimation of rotor position and speed of permanent magnet synchronous motors with guaranteed stability. *IEEE Trans. Control Syst. Technol.*, 19(3):601–614, 2011.

[120] R. Ortega, N. Monshizadeh, P. Monshizadeh, D. Bazylev, and A. Pyrkin. Permanent magnet synchronous motors are globally asymptotically stabilizable with PI current control. *Automatica*, 98:296–301, 2018.

[121] H. Aschemann, B. Haus, and P. Mercorelli. Second-order SMC with disturbance compensation for robust tracking control in PMSM applications. In *Proc. 21st IFAC World Congress*, 2020.

[122] S. Bolognani, L. Ortombina, F. Tinazzi, and M. Zigliotto. Model sensitivity of fundamental-frequency based position estimators for sensorless PM and reluctance synchronous motor drives. *IEEE Trans. Ind. Electron.*, 65(1), 2018.

[123] M. De Soricellis, D. Da Rú, and S. Bolognani. A robust current control based on proportional-integral observers for permanent magnet synchronous machines. *IEEE Trans. Ind. Appl.*, PP(99):1–1, 2017.

[124] L. Vesely and D. Zamecnik. Extending mathematical model of permanent magnet synchronous motors in alpha-beta coordinate system for EKF. In *Proc. 12th Int. Symp. on Computational Intelligence and Informatics (CINTI)*, pages 457–460, 2011.

[125] S. Bolognani, L. Tubiana, and M. Zigliotto. EKF-based sensorless IPM synchronous motor drive for flux-weakening applications. *IEEE Trans. Ind. Appl.*, 39(3):768–775, 2003.

[126] F. Toso, D. Da Rù, P. Alotto, and S. Bolognani. A moving horizon estimator for the speed and rotor position of a sensorless PMSM drive. *IEEE Trans. Power Electron.*, 34(1):580–587, 2019.

[127] M. N. Uddin M. A. Rahman, D.M. Vilathgamuwa and T. King-Jet. Nonlinear control of interior permanent magnet synchronous motor. *IEEE Trans. Ind. Appl.*, 39(2):408–416, 2003.

[128] D. A. Khaburi and M. Shahnazari. Parameters identification of permanent magnet synchronous machine in vector control. In *Proc. 10th European Conference on Power Electronics and Applications (EPE)*, Toulouse, 2003.

[129] Stephen D. Umans. *Fitzgerald & Kingsley's Electric Machinery*. Electrical Engineering. McGraw-Hill Education, 7th edition edition, 2013.

[130] M. Koteich, G. Duc, A. Maloum, and G. Sandou. Observability of Sensorless Electric Drives. *arXiv e-prints*, page arXiv:1602.04468, 2018.

[131] P. Vaclavek, P. Blaha, and I. Herman. AC Drive Observability Analysis. *IEEE Trans. Ind. Electron.*, 60(8):3047–3059, 2013.

[132] M. Koteich, A. Maloum, G. Duc, and G. Sandou. Discussion on "AC Drive Observability Analysis". *IEEE Trans. Ind. Electron.*, 62(11):7224–7225, 2015.

[133] A. Bado, S. Bolognani, and M. Zigliotto. Effective estimation of speed and rotor position of a PM synchronous motor drive by a Kalman filtering technique. In *PESC '92 Record. 23rd Ann. IEEE Power Electro. Spec. Conf.*, pages 951–957, 1992.