DISSERTATION

Selected Topics of Adaptive and Iterative Learning Control

Ausgeführt zum Zwecke der Erlangung des akademischen Grades eines
Doktors der technischen Wissenschaften (Dr. techn.)

unter der Leitung von

Andreas KUGI
E376
Institut für Automatisierungs- und Regelungstechnik

eingereicht an der
Technischen Universität Wien
Fakultät für Elektrotechnik und Informationstechnik

von

Michael SCHWEGEL
Matrikelnummer: 00926940

Wien, März 2024

Studiendekan
Univ.-Prof. Dr.sc. Silvan Schmid

Betreuer
Univ.-Prof. Dr.techn. Andreas Kugi

Tag des Rigorosums
3. Mai 2024

Prüfungsvorsitzender
Univ.-Prof. Dr.sc. Silvan Schmid

Erster Gutachter
Univ.-Prof. Dr.-Ing. Boris Lohmann

Zweiter Gutachter
Associate Prof. Dr.-Ing. Wolfgang Kemmetmüller

Vorwort

Zuallererst möchte ich Andreas Kugi danken. Durch deinen rigorosen Zugang zur Technik, dein Vertrauen in meine Fähigkeiten und die Freiheit, die du mir zugestanden hast, hast du diese Arbeit überhaupt ermöglicht.
Dank gebührt auch der Robert Bosch GmbH, mit der vor allem durch Adrian Trachte und Michael Hilsch eine anregende Zusammenarbeit entstanden ist. Ebenso möchte ich meinem Betreuer Tobias Glück danken. Deinem Antrieb, deinen Ideen und einigen Lektionen ist es zu verdanken, dass diese Arbeit in praktischen und theoretischen Aspekten bereichert wurde.
Als zweiten starken Partner danke ich der KEBA Industrial Automation GmbH, und ganz besonders Christoph Mittermayer. Vielen Dank für die vielen Details und Gespräche über deine langjährige Erfahrung und deine spannenden Ideen im Bereich der Robotik. Es war mir stets eine Freude, mit dir zusammen arbeiten zu dürfen und so einen vertrauensvollen Partner zu haben.
Ein ganz spezieller Dank gebührt meinen Kolleg:innen am Institut für die farbenfrohe und anspornende Atmosphäre und die gemeinsamen Sport- und Freizeitaktivitäten. Insbesondere Thomas Weingartshofer und Minh Nhat Vu, eure Freundschaft und unsere Zusammenarbeit haben mich jeden Tag mindestens einmal zum Lachen gebracht. Weiters danke ich meinen Bürokolleg:innen Martin Meiringer, Christian Hartl-Nešić und Lucia Moser-Lauxmann für die motivierenden und inspirierenden und vor Allem auch für die spaßigen und phantasievollen Diskussionen und den gemeinsamen Arbeits- und Rückzugsort.
Allerliebsten Dank an meine Eltern Cornelia und Peter, sowie meinen Geschwistern Florian und Hanna für die fortwährende bedingungslose Unterstützung. Euer Rückhalt und die Zuversicht, die ihr mir gegeben habt, haben mich über viele Hindernisse hinweg begleitet und sind ein steter Begleiter auf meinen Wegen.
Aus tiefstem Herzen Danke ich dir, Maria, für deine unfassbar starke Liebe, die Geborgenheit, die du mir gibst und deine Hilfe in allen Lebenslagen. Nur durch dich habe ich es geschafft und ich möchte dir dafür Danken! Möge unser Magnet nie an Kraft verlieren.

Kurzzusammenfassung

Moderne Regelungssysteme nutzen das Modellwissen der ihnen zugrundeliegenden Prozesse um zunehmend komplexere Automatisierungsaufgaben zu lösen. Diese Komplexität und die unvermeidbaren Unsicherheiten erfordern jedoch eine Kombination von adaptiven und lernenden Methoden mit traditionellen modellbasierten Ansätzen, um die Leistung komplexer Regelungssysteme weiter zu steigern und weitere Funktionalitäten zu ermöglichen. In dieser Arbeit werden drei Hauptthemen betrachtet, die beispielhaft veranschaulichen, welche Vorteile sich aus der Kombination aus Domänenwissen mit echtzeitfähigen adaptiven und lernenden Verfahren zur Erreichung der gewünschten Regelgüte ergeben. Zunächst wird eine Klasse von Algorithmen zur Parameterschätzung analysiert, die in einem weiteren Schritt für den Entwurf eines adaptiven Regelkonzeptes herangezogen wird. In einem weiteren Kapitel wird durch die Kombination aus modellbasierter und iterativ lernender Regelung die absolute Genauigkeit eines Industrieroboters durch Kompensation komplexer dynamischer Effekte, die nur schwer oder gar nicht durch physikalisch-basierte mathematische Modelle erfasst werden können, deutlich erhöht.

In der Literatur finden sich zahlreiche Varianten der Parameterschätzung mit Hilfe der kleinsten Fehlerquadrate und (exponentiellem) Vergessen. Sie alle haben Vor- und Nachteile hinsichtlich Konvergenzrate, Robustheit gegenüber Messrauschen sowie des Parametrierungs- und Rechenaufwandes. In dieser Arbeit wird ein allgemeiner Algorithmus zur Parameterschätzung mit Vergessen vorgestellt, auf dessen Basis sich eine Vielzahl von Algorithmen, die aus der Literatur her bekannt sind, durch Angabe einer allgemeinen Gewichtungsmatrix als Spezialfälle herleiten lassen. Dieser Algorithmus dient auch als Grundlage für den Entwurf eines neuartigen Parameterschätzverfahrens mit Vergessen, der vordefinierte obere und untere Schranken für die Verstärkungsmatrix garantiert. Dieser Algorithmus kombiniert die Vorteile des exponentiellen Vergessens und Zurücksetzens mit der Eigenwertmodulation des selektiven Vergessens bei geringem Rechenaufwand.

Anhand dieser Ergebnisse wird ein neues adaptives Regelungskonzept zur Stromregelung von Elektromagnetakuatoren entwickelt. Die im Vergleich zur Literatur

zusätzlich eingeführte adaptive Vorsteuerung nutzt die geschätzten Systemparameter und die Modellstruktur, um die Regelgüte weiter zu erhöhen. Die Stabilität des geschlossenen Regelkreises bestehend aus der Strecke, der Parameteridentifikation und der adaptiven Regelung wird mathematisch rigoros nachgewiesen. Die vorgeschlagene Lösung unterscheidet sich von bestehenden Ansätzen in der Literatur durch die adaptive Vorsteuerung und die Art der Parameterschätzung. Das Regelungskonzept wird mit gleichbleibender Reglerparametrierung auf drei Elektromagnete unterschiedlicher Bauart mit deutlich verschiedenen Parametern angewendet. Die experimentellen Ergebnisse zeigen eine hohe Regelgüte und schnelle Parameterkonvergenz auch bei schlechten Anfangswerten und trotz geringer Systemanregung. Die experimentellen Ergebnisse werden auch mit zwei aus der Literatur bekannten Regelungsansätzen verglichen, die beide von dem vorgeschlagenen Konzept übertroffen werden.

Im Weiteren wird ein numerisch effizientes Regelungskonzept für die Verbesserung der absoluten Genauigkeit von Industrierobotern entwickelt und experimentell validiert. Ein modellbasierter Regler, der alle verfügbaren Systemparameter nutzt, wird mit einer iterativ lernenden Regelung (ILR) kombiniert. Diese ILR wird zur Kompensation der unbekannten Restfehlerdynamik, die durch elastisches Verhalten und Getriebeeffekte verursacht wird, eingesetzt. Der präsentierte Ansatz kombiniert mehrere Vorteile, darunter die Möglichkeit einer kontinuierlichen Durchführung von Iterationen, eine einfache Verallgemeinerung der gelernten Daten auf unterschiedliche Ausführungsgeschwindigkeiten des Roboters und das Lernen aus Teilversuchen. Die experimentelle Validierung an einem 6-achsigen Industrieroboter, bei dem die absolute Position des Endeffektors mit Hilfe eines Laser-Trackers gemessen wird, zeigt eine Verbesserung der absoluten Genauigkeit um 95 % nach bereits zwei Versuchen. Wenn der Lasertracker entfernt wird, kann die erreichte Genauigkeit durch die gelernte Vorsteuerung auch ohne weiteres Lernen von Versuch zu Versuch im Wesentlichen aufrechterhalten werden.

Abstract

Modern control systems leverage intricate mathematical models of the underlying process. However, the increasing complexity of automation tasks and inevitable uncertainties require a combination of learning strategies with classical model-based control to increase the performance of complex control systems further and enhance their capabilities. This work considers three main topics, exemplifying the benefits of combining process domain knowledge with online adaptation and learning methods to achieve the desired system performance. First, a class of parameter estimation algorithms is analyzed and used in the second step to design an adaptive control concept. Third, the accuracy of an industrial robot is significantly increased by compensating complex dynamical effects, which are difficult or impossible to capture using a first-principles mathematic model, through the combination of model-based and iterative learning control.

Many variants of the least-squares forgetting parameter estimation algorithms can be found in the literature. They all have advantages and disadvantages regarding adaptation rate, robustness against noise, parameter tuning, and computational effort. This work presents a general forgetting least-squares algorithm, where many algorithms known from the literature can be recovered by just specifying a general weighting matrix. This also serves as the basis for designing a novel least-squares forgetting algorithm that guarantees predefined upper and lower bounds on the gain matrix. This algorithm combines the benefits of exponential forgetting and resetting with the eigenvalue modulation of selective forgetting at low computational costs.

Based on these results, an adaptive two-degrees-of-freedom control algorithm for controlling the current of solenoids is derived. An additional adaptive feedforward controller takes advantage of the estimated plant parameters and the model structure to enhance the achieved tracking performance. The stability of the overall closed-loop system comprising the plant, the least-squares identification, and the adaptive control scheme is rigorously proven. The proposed solution differs from existing approaches by the adaptive feedforward term and how the parameter estimation is performed based on system domain knowledge. The control concept

is applied with the same controller parametrization to three solenoids with different designs and substantially differing parameters. The experimental results show high tracking performance and fast parameter convergence despite poor initial estimates and little excitation. The experimental results are also compared to two benchmark control designs known from the literature, which are outperformed by the proposed control concept.

Moreover, a numerically efficient flexible control scheme for improving the absolute accuracy of industrial robots is presented and experimentally validated. A model-based controller that leverages all typically available parameters of the robot is combined with an online path iterative learning controller (ILC). The ILC law compensates for the unknown residual error dynamics caused by elasticity and transmission effects. The proposed approach combines several benefits, including the possibility of a continuous execution of trials, a straightforward generalization of the learned data to different execution speeds, and learning from partial trials. After two trials, the experimental validation on a 6-DoF industrial robot with absolute position measurements by a laser tracker shows a 95 % improvement in absolute accuracy. When the laser tracker is removed, the achieved accuracy can be sustained by the learned feedforward controller even without trial-by-trial learning.

Contents

Chapter 1

Introduction

Control systems are crucial in many engineering disciplines, such as aerospace, robotics, and manufacturing. The main goal of control is to design and stabilize the dynamics of a closed-loop system, regulate the variables of interest to follow desired reference signals, suppress disturbances and parameter variations, and adapt behavior of the system to changing environmental conditions. Model-based control strategies have pushed the performance of many control applications and facilitated increasingly complex tasks by leveraging simulations and advanced (nonlinear) control design techniques. However, the ever-increasing complexity of the tasks, the inevitable model uncertainties, and the demanding performance requirements for a product portfolio with an extensive range of variants challenge a pure model-based control approach. These challenges can be overcome by combining model-based control with learning control strategies.

Adaptive control and iterative learning control are two strategies that enabled significant advancements in control systems in a various industrial applications. These strategies are based on the idea that the control system can learn from past data and adapt to changes in the environment and the system itself, allowing it to improve the control performance over time. This thesis focuses on selected online learning and adaptation strategies that efficiently use the data measured during operation to improve the achieved control performance.

Adaptive control is a control strategy that deals with systems whose model parameters are uncertain, time-varying, or even unknown. The main idea behind adaptive control is to estimate the system or control parameters online and use these estimated parameters to adjust the control law. This approach aims to achieve a high control performance despite the presence of uncertain or even unknown parameters in the system.

Iterative learning control, is a control strategy where repeated trials of a task are executed, and the gathered data is used to improve the control performance of the control system. Here, the main idea is to leverage the encountered control error of

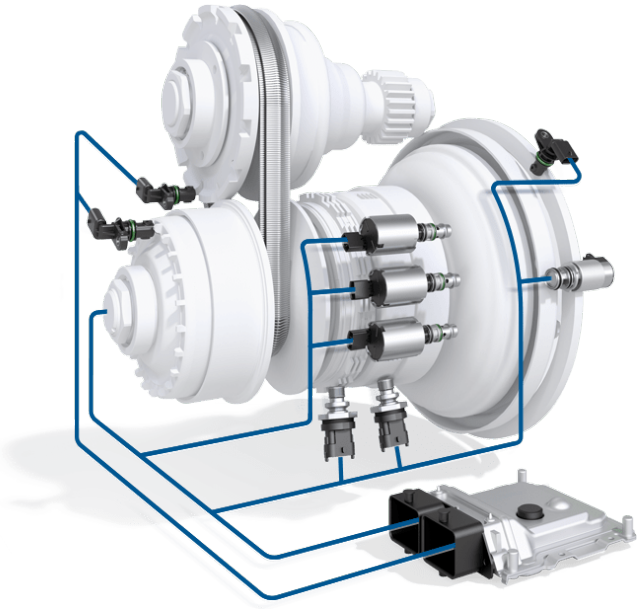

Figure 1.1: Continuously variable transmission with solenoid valves (three in the middle section, one to the right), sensor units, and transmission control unit (bottom). Picture courtesy Robert Bosch GmbH.

a trial to update the reference signal for the following trials. This approach allows the control system to increase the precision of the task trial-by-trial. Further, it can compensate for repetitive uncertainties and effects that are difficult to model and thus handle with model-based control strategies.

Throughout this thesis, the proposed control concepts will be applied to two examples of industrial applications to validate their efficacy and illustrate their advantages and limitations. Additionally, the thesis will provide a thorough mathematical analysis to support the proposed methods and results. The adaptive control methods are demonstrated and evaluated on solenoid actuators with unknown parameters. The presented iterative learning concept is demonstrated on an industrial robot performing high-accuracy repetitive motions.

Electromagnetic actuators, particularly solenoid valves, are critical components in hydraulic systems and are widespread across various fields, such as the automotive sector, excavators, heavy-duty vehicles, air- and spacecraft, industrial equipment, and factories. Figure 1.1 depicts a continuously variable transmission featuring four solenoid valves, sensors, and the control unit. Solenoid valves are commonly controlled by a dedicated electronic control unit (ECU). The controller parameters

Figure 1.2: Rendering of the considered industrial robot.

of the ECU have to be tuned by an expert to the nominal specification. An adaptive control approach can potentially supersede this individual tuning procedure, requiring only a reference specification and online learning from measurements. This approach also works for significant deviations from nominal product specifications due to production tolerances and uncertain environmental conditions.

An adaptive controller can achieve the desired control performance even for multiple solenoids of different designs. Furthermore, by considering the estimated parameters, useful information about the state of the solenoid can be obtained that can be used for predictive maintenance of critical components. This thesis discusses some results on the achievable performance of controlling a solenoid by adaptive control.

Industrial robots like the one shown in Figure 1.2 are essential for many manufacturing and production processes. While in the past these robots have been used mainly for pick and place or product handling tasks, industrial robots have evolved to perform complex tasks with increasingly demanding requirements. In state-of-the-art applications, industrial robots are employed to perform high-accuracy assembly, high-speed welding, glue or sealant dispensing, and many other challenging tasks. These applications impose increasingly stringent requirements on the execution and contraints robots under which robots operate.

Robots are often employed in tasks that are executed repeatedly. Thus, learning from previous task trials can compensate for misalignment, tolerances, and dynamical errors. This is the main idea of iterative learning control, which can be combined with well-known model-based control concepts to increase the achievable control accuracy. A practical challenge for many iterative learning control approaches is that data has to be processed for a complete trial in a batch-wise operation. In this thesis, the focus is on learning approaches that can be computed online and allow for learning from trials with variations of the execution speed.

1.1 Aim of this Thesis

This work aims to investigate two industrial applications of combining online learning control with model-based methods, develop novel approaches to improve the system performance, and experimentally validate the results. A novel adaptive control concept for solenoids is developed in the first two chapters. This concept consists of a computationally efficient algorithm for adaptation and control, which allows the control of different solenoids with identical tuning parameters. By ensuring that the desired control performance criteria are met for multiple solenoids of various designs, the deployment of solenoids is significantly facilitated.
First, the identification of parameters through least-squares (LS) estimation algorithms is considered. The main objective is to investigate the performance and trade-off involved with different forms of forgetting algorithms in the context of continuous-time indirect adaptive control. The analyzed properties of various methods known from the literature and the developed approach are validated in simulations.
Based on these parameter estimation strategies, an adaptive controller with a feedforward part that can achieve the same high control performance with identical tuning parameters for different solenoid types is designed. This approach is thoroughly analyzed, and the stability of the closed-loop system is proven. The modular indirect adaptive control concept investigated in this thesis is compatible with a class of recursive gradient and LS adaptation algorithms. Three solenoids of different types are used in an experimental setup to show the efficacy and flexibility of the proposed approach.
The last chapter investigates the repetitive motion of an industrial 6-axis robot. This investigation aims to improve the absolute accuracy of the robot traversing

a known path repeatedly. A learning-based approach that allows for a flexible handling of repetitive tasks is developed. Here, a data-driven learning control approach compensates for errors caused by uncertainties and unknown dynamics, such as joint elasticities, transmission and gearing errors, and friction effects. A particular emphasis lies in the easy implementation of this learning method. The algorithm facilitates learning from trials with variations in the execution speed. The results are validated on an industrial robot using laser-tracker measurements.

1.2 Main Contributions

In Chapter 2, least-squares (LS) identification methods known from the literature are generalized to a general forgetting LS algorithm. A proof of convergence and the relevant properties for adaptive control are derived, revealing easy-to-check conditions that guarantee these properties for a broad class of algorithms. This allows the design of problem-tailored algorithms simply by specifying a general forgetting matrix. A novel, efficient LS algorithm is proposed, combining the benefits of other existing approaches. This algorithm is specifically designed to guarantee a predefined upper and lower bound of the gain matrix, easy tuning, and computational efficiency.

Chapter 3 presents a flexible, high-performance control method with low computational costs for solenoid currents. The main contributions of this chapter are threefold: First, an indirect adaptive control strategy known from the literature is reformulated to account for practical problems and enhance parameter convergence. Second, the adaptive control strategy is extended by an adaptive feedforward controller, and the stability of the overall closed-loop system is proven. Third, an experimental validation underlining the practical value of the proposed control scheme is demonstrated by comparing the performance with two benchmark controllers from the literature.

In Chapter 4, a simple, efficient, and flexible learning control scheme for the absolute accuracy of an industrial robot is presented. The proposed iterative learning control (ILC) law can supplement the performance of existing control strategies with little requirements on the computational hardware and control structure. In contrast to many prior investigations, in this chapter, the focus lies on the absolute accuracy of the robot, which is measured using a laser tracker. Since the laser tracker may only be available for an initial learning phase, the efficacy of

the learned ILC signals as feedforward trajectories is investigated. The presented approach features a high flexibility and ease of practical application, as some of the classical requirements for the ILC are softened. All calculations can be done recursively, and a continuous learning phase is possible even with variations in the reference trajectory speed. The straightforward learning approach does not require any intervention between trials and achieves high accuracy for a complex trajectory on an industrial robot.

1.3 Outline

This thesis is organized into three main chapters.
In Chapter 2 the estimation of uncertain or unkown parameters using LS algorithms is analyzed, presenting the fundamental concepts, a comparison with methods from the literature, and the derviation of a novel algorithm. The presented results are supported by simulation, which highlights the properties of the presented methods.
In Chapter 3, the adaptive control problem is presented. First, the adaptation framework and the necessary filtering are discussed, and the two-degrees-of-freedom adaptive control law is derived. Next, the main stability theorems and their proofs are summarized, and two benchmark control approaches are introduced. Finally, the proposed control scheme is experimentally validated and compared with the benchmark controllers.
In Chapter 4, the ILC problem is formulated, a novel algorithm is developed and the experimental resutls are presented for a 6-axis industrial robot.
Finally, in Chapter 5, the main results of this work are concluded, and directions for future research are presented.

Chapter 2

Recursive Least-Squares Identification

Designing a system identification scheme implies a trade-off between adaptation rate and robustness against noise. While quick adaptation is required for fast convergence, robustness can only be achieved by moderate reactions to possibly faulty data. This chapter aims to give a general analysis of the adaptivity of recursive least-squares algorithms for continuous-time adaptive control. The analysis reveals the main aspects of algorithm design and tuning and facilitates the design of problem-tailored solutions. Furthermore, a novel algorithm is proposed, which combines the benefits of two approaches known from the literature. *Essential parts of this chapter are identical to the author's publication [1].*

2.1 Literature Review

It is well known that gradient algorithms converge proportional to the inverse of the time. Second-order least-squares algorithms achieve exponential convergence to the parameters under persistently exciting conditions, see, e.g., [2, 3, 4]. This benefit in convergence comes at the cost of computational complexity. In least-squares methods, the gradient and the Hessian matrix of the quadratic cost function must be calculated or approximated. The necessary forgetting mechanisms require additional computations and, thus, need to be computationally efficient.
The excellent robustness of pure least-squares algorithms results from the estimation gain converging to zero. This decay slows down the parameter convergence, which makes pure least-squares algorithms unsuitable for adaptive control. Exponential forgetting least-squares algorithms establish lower bounds for the adaptation gain, resulting in exponential parameter convergence under sufficient excitation. However, exponential forgetting may exhibit an unbounded growth of the gain matrix. Therefore, the estimator becomes extremely sensitive to noise and

susceptible to numerical and computational errors. In contrast, pure least-squares algorithms only guarantee parameter convergence. These arguments motivate the design of an algorithm with an upper and lower bounded gain matrix.

Albeit being a well-studied area, least-squares algorithms continue to be the subject of scientific research in many applications, such as parameter identification [5, 6, 7], set-membership filtering [8], as well as recent theoretical advances [9, 10]. The variety of identification methods in the literature suggests no general solution to the problem [11]. It is, therefore, important to incorporate as much problem domain-specific information as possible into the identification algorithm. Many recent examples exist of application-driven approaches [12, 13, 14, 15, 16, 17]. For instance, l_p-optimization using iterative reweighted least-squares concepts is widely used in signal processing and filtering [18]. The general forgetting convergence analysis provided in this thesis is essential for the design of tailored algorithms.

Various modifications of the exponential forgetting scheme were proposed in the literature to limit the gain matrix of a least-squares estimator with forgetting. Early methods restrict the growth of the gain matrix by their trace or norm, which leads to noncontinuous gain trajectories [19]. Other approaches provide a bounded estimator gain. However, it is not trivial to specify the bounds explicitly [20]. An algorithm with a time-varying forgetting factor that upholds a specified upper bound on the gain can also be found in the literature [21].

A Bayesian perspective on the identification problem reveals that a deterministic forgetting scheme corresponds to a stochastic model of the parameter noise [11, 22]. Identification algorithms derived from of a stochastic parameter model are typically difficult to tune. Furthermore, viewed as a Gaussian process, the least-squares identification approach can also be interpreted as a machine learning algorithm. In this context, a forgetting scheme can be associated with the kernel of a kernel learning machine [23, 10]. Similarly, in machine learning, small data batches have been successfully employed in different problems, and many recent approaches use so-called *mini-batch* updates [24, 25]. The learning process for a batch size of one can be seen as recursive gradient descent updates, which has sparked further interest in recursive optimization techniques. Besides these approaches, the forgetting algorithm can be interpreted as a regularization of the least-squares problem, see, e.g., [26].

In general, measurement information is not uniformly distributed over all parameters. This results in unlimited growth of some eigenvalues of the gain matrix

and can lead to parameter drift. The main idea of directional forgetting is that old information should only be discarded for parameter directions for which new data are available. The convergence of a subclass of directional forgetting algorithms was analyzed, and the drawbacks of such methods were discussed [27]. The exponential convergence of recursive least-squares with forgetting factor for multiple-output systems were recently analyzed in [28]. Another active topic is the robustness of least-squares algorithms, see, e.g., [29, 30].

2.2 Contribution

In this chapter, a continuous-time general forgetting least-squares (GLS) algorithm is derived. This algorithm emerges as the solution to an optimization problem in which the original least-squares cost functional is supplemented by an additional weighted parameter variation component. Exponential forgetting and many other algorithms from the literature are shown to be special cases of this generic type.
Additionally, conditions are derived for which a general least-squares forgetting algorithm has the convergence properties required for continuous-time adaptive control stability proofs. These conditions and proofs cover a wide range of continuous-time forgetting algorithms. Thus, they allow the designer to rely on ad-hoc modifications incorporating prior domain knowledge of the problem at hand while retaining a guarantee for the required convergence properties. This step is the continuous-time equivalent of a discrete-time analysis that was started by [31] in a very general context and applied by [32]. More recently, the relation of known least-squares approaches was investigated in [10, 33]. In [29], a discrete-time version of the work in this text was presented, and some results on the robustness of the algorithms were derived.
Furthermore, a matrix forgetting algorithm, which is well suited for continuous-time adaptive control, is presented. The algorithm allows direct specification of an upper and lower bound of the gain matrix and does not require an eigenvalue decomposition [31]. This bounded gain least-squares (BGLS) algorithm combines the benefits of earlier schemes while being computationally efficient.
The main contributions of this chapter are twofold: First, the formulation of the GLS, its convergence proof, and a proof of the relevant properties for adaptive control are derived. It allows the design of problem-tailored algorithms solely by specifying a general forgetting matrix. Second, the novel efficient BGLS algorithm,

which combines the benefits of other existing approaches, is proposed. This algorithm is specifically designed to guarantee a predefined upper and lower bound of the gain matrix, easy tuning, and computational efficiency.

2.3 Notation

The derivative of a variable $\boldsymbol{\theta}$ with respect to the time t is denoted by $\dot{\boldsymbol{\theta}}$. For the Euclidean norm of a vector and the induced 2-norm of a matrix the symbol $\|\cdot\|$ is used. Moreover, $|\cdot|$ denotes the absolute value of a scalar. Furthermore, $\mathbf{I}$ denotes the identity matrix of appropriate dimension. For symmetric matrices $\mathbf{A}$ and $\mathbf{B}$, the symbols $\mathbf{A} > \mathbf{B}$ and $(\mathbf{A} \geq \mathbf{B})$ are used if and only if $\mathbf{A} - \mathbf{B}$ is positive (semi-) definite. The space of bounded functions is denoted by $\mathcal{L}_\infty$, and the space of quadratically-integrable functions by $\mathcal{L}_2$.

2.4 Problem Statement

In this chapter, the linear regression equation

$$z = \boldsymbol{\varphi}^{\mathrm{T}} \boldsymbol{\theta}^* \tag{2.1}$$

is considered, where $z \in \mathbb{R}$ denotes the observed output, $\boldsymbol{\theta}^* \in \mathbb{R}^n$ is the true parameter vector, and $\boldsymbol{\varphi} \in \mathbb{R}^n$ is the regression vector of measurements. Many nonlinear systems can be formulated in this form, see, e.g., [34, p. 316]. The goal of this section is to calculate a continuous-time least-squares estimate $\boldsymbol{\theta}(t)$ of $\boldsymbol{\theta}^*$ at time t given the measured output $z(\tau)$ and regression vector $\boldsymbol{\varphi}(\tau)$ for $0 \leq \tau \leq t$.

Without loss of generality, the regression vector $\boldsymbol{\varphi}$ is assumed to be bounded by 1, i.e.,

$$\|\boldsymbol{\varphi}\| \leq 1 \,. \tag{2.2}$$

Practically, this is achieved by scaling the regression equation (2.1), e.g., by $m = 1 + \boldsymbol{\varphi}^{\mathrm{T}} \boldsymbol{\varphi}$. Details on the scaling procedure can be found in adaptive-control textbooks, e.g., [19].

2.5 General Forgetting Least-Squares Algorithm

In Section 2.5.1, the GLS algorithm is proposed, which is a continuous-time recursive least-squares algorithm with general forgetting. Additionally, an interpretation of the algorithm as the solution to a minimization problem is given and discussed. In Section 2.5.2, the convergence properties of this algorithm for a general forgetting scheme are analyzed. A large number of least-squares algorithms known from the literature can be recovered by this algorithm, which will be thoroughly discussed in Section 2.5.4.

2.5.1 Algorithm

The general forgetting least-squares algorithm has the form

$$\dot{\boldsymbol{\theta}}(t) = \mathbf{P}(t)\boldsymbol{\varphi}(t)\epsilon(t) \tag{2.3a}$$

$$\dot{\mathbf{P}}(t) = -\mathbf{P}(t)\left(\boldsymbol{\varphi}(t)\boldsymbol{\varphi}^{\mathrm{T}}(t) - \mathbf{F}(t)\right)\mathbf{P}(t)\,, \tag{2.3b}$$

with the parameter estimate vector $\boldsymbol{\theta}(t)$, a positive definite gain matrix $\mathbf{P}(t)$ and the initial conditions $\boldsymbol{\theta}(0) = \boldsymbol{\theta}_0$ and $\mathbf{P}(0) = \mathbf{P}_0 > \mathbf{0}$, respectively.[1] Here, the estimation error is

$$\epsilon(t) = z(t) - \boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\theta}(t)\,, \tag{2.4}$$

which represents the difference of the measured output $z(t)$ and the predicted output of (2.1). The matrix $\mathbf{F}(t)$ is a positive semidefinite forgetting matrix, which facilitates the design and interpretation of least-squares forgetting schemes. The significance of this matrix will be explained in more detail later in this section.

Remark 1. *It can be shown that the GLS algorithm* (2.3) *is the solution of the generalized continuous-time least-squares problem*

$$\boldsymbol{\theta}(t) = \arg\min_{\boldsymbol{\theta}(t)} J(\boldsymbol{\theta}(t))\,, \tag{2.5}$$

[1] It can be shown that $\mathbf{P}(t)$ is positive definite for all times $t > 0$ if $\mathbf{P}_0$ is chosen to be positive definite.

with the augmented cost functional

$$J(\boldsymbol{\theta}(t)) = \int_0^t \Big(z(\tau) - \boldsymbol{\varphi}^{\mathrm{T}}(\tau)\boldsymbol{\theta}(t)\Big)^2 + \big(\boldsymbol{\theta}(t) - \boldsymbol{\theta}_0\big)^{\mathrm{T}} \mathbf{P}_0^{-1} \big(\boldsymbol{\theta}(t) - \boldsymbol{\theta}_0\big) - \big(\boldsymbol{\theta}(t) - \boldsymbol{\theta}(\tau)\big)^{\mathrm{T}} \mathbf{F}(\tau) \big(\boldsymbol{\theta}(t) - \boldsymbol{\theta}(\tau)\big)\, \mathrm{d}\tau \,. \tag{2.6}$$

The forgetting matrix $\mathbf{F}(\tau) \geq \mathbf{0}$ *rewards the difference between the estimate* $\boldsymbol{\theta}(t)$ *at the time* t *and the estimates* $\boldsymbol{\theta}(\tau)$ *at past times* $\tau \leq t$*. Thus, using a forgetting matrix favours a variation of the estimates. In particular, for*

$$\mathbf{F}(\tau) = \beta\, \mathbf{P}^{-1}(\tau) \,, \tag{2.7}$$

the least-squares functional (2.6) *leads to the well-known continuous-time recursive least-squares algorithm with exponential forgetting, see also* (2.3)*.*

2.5.2 Convergence Analysis

This section summarizes the main convergence results of the GLS algorithm (2.3) with a general positive semidefinite forgetting matrix $\mathbf{F}(t)$. The analysis provides convergence properties for the parameter estimation equation (2.3a) with the estimation gain matrix (2.3b). Hereafter, the dependence of the variables on the time t is omitted to improve readability.

Theorem 1. *The GLS algorithm* (2.3) *guarantees that*

(i.) $\epsilon, \dot{\boldsymbol{\theta}}, \boldsymbol{\theta} \in \mathcal{L}_\infty$

(ii.) $\epsilon, \dot{\boldsymbol{\theta}} \in \mathcal{L}_2$,

if

- *the general forgetting matrix is positive semidefinite, i.e.,* $\mathbf{F} \geq \mathbf{0}$*,*
- *the gain matrix* $\mathbf{P}$ *is upper bounded, i.e.,* $\|\mathbf{P}\| \leq P_{\max}$*, and*
- *the ideal parameter vector* $\boldsymbol{\theta}^*$ *is constant.*

Proof. In the following, we consider the Lyapunov-like function

$$\mathcal{V} = \frac{1}{2} \tilde{\boldsymbol{\theta}}^{\mathrm{T}} \mathbf{R} \tilde{\boldsymbol{\theta}} \,, \tag{2.8}$$

with the definition of the inverse gain matrix

$$\mathbf{R} = \mathbf{P}^{-1} \tag{2.9}$$

and the parameter error [19] defined by $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta} - \boldsymbol{\theta}^*$. Using the identity

$$\frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{PR}) = \mathbf{0} = \dot{\mathbf{P}}\mathbf{R} + \mathbf{P}\dot{\mathbf{R}} \,, \tag{2.10}$$

it follows from (2.3) that

$$\dot{\mathbf{R}} = \boldsymbol{\varphi}\boldsymbol{\varphi}^{\mathrm{T}} - \mathbf{F} \,. \tag{2.11}$$

The time derivative of $\mathcal{V}$ using (2.11) reads as

$$\dot{\mathcal{V}} = \tilde{\boldsymbol{\theta}}^{\mathrm{T}}\mathbf{R}\dot{\tilde{\boldsymbol{\theta}}} + \frac{1}{2}\tilde{\boldsymbol{\theta}}^{\mathrm{T}}\left(\boldsymbol{\varphi}\boldsymbol{\varphi}^{\mathrm{T}} - \mathbf{F}\right)\tilde{\boldsymbol{\theta}} \,. \tag{2.12}$$

Because the true parameter vector $\boldsymbol{\theta}^*$ is assumed to be constant,

$$\dot{\mathcal{V}} = \tilde{\boldsymbol{\theta}}^{\mathrm{T}}\boldsymbol{\varphi}\epsilon + \frac{1}{2}\left((\tilde{\boldsymbol{\theta}}^{\mathrm{T}}\boldsymbol{\varphi})^2 - \tilde{\boldsymbol{\theta}}^{\mathrm{T}}\mathbf{F}\tilde{\boldsymbol{\theta}}\right) \tag{2.13}$$

can be found by using (2.3a). Equations (2.1) and (2.4) yield

$$\epsilon = -\tilde{\boldsymbol{\theta}}^{\mathrm{T}}\boldsymbol{\varphi} \tag{2.14}$$

and thus equation (2.13) simplifies to

$$\dot{\mathcal{V}} = \frac{1}{2}\left(-\tilde{\boldsymbol{\theta}}^{\mathrm{T}}\mathbf{F}\tilde{\boldsymbol{\theta}} - \epsilon^2\right) \,. \tag{2.15}$$

Since $\mathbf{F}$ is positive semidefinite, $\dot{\mathcal{V}} \leq -\epsilon^2/2 \leq 0$ for all times t. Hence, $\mathcal{V}(t) \leq \mathcal{V}(0)$ for $t > 0$. If $\mathcal{V}(0)$ is bounded, it follows that $\mathcal{V} \in \mathcal{L}_\infty$. Moreover,

$$\int_0^\infty \epsilon^2(\tau)\,\mathrm{d}\tau \leq -2\int_0^\infty \dot{\mathcal{V}}(\tau)\,\mathrm{d}\tau = 2\left(\mathcal{V}(0) - \mathcal{V}(\infty)\right) \tag{2.16}$$

proves that $\epsilon \in \mathcal{L}_2$ if the initial conditions of (2.3) are chosen such that $\tilde{\boldsymbol{\theta}}(0) \in \mathcal{L}_\infty$ and $\mathbf{P}(0) \in \mathcal{L}_\infty$. Because $\|\mathbf{P}\| \leq P_{\max}$ ensures that $\mathbf{0} < P_{\max}^{-1}\mathbf{I} \leq \mathbf{R}$, it follows from $\mathcal{V} \in \mathcal{L}_\infty$ and (2.8) that $\tilde{\boldsymbol{\theta}}, \boldsymbol{\theta} \in \mathcal{L}_\infty$. Furthermore, since the regression vector and the parameter error are bounded, cf. (2.2), according to (2.14), the estimation

error is bounded, i.e., $\epsilon \in \mathcal{L}_\infty$. From (2.3a) and the Cauchy-Schwarz inequality, it follows that

$$\|\dot{\boldsymbol{\theta}}\| \leq \|\mathbf{P}\| \, \|\boldsymbol{\varphi}\| \, |\epsilon| \leq P_{\max} \|\boldsymbol{\varphi}\| \, |\epsilon| \, . \tag{2.17}$$

Since $\boldsymbol{\varphi} \in \mathcal{L}_\infty$ according to (2.2) and $\epsilon \in \mathcal{L}_\infty \bigcap \mathcal{L}_2$, also $\dot{\boldsymbol{\theta}} \in \mathcal{L}_\infty \bigcap \mathcal{L}_2$ holds true. □

Remark 2. *Theorem 1 can be used for algorithms without upper bound $P_{\max}$ on the gain matrix $\mathbf{P}$ by replacing* (2.3b) *with*

$$\dot{\mathbf{P}} = \begin{cases} -\mathbf{P}\left(\boldsymbol{\varphi}\boldsymbol{\varphi}^{\mathrm{T}} - \mathbf{F}\right)\mathbf{P} & \text{if } \|\mathbf{P}\| \leq P_{\max} \\ -\mathbf{P}\boldsymbol{\varphi}\boldsymbol{\varphi}^{\mathrm{T}}\mathbf{P} & \text{otherwise.} \end{cases} \tag{2.18}$$

It is straightforward to show that, in this case, the derivative of the function $\mathcal{V}$ reads as

$$\dot{\mathcal{V}} = \begin{cases} -\frac{1}{2}\,\epsilon^2 - \frac{1}{2}\tilde{\boldsymbol{\theta}}^{\mathrm{T}}\mathbf{F}\tilde{\boldsymbol{\theta}} & \text{if } \|\mathbf{P}\| \leq P_{\max} \\ -\frac{1}{2}\,\epsilon^2 & \text{otherwise .} \end{cases} \tag{2.19}$$

This modification can be regarded as switching to pure least-squares, whenever the upper bound $P_{\max}$ of the gain matrix $\mathbf{P}$ is reached.

Remark 3. *Some special cases of Theorem 1 can be regarded as Ricatti equations, whose stability was shown in the literature, see, e.g., [35]. However, the generalized approach with $\mathbf{F} \geq \mathbf{0}$ contains various possible forgetting methods not covered by these proofs, e.g., approaches of the form* (2.18) *and forgetting matrices depending on powers of $\mathbf{P}$ greater than two.*

The properties guaranteed in Theorem 1 are those typically required by indirect adaptive control schemes, see also Chapter 3. Therefore, any algorithm that satisfies the requirements of Theorem 1 can be integrated into these schemes.
The well-known discrete-time LS algorithm is unbiased only under the assumption that the noise is independent of the measurement distribution, see, e.g., [36, p. 205]. However, this result holds only for well-conditioned problems, where the rank of the measurement vector equals the number of parameters.
In order to additionally ensure exponential convergence to the ideal parameters for recursive LS algorithms, further assumptions must be satisfied. The gain matrix $\mathbf{P}$ needs to be bounded from below and above. Moreover, the regression vector $\boldsymbol{\varphi}$ has to be persistently exciting, see Definition 1 in Appendix Section A.1, i.e., the

measurement signal must contain enough information to estimate all parameters. Intuitively, the signal has to excite all parameters within a finite timespan T_0 with nonzero bounded energy.

Theorem 2. *The GLS algorithm guarantees that* $\boldsymbol{\theta}$ *converges exponentially to* $\boldsymbol{\theta}^*$*, if the following conditions are satisfied:*

- *the general forgetting matrix is positive semidefinite, i.e.,* $\mathbf{F} \geq \mathbf{0}$*,*
- *the gain matrix* $\mathbf{P}$ *is lower and upper bounded, i.e.,* $0 < P_{\min} \leq \|\mathbf{P}\| \leq P_{\max}$*,*
- *the true parameter vector* $\boldsymbol{\theta}^*$ *is constant, and*
- $\boldsymbol{\varphi} \in \mathcal{L}_\infty$ *and* $\boldsymbol{\varphi}$ *is persistently exciting.*

The proof of Theorem 2 is given in Appendix A.2.

2.5.3 Desired Properties

It is desirable for continuous-time adaptive control applications [19] that the following properties of the GLS algorithm are satisfied:

1. bounded error, parameter estimates and gain matrix $\epsilon, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \mathbf{P} \in \mathcal{L}_\infty$
2. quadratic integrable estimation error and parameter variation $\epsilon, \dot{\boldsymbol{\theta}} \in \mathcal{L}_2$
3. exponential parameter convergence under sufficient excitation, i.e., $\boldsymbol{\theta}$ converges exponentially to $\boldsymbol{\theta}^*$ if $\boldsymbol{\varphi}$ is persistently exciting
4. upper bounded gain matrix, $\mathbf{P} \leq P_{\max}\mathbf{I}$ for a prespecified bound $P_{\max}$
5. lower bounded gain matrix, $\mathbf{P} \geq P_{\min}\mathbf{I}$ for a prespecified bound $P_{\min}$
6. exponential resetting, i.e., eigenvalues of unexcited directions tend to $P_{\max}$
7. small number of tuning parameters
8. low computational effort.

Different algorithms known from the literature cover all of these specifications, but to the best of the authors' knowledge no single algorithm features *all* these properties together. Specifically, the selective forgetting algorithm [31] guarantees many of these benefits, but at high computational costs. Therefore, in Section 2.6, a new algorithm called bounded gain least-squares algorithm (BGLS) is proposed by specifically designing the general forgetting matrix $\mathbf{F}$ in the GLS algorithm (2.3). This algorithm combines the benefits of exponential forgetting and resetting with smooth bounds on the gain matrix at low computational costs.

2.5.4 Relation of GLS to Existing LS Algorithms

This section shows that well-known least-squares forgetting algorithms are special cases of the GLS algorithm. To this end, only the general forgetting matrix $\mathbf{F}$ has to be specified accordingly. Moreover, the resulting properties of these algorithms will be discussed.

Modifications

A simple approach to satisfy the conditions of Theorem 1 is the modified LS method. Here, the forgetting matrix is chosen as

$$\mathbf{F} = \begin{cases} \beta\mathbf{R} & \text{if } \|\mathbf{P}\| \leq P_{\max} \\ \mathbf{0} & \text{otherwise} , \end{cases} \tag{2.20}$$

which upper bounds $\mathbf{P}$ by switching between a pure LS and an exponential forgetting LS algorithm. It is an example of an algorithm of the type (2.18) with exponential forgetting. However, forgetting is applied equally in all directions, and the gain matrix $\mathbf{P}$ does not change smoothly. Furthermore, there is no lower bound on the gain matrix $\mathbf{P}$ such that even good excitation may lead to slow adaptation.

Time-varying Forgetting Factor

Some schemes are based on a time-varying forgetting factor [37, 21]. There, the value of a scalar forgetting factor $\beta(t)$ can be varied such that the gain matrix $\mathbf{P}$ remains bounded. An example of such an algorithm is the Bounded-Gain Forgetting algorithm [21]. This exponential forgetting scheme uses the forgetting matrix

$$\mathbf{F} = \beta(t)\mathbf{R} \,. \tag{2.21}$$

The corresponding forgetting factor is given by

$$\beta(t) = \beta_0 \left(1 - \frac{\|\mathbf{P}(t)\|}{P_{\max}}\right) \,, \tag{2.22}$$

where $\beta_0 > 0$ and $P_{\max} > 0$ are the maximum forgetting factor and the desired upper bound on $\mathbf{P}$. This algorithm guarantees that $\|\mathbf{P}(t)\| \leq P_{\max}$ and exhibits exponential convergence when the regression vector $\boldsymbol{\varphi}$ is persistently exciting [21]. However, a lower bound on the gain matrix $\mathbf{P}$ cannot be specified.

Directional Forgetting

Intuitively, forgetting should be higher in directions where new information is available and lower in directions where new data is lacking. This concept is known as directional forgetting [27, 38, 37, 31, 20]. Early works on directional forgetting [38, 37] use a Bayesian viewpoint to suppress obsolete information. Although these algorithms bound the gain from above, slow adaptation may occur since they are not lower-bounded [27].

Information Matrix Decomposition

A directional algorithm can also be achieved by decomposing the information matrix [27]. There, the forgetting matrix is a projection of the information matrix onto the regression vector with exponential forgetting

$$\mathbf{F} = \beta\mathbf{R}\boldsymbol{\varphi}\boldsymbol{\varphi}^{\mathrm{T}}\mathbf{R}^{\mathrm{T}}/(\boldsymbol{\varphi}^{\mathrm{T}}\mathbf{R}\boldsymbol{\varphi}) \,. \tag{2.23}$$

This algorithm does not require substantial computational effort and was shown to have an upper and lower bounded gain matrix. However, these bounds cannot be specified [31] or calculated [20]. An inherent drawback of the algorithm is that although the information in some directions may be old, the gain in these directions does not increase.

Resetting

In contrast, increasing the gain when data is old is known as (exponential) resetting, which is a key property of the exponential forgetting and resetting algorithm [20]. This algorithm guarantees that the gain matrix $\mathbf{P}$ is bounded from both above and below. However, these bounds cannot be directly set. This algorithm has to be tuned by four interdependent design variables. The corresponding forgetting matrix reads as

$$\mathbf{F} = -\gamma\mathbf{I} + \beta\mathbf{R} + \delta\mathbf{R}^2 \,, \tag{2.24}$$

with the design constants γ, β, δ and an additional scaling $\alpha\boldsymbol{\varphi}\boldsymbol{\varphi}^{\mathrm{T}}$ of the measurements.

Selective Forgetting

The selective forgetting LS algorithm [31] is based on an eigenvalue decomposition of the gain matrix. With this approach, the eigenvalues can be explicitly bounded from above and below. The main drawback of this approach is the high computational effort associated with performing an eigenvalue decomposition at each time step.

Multiple Forgetting

Many recent papers consider parameters with different rates of change, such as [39]. For this type of problem, individual forgetting factors can be employed for each parameter. Further methods [10] can be represented by the GLS algorithm using

$$\mathbf{F} = \boldsymbol{\Lambda}\mathbf{R}\boldsymbol{\Lambda} \,, \tag{2.25}$$

where the exponential forgetting factor matrix $\boldsymbol{\Lambda} \geq \mathbf{0}$ contains forgetting factors for each parameter. This matrix can be interpreted as a kernel matrix in the context of machine learning [23].

2.6 Bounded Gain Least-Squares Algorithm

In this section, a novel least-squares forgetting algorithm is presented, which combines the benefits of exponential resetting with smooth bounds on the gain matrix.

Furthermore, the algorithm is easy to tune and allows for the explicit specification of the desired bounds on the gain matrix. The main idea is to combine the favorable properties of exponential forgetting and resetting with the eigenvalue modulation of selective forgetting. Since no eigenvalue decomposition is required for online computation, the calculation costs are kept low.

The desired properties 1–3 from Section 2.5.3 follow from Theorem 1 and Theorem 2 if the general forgetting matrix $\mathbf{F}$ is positive semidefinite and the gain matrix $\mathbf{P}$ is bounded from below and above. In order to ensure this, a quadratic polynomial in $\mathbf{R} = \mathbf{P}^{-1}$ is used as the forgetting matrix

$$\mathbf{F} = \alpha\mathbf{R}^2 + \beta\mathbf{R} - \gamma\mathbf{I} \,, \tag{2.26}$$

with positive parameters α, β, and γ, similar to the discrete-time algorithm [20]. A quadratic polynomial is sufficient to guarantee an upper and lower bound on the gain matrix, and requries the minimum number of tuning parameters. With a positive definite initial condition $\mathbf{P}(0) = \mathbf{P}_0$, the update law (2.3b) with (2.26) reads as

$$\dot{\mathbf{P}} = -\mathbf{P}\boldsymbol{\varphi}\boldsymbol{\varphi}^{\mathrm{T}}\mathbf{P} + \alpha\mathbf{I} + \beta\mathbf{P} - \gamma\mathbf{P}^2 \tag{2.27}$$

and $\mathbf{P}$ remains symmetric for all times $t \geq 0$.

Since $\mathbf{P}$ will be forced to be positive definite for all times $t \geq 0$, there is a well-defined eigenvalue decomposition

$$\mathbf{P}(t) = \sum_{i=1}^{n} \mu_i(t)\mathbf{v}_i(t)\mathbf{v}_i^{\mathrm{T}}(t) \tag{2.28}$$

for all times $t \geq 0$. Here, $\mu_i(t)$ is the i-th eigenvalue of the matrix $\mathbf{P}(t)$, and $\mathbf{v}_i(t)$ is its corresponding eigenvector of unit length, i.e., $\mathbf{v}_i^{\mathrm{T}}(t)\mathbf{v}_j(t) = \delta_{i,j}$ with $\delta_{i,j} = 1$ for $i = j$ and $\delta_{i,j} = 0$ otherwise, at the time t.

The positive parameters α, β, and γ will be chosen such that the eigenvalues $\mu_i(t)$, $i = 1, \ldots, n$ of $\mathbf{P}(t)$ satisfy the condition

$$\mu_{\min} \leq \inf_t \mu_i(t) \leq \mu_i(t) \leq \sup_t \mu_i(t) \leq \mu_{\max} \tag{2.29}$$

for all times $t \geq 0$ and chosen values $\mu_{\min} > 0$ and $\mu_{\max} > 0$. Clearly, if (2.29)

holds true, the quadratic form $\mathbf{x}^{\mathrm{T}}(t)\mathbf{P}(t)\mathbf{x}(t)$ is bounded from below and above

$$\mu_{\min}\|\mathbf{x}(t)\|^2 \leq \mathbf{x}^{\mathrm{T}}(t)\mathbf{P}(t)\mathbf{x}(t) \leq \mu_{\max}\|\mathbf{x}(t)\|^2 \tag{2.30}$$

for all times $t \geq 0$ and any vector $\mathbf{x}(t)$. Moreover, $\mathbf{P}(t)$ is bounded by the spectral matrix norm satisfying the conditions for Theorem 1 and Theorem 2. In particular, $\mu_{\min} \leq \|\mathbf{P}\|_2 \leq \mu_{\max}$ holds.

First, the condition for the upper bound $\mu_{\max}$ is formulated. For the orthonormal set of eigenvectors $\mathbf{v}_i$, the following relations hold

$$\mathbf{I} = \sum_{i=1}^{n} \mathbf{v}_i(t)\mathbf{v}_i^{\mathrm{T}}(t), \quad \mathbf{P}^2(t) = \sum_{i=1}^{n} \mu_i^2(t)\mathbf{v}_i(t)\mathbf{v}_i^{\mathrm{T}}(t) \; . \tag{2.31}$$

Substitution of (2.28) and (2.31) into (2.27) yields

$$\dot{\mathbf{P}} \leq \sum_{i=1}^{n} f(\mu_i)\mathbf{v}_i\mathbf{v}_i^{\mathrm{T}} \; , \tag{2.32}$$

with

$$f(\mu_i) = \alpha + \beta\mu_i - \gamma\mu_i^2 \; . \tag{2.33}$$

Since the forgetting matrix $\mathbf{F}(t)$ from (2.26) must be positive semidefinite, the relation $f(\mu_i) \geq 0$ must hold for all times $t \geq 0$. Thus, whenever an eigenvalue $\mu_i(t)$ tends to $\mu_{\max}$, it must not further increase. This condition is guaranteed if

$$\alpha + \beta\mu_{\max} - \gamma\mu_{\max}^2 = 0 \; . \tag{2.34}$$

Analogously, assumption (2.2) is exploited for the lower bound, which leads to $\boldsymbol{\varphi}\boldsymbol{\varphi}^{\mathrm{T}} \leq \mathbf{I}$, and thus (2.27) can be written in the form

$$\dot{\mathbf{P}} \geq -\mathbf{P}^2 + \mathbf{PFP} = \alpha\mathbf{I} + \beta\mathbf{P} - (1+\gamma)\mathbf{P}^2 \; . \tag{2.35}$$

Substitution of (2.28) and (2.31) into (2.35) yields

$$\dot{\mathbf{P}} \geq \sum_{i=1}^{n} \left(\alpha + \beta\mu_i - (1+\gamma)\mu_i^2\right) \mathbf{v}_i\mathbf{v}_i^{\mathrm{T}} \; . \tag{2.36}$$

If an eigenvalue $\mu_i(t)$ reaches the lower bound $\mu_{\min}$, it must not further decrease.

This can be guaranteed if the condition

$$\alpha + \beta\mu_{\min} - (1+\gamma)\mu_{\min}^2 = 0 \tag{2.37}$$

or equivalently $f(\mu_{\min}) = \mu_{\min}^2$ holds. Figure 2.1 shows the function (2.33) satisfying condition (2.34) and (2.37). The conditions for the upper bound (2.34) and the

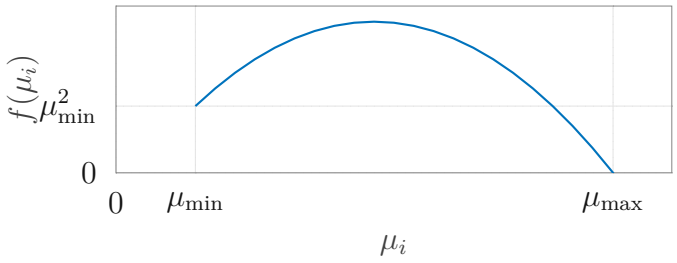

Figure 2.1: Forgetting function of the eigenvalues of the gain matrix $\mathbf{P}$.

lower bound (2.37) on the gain matrix $\mathbf{P}$ determine two of the three parameters α, β, and γ of (2.27). A third condition is introduced to specify the third parameter using the upper and lower bounds on the gain matrix $\mu_{\min}$ and $\mu_{\max}$. Here, we choose that exponential forgetting is the dominant forgetting term, i.e.,

$$\beta\mathbf{R} \geq \mathbf{F} \,. \tag{2.38}$$

Hence,

$$\alpha\frac{1}{\mu_i^2} - \gamma \leq 0 \,, \tag{2.39}$$

for all μ_i, and thus

$$\alpha - \gamma\mu_{\min}^2 = 0 \,. \tag{2.40}$$

The desired properties of the gain matrix $\mathbf{P}$ were formulated as algebraic equations of the parameters α, β, and γ. The condition

$$\mu_{\min}\mathbf{I} \leq \mathbf{P} \leq \mu_{\max}\mathbf{I} \tag{2.41}$$

and (2.38) are thus satisfied for a set of parameters α, β, and γ that fulfill (2.34),

(2.37), and (2.40). These equations have the unique solution

$$\alpha = \frac{\mu_{\max}\,\mu_{\min}^3}{\mu_{\max}^2 - \mu_{\min}^2} \tag{2.42a}$$

$$\beta = \mu_{\min} \tag{2.42b}$$

$$\gamma = \frac{\mu_{\min}\,\mu_{\max}}{\mu_{\max}^2 - \mu_{\min}^2}\,, \tag{2.42c}$$

with the desired upper bound $\mu_{\max}$ and lower bound $\mu_{\min}$ on the eigenvalues of the gain matrix $\mathbf{P}$, cf. properties 4) and 5) of Section 2.5.3, respectively.

Remark 4. *Note that enforcing* (2.38) *causes the lower bound on the gain* $\mu_{\min}$ *to be equivalent to the least-squares exponential forgetting factor* β*, see* (2.42b)*.*

Remark 5. *It can be shown that the cost functional (2.6) is convex and has a unique solution if (2.38) is satisfied.*

The bounded gain least-squares (BGLS) algorithm can be summarized as follows

$$\dot{\boldsymbol{\theta}} = \mathbf{P}\boldsymbol{\varphi}\epsilon \tag{2.43a}$$

$$\dot{\mathbf{P}} = -\mathbf{P}\boldsymbol{\varphi}\boldsymbol{\varphi}^{\mathrm{T}}\mathbf{P} + \alpha\mathbf{I} + \beta\mathbf{P} - \gamma\mathbf{P}^2\,, \tag{2.43b}$$

with appropriate initial conditions $\boldsymbol{\theta}(0) = \boldsymbol{\theta}_0$ and $\mathbf{P}(0) = \mathbf{P}_0 > \mathbf{0}$ and parameters α, β, and γ from (2.42).
By construction, this estimator features all the desired properties mentioned in Section 2.5.3. The computational effort is kept low since the matrix differential equation (2.43b) is symmetric and can be calculated efficiently and no eigenvalue decomposition is needed.

2.7 Simulation Results

In this section, a simple numerical demonstration example is presented to highlight the practical benefits of the theoretical properties of the BGLS algorithm presented in the previous section. Simulations with higher-order systems and different noise and input scenarios were investigated and similar results were attained. Furthermore, the tuning parameters of the algorithm are explained in more detail. In the following, the BGLS algorithm is compared to two classical LS approaches from

the literature. The simulation scenario includes output measurements corrupted by measurement noise, switching parameters, and periods of low excitation.
In this example, the linear time-invariant system

$$y = \frac{b}{s+a}u \tag{2.44}$$

is investigated, where u and y are the system input and output, respectively, and s denotes the Laplace variable. The derivative signal sy is realized by filtering the input and output of the system (2.44), denoted with a subscript f, using a first-order linear filter with the pole at $-2.5\,\mathrm{rad/s}$. The regression equation of the form (2.1) is obtained from (2.44) as

$$sy_f = \begin{bmatrix} u_f & -y_f \end{bmatrix} \begin{bmatrix} b \\ a \end{bmatrix} = \tilde{\boldsymbol{\varphi}}^{\mathrm{T}}\boldsymbol{\theta}^* \,, \tag{2.45}$$

and the equation is normalized by $m = 1 + \tilde{\boldsymbol{\varphi}}^{\mathrm{T}}\tilde{\boldsymbol{\varphi}}$. This yields the normalized quantities $\boldsymbol{\varphi} = \tilde{\boldsymbol{\varphi}}/m$ and $z = sy_f/m$. The ideal parameters are $a = 1.5$ and $b = 0.5$ and are switched to $a = 2$ and $b = 0.75$ after $5\,\mathrm{s}$. All simulations are initialized with $\boldsymbol{\theta}(0) = [1\ 1]^{\mathrm{T}}$ and $\mathbf{P}(0) = 100\,\mathbf{I}$.
Figure 2.2 shows the time evolution of the system output y and the input u for a series of input steps together with the corresponding filtered signals y_f and u_f, as well as $\dot{y}_f$. White Gaussian noise with a standard deviation of 0.03 was added to the output to simulate measurement noise.

Upper Bound on the Gain Matrix

The first simulation scenario shows the effects of the upper bound on the gain matrix compared to the classical exponential forgetting least-squares algorithm. Figure 2.3 shows the estimated parameters for the BGLS algorithm (2.43) with two parameter sets calculated using (2.42). Simulations 1 and 2 were carried out with the lower bound $\mu_{\mathrm{min,sim1}} = \mu_{\mathrm{min,sim2}} = 0.6$. The upper bound on the eigenvalues of the gain matrix $\mathbf{P}$ for simulation 1 and 2 is $\mu_{\mathrm{max,sim1}} = 100$ and $\mu_{\mathrm{max,sim2}} = 1000$, respectively. Simulation 3 was conducted using the classical exponential least-squares algorithm according to (2.3) with (2.7). The forgetting factor was chosen $\beta = 0.6$ due to the analogy (2.42b).
Note that in steady-state, the system is not sufficiently excited. In this case, only

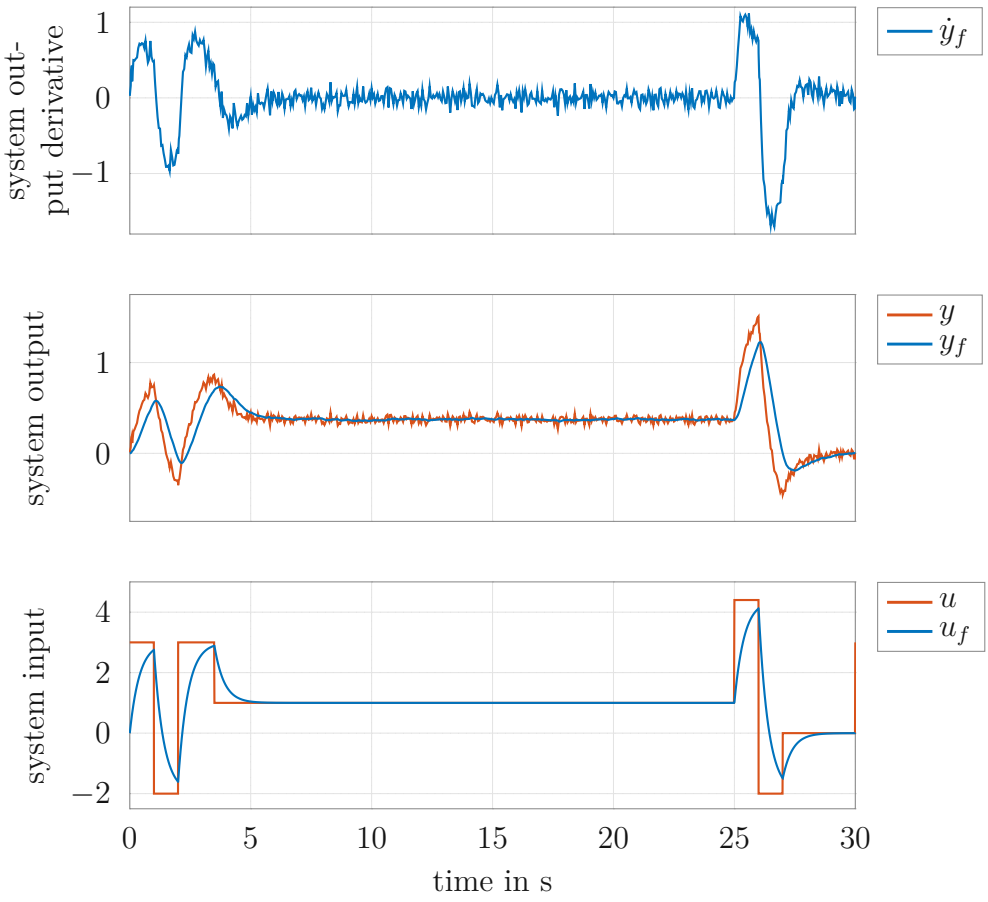

Figure 2.2: System output and input and their filtered time evolutions denoted by the index f.

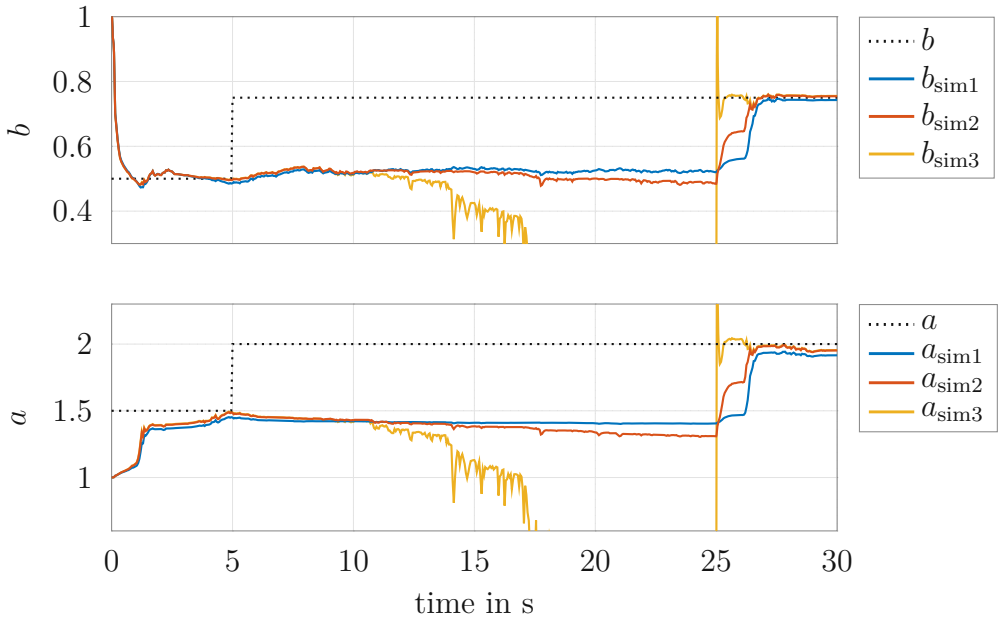

Figure 2.3: Estimated parameters and ideal values.

the ratio a/b is identifiable, while their individual values cannot be inferred from measurements. Therefore, the parameters drift due to the measurement noise, and the quotient a/b converges to the ideal value.

The estimates of all three simulations quickly converge to the ideal values for each input step. This can be explained by the exponential forgetting, which leads to large eigenvalues of the gain matrix, see Figure 2.4. From this figure, it can be seen that the BGLS algorithm ensures that the eigenvalues of the gain matrix $\mathbf{P}$ comply with their upper bounds. A smaller bound leads to slower adaptation. However, the estimates are more stable in phases of low excitation. The exponential forgetting LS algorithm does not provide bounds on the gain matrix, leading to exponential unbounded growth of the gain with insufficient excitation, also known as *blow-up* in the literature. The high gain leads to a rapid drift of the parameters during periods of low excitation.

For the BGLS algorithm, the parameters show only little drift even after a sudden change in the system parameters at 5 s without sufficient excitation. This behavior is a direct consequence of the upper bound on the gain matrix μ_{max}. Between 5 s and 25 s, the gain matrix increases in the direction of one eigenvector due to the lack of excitation. This effect leads to increased noise sensitivity and a

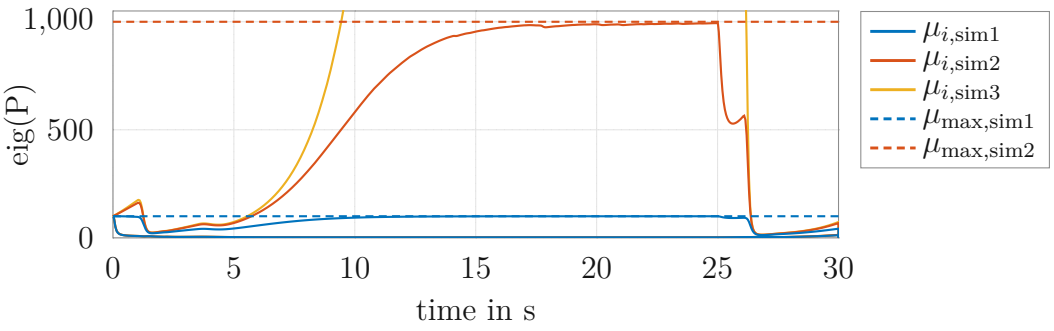

Figure 2.4: Eigenvalues of the gain matrix $\mathbf{P}$.

drift in the estimated parameters. The tuning of simulation 1 allows for less drift due to the smaller upper bound $\mu_{\max}$ on the gain matrix. Under sufficient excitation, the parameters quickly converge to the ideal values even after changes in the system parameters, e.g., at 5 s. The corresponding decrease in the gain matrix is called resetting. The parameters of simulation 2 and 3 quickly converge with new excitation due to the high gain at 25 s. Compared to simulation 2, simulation 1 shows less drift under insufficient excitation. Hence, the robustness of the BGLS algorithm to noise can be traded off with the convergence speed by selecting appropriate values for $\mu_{\min}$ and $\mu_{\max}$. Due to the equivalent exponential forgetting factors, the growth of the gain matrix under insufficient excitation is similar in all three simulations. Since the BGLS algorithm provides smooth bounds on the gain, the growth is successively reduced as the eigenvalues approach the upper bound.

Lower Bound on the Eigenvalues of the Gain Matrix

The lower bound guarantees a minimal convergence speed under persistence of excitation. The parameter $\mu_{\min}$ determines the exponential forgetting factor according to (2.42b). This value guarantees a minimum parameter tracking speed and prevents the eigenvalues of the gain matrix from converging to zero. It can be tuned analogously to conventional exponential forgetting LS approaches. Recall, that in this case $\lambda = 1 - \beta T_s$ holds, with λ the discrete-time forgetting factor and T_s the sample time.

The following second simulation scenario is based on the system signals depicted in Figure 2.2. Here, the BGLS algorithm is compared to the BGF algorithm. Simulation 4 and 5 are carried out with the upper bound $\mu_{\max,\text{sim4}} = \mu_{\max,\text{sim5}} =$

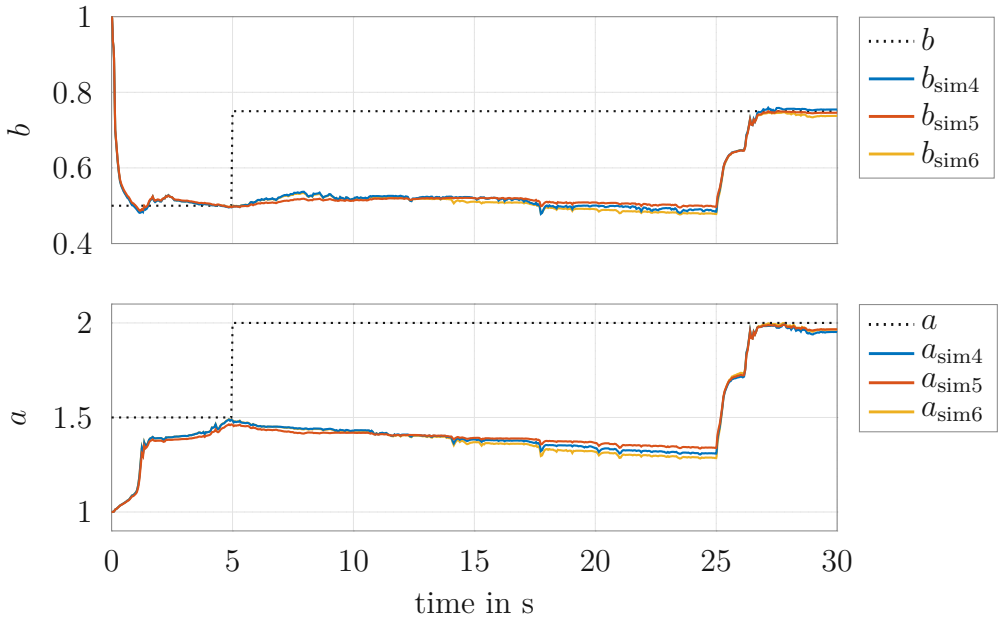

Figure 2.5: Estimated parameters and ideal values.

1000. The lower bounds on the eigenvalues of the gain matrix $\mathbf{P}$ for simulation 4 and 5 are $\mu_{\min,\text{sim4}} = 0.6$ and $\mu_{\min,\text{sim5}} = 0.3$, respectively. Simulation 6 was conducted using the BGF algorithm according to (2.3) with (2.21) and (2.22). The forgetting factor in (2.22) was chosen $\beta_0 = 0.6$, and the bound on the gain matrix is $P_{\max} = 1000$.

From Figure 2.5, it can be seen that due to the equal tuning, the algorithms lead to very similar parameter trajectories. However, from Figure 2.6, it is clear that this algorithm does not guarantee a specified lower bound on the gain matrix. Thus, no guarantees on the transient bahavior of the algorithm can be stated even under persistance of excitation. Due to the lower factor $\mu_{\min,\text{sim5}}$, the growth of the largest eigenvalue for this simulation is limited. These simulations show that the exponential forgetting factor should be tuned depending on the expected duration of phases with insufficient excitation.

This demonstration example shows that the BGLS algorithm has a fast convergence rate due to its exponential forgetting. After long periods of insufficient excitation, the algorithm quickly converges as soon as new data is available. The lower and upper bound on the gain matrix provide easy means to tune the trade-off between robustness and convergence for this estimation algorithm.

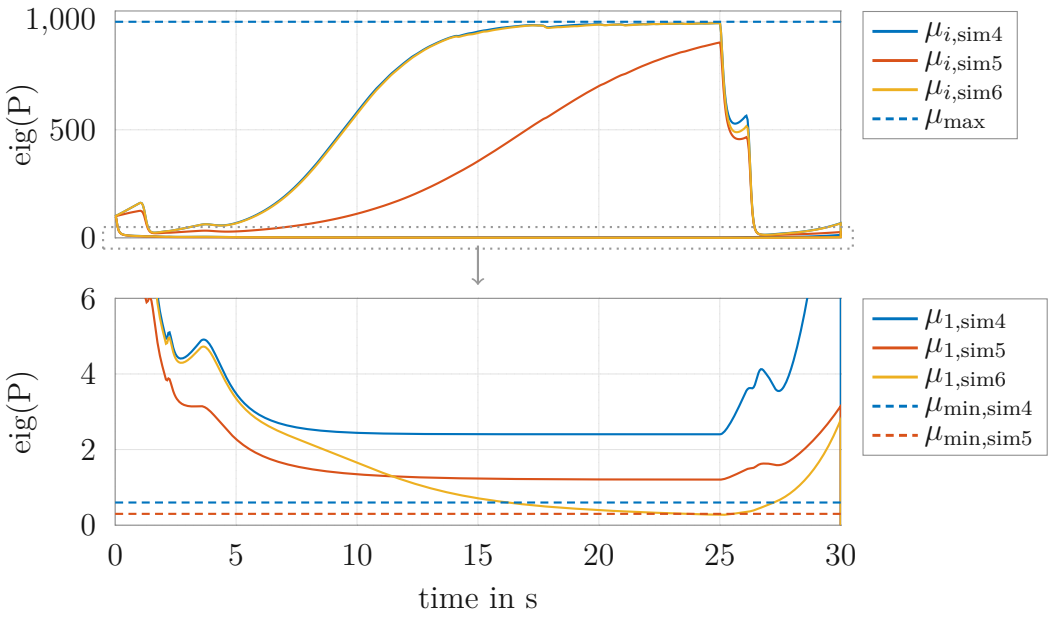

Figure 2.6: Eigenvalues of the gain matrix $\mathbf{P}$.

2.8 Conclusions

In this chapter, a general approach for continuous-time least-squares forgetting algorithms is presented. A large number of algorithms known from the literature can be recovered by this algorithm by choosing an appropriate positive semidefinite forgetting matrix. Important properties for convergence and stability necessary for continuous-time adaptive control were rigorously proven for this general algorithm. These properties are used in Chapter 3 to prove the stability of an adaptive control strategy with a LS parameter estimation algorithm.

A novel bounded gain least-squares (BGLS) forgetting algorithm was derived, based on the generalized LS framework. It combines the favorable properties of exponential forgetting and resetting with the eigenvalue modulation of selective forgetting at low computational costs. With this algorithm, it is possible to explicitly specify the desired lower and upper bounds of the gain matrix. In an illustrative simulation example, the exponential resetting and boundedness properties of the BGLS algorithm were demonstrated. Furthermore, the algorithm was compared to other algorithms from the literature.

Chapter 3

Adaptive Control of a Solenoid

The primary focus of adaptive control is to achieve a high tracking performance of the closed-loop system by adaptin the controller based on measurements of the system. Adaptive controllers can be categorized into direct methods, adapting the controller parameters directly, and indirect methods, where the uncertain parameters are estimated and based on an updated model, a new control law is parametrized. In this thesis, the concept of indirect adaptive control is investigated, since further model information can be included in the controller and due to the flexibility to use various adaptation algorithms, as indicated in the previous chapter.

Adaptive control can mitigate control performance degradation due to manufacturing tolerances. In contrast to robust control, adaptive control aims to achieve high control performance even with time-varying, uncertain, or unknown system parameters. Moreover, adaptive control allows the same controller to be employed within a class of structurally comparable systems. Due to the adaptive scheme, no manual adjustment of the controller parameters is necessary.

Parameter variations are expected for solenoids, widely used in pneumatic and hydraulic drive systems for utility vehicles such as excavators and cranes, in vehicle powertrains, and braking systems, e.g., [40, 41, 42, 43, 44]. Further fields of application include air- and spacecraft applications, industrial hydraulic actuators, and many applications of electro-mechanical actuators. In these applications, an adaptive controller can alleviate individual tuning procedures for different types of solenoids without compromising the tracking performance. In the following, an extension to existing adaptive control approaches is presented that achieves a high control performance for systems with uncertain or unknown parameters. Three solenoids of different types are used in an experimental setup to show the efficacy and flexibility for a maintenance-free application of the presented approach.

The main parts of this chapter have already been published in the author's publication [45].

3.1 Literature Review

3.1.1 A Brief History of Adaptive Control

Several control methods are at the boundaries of adaptive control, such as sliding-mode control, reinforcement learning, and iterative learning control. Compared to robust control techniques, adaptive control aims at optimal performance by explicitly considering the uncertainty in the adaptation, rather than tolerating a degradation in performance to maintain stability. In classical iterative learning control, the system trajectory is known a priori and repeatedly traversed, such that the acquired knowledge about the process can be iteratively improved. Adaptive control has a specifically rich intersection with reinforcement learning, which can be seen as an adaptive optimal control problem, e.g., [46].

Historically, the evolution of adaptive control was driven by different challenges encountered in practical applications. In the 1960s, adaptive mechanisms were investigated in search of a control algorithm for the wide range of conditions encountered by high-performance aircrafts. This challenge led to the model-reference adaptive control and what later became known as the *MIT-rule* in [47]. This concept is based on a gradient adaptation of a parametric controller, see, e.g., [48, 49].

Following these promising results, in the 1980s and 1990s, the field shifted towards providing stability proofs to ensure stable operation for time-varying parameter controllers. Several seminal results were achieved using Lyapunov's theory that inspired new algorithms and techniques. Many important survey papers, such as [50], were followed by classical textbooks, such as [48, 19, 4, 49, 51]. At the same time, the deterministic model-reference adaptive control and the stochastic self-tuning regulators evolved, and applications in various dynamic systems were addressed in the literature.

Another milestone in the evolution of adaptive control was improving the robustness of the proposed algorithms to perturbations and unmodelled dynamics. Seminal works developed methods based on feedback linearization, backstepping, and averaging, see, e.g., [21, 51]. A more detailed view of the evolution of the field of adaptive control can be found in [52].

It is worth noting that there has been a vigorous rise in activities in the area in the past five to ten years. Recent developments in adaptive control addressed the areas of transient adaptation performance, alleviating requirements on the excita-

tion of the system, and hybrid approaches, see, e.g., [52, 53, 54, 55, 1, 33]. The parameters of adaptive control algorithms are used to improve the performance of control systems with unknown parameters and to actively monitor critical system parameters. With the advance of distributed control systems and local maintenance requirements, adaptive control is essential in practical process control and automation.

3.1.2 Literature Review on Adaptive Control of a Solenoid

In the literature, different control approaches, such as classical proportional-integral (PI) control, proportional-integral-derivative (PID) control, internal model control (IMC), or sliding mode control (SMC) were investigated for the current control problem of solenoid valves. PID control and IMC are well-known standard control methods that result in equivalent output control structures and are thus comparable in terms of their robustness and tracking performance. The main idea of SMC is the robust control of a system using a discontinuous feedback law, see, e.g., [56, 57, 58, 59]. This approach can robustly handle parameter variations but typically requires tailored tuning for a given system to meet the high demands of the closed-loop performance. A high performance without individual tuning can be achieved by using adaptive control.
The adaptive output feedback control design problem for linear systems is well established and was solved in the late 90s, see, e.g., the textbooks [60, 19, 4, 3]. Therein, three main approaches are distinguished: The first is model reference adaptive control (MRAC), the adaptive version of the well-known model reference control (MRC) design. Here, the objective is to design a feedback controller to eliminate the output error between a reference model and the plant. The other two approaches refer to direct and indirect adaptive control [60]. In direct adaptive control, the control parameters are adjusted directly to improve the control performance. Direct adaptive control approaches have the drawback that the parameters typically used for adaptation can hardly be interpreted from a physical point of view. In contrast, in indirect adaptive control, the plant parameters are estimated online, and the control parameters are adjusted based on these estimates. These estimated plant parameters are not only instrumental for the parametrization of the controller but can also be employed for fault diagnosis and monitoring. New advances in hybrid and event-triggered adaptive control have targeted specific system classes, such as systems with exogenous inputs in [61], or provided non-

linear methods that are numerically more expensive compared to classical control schemes, including the adaptive control strategy proposed in this chapter, see, e.g., [15].

The primary purpose of adaptive control is to achieve high tracking performance despite unknown and changing system parameters. A well-known strategy to improve tracking performance is based on the idea of adding a feedforward path to an existing feedback control algorithm. Feedforward control is widely adopted, particularly in nonlinear adaptive control based on feedback linearization, e.g., [4, 62]. In these approaches, the parameter adaptation is mostly based on Lyapunov's theory, which guarantees convergence from a theoretical point of view but often results in an unsatisfactory slow convergence behavior in practical applications. Parameter adaptation based on least-squares methods ensures a balanced convergence rate across all parameters, see, e.g., [15]. These methods exhibit faster (second-order) convergence than typical Lyapunov-based approaches. Regularized recursive least-squares algorithms, see, e.g., [21, 1], mitigate the effect of noise on the parameter estimates by modifying the objective function and thus the gain matrix update to prevent the blow-up due to insufficient excitation [63, 64]. In recent works on robust least-squares system identification, non-asymptotic confidence intervals were computed [65, 66, 67, 68]. In addition, modifications of the least-squares algorithm known from the literature can account for problem-specific challenges, such as structural uncertainties or unknown constraints, see, e.g., [69, 70, 1].

3.2 Contribution

In this chapter an indirect adaptive two-degrees-of-freedom control scheme for solenoids without position measurements at low computational costs is presetend. It consists of an adaptive feedforward and a feedback path to fully utilize the estimated plant parameters. The plant parameters are estimated using a regularized least-squares adaptation law. Here, a reformulation and a modification of an adaptive control scheme are proposed to avoid the practical problems encountered when using the classical approach known from the literature, i.e., the indirect adaptive control from [19, 4]. These modifications ultimately lead to a significant improvement in the control performance while maintaining the flexibility and ease of tuning the original method.

The flexibility and performance of the control scheme are experimentally demon-

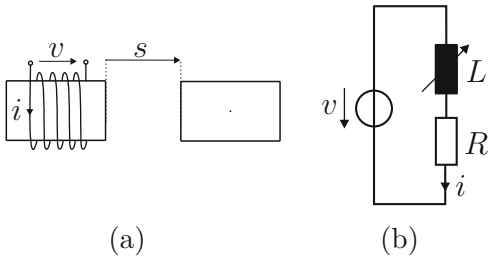

Figure 3.1: Mechanical and electrical schematics of a solenoid.

strated using three different solenoid types. Moreover, the proposed current control method is experimentally compared to other benchmark control methods from the literature. It is shown that a robust second-order sliding mode controller requires retuning to achieve adequate control performance across multiple solenoid types. Furthermore, a nonlinear model reference adaptive control method is a benchmark for the assessing the proposed control concept. The experimental results demonstrate that this benchmark controller is outperformed by the proposed control scheme in both parameter convergence and control performance.

In summary, the main contribution of this chapter is threefold: First, an indirect adaptive control strategy known from the literature is reformulated to account for practical problems and enhance parameter convergence. Second, the adaptive control strategy is extended by an adaptive feedforward controller. The stability of the overall closed-loop system is proven. Third, an experimental validation underlining the practical value of the proposed control scheme is provided by comparing the performance with two benchmark controllers from the literature.

3.3 Problem Statement

An adaptive current controller for solenoids is designed. A key concern is the achievable control performance without knowledge of the solenoid parameters. To reduce the costs, only the current i is measured, whereas the plunger position is not measured. Furthermore, since the nonlinear effects of a solenoid strongly depend on the respective design, these effects are not modeled.

Figure 3.1 shows the simplified mechanical and electrical schematics of a solenoid. The setup comprises the moving plunger and the magnetic core with the associated coil. Both the plunger and the magnetic core are made of highly-permeable

material with a relative permeability $\mu_r \gg 1$. The coil of the electromagnet is attached to the core and has N windings. Applying a voltage v to the terminals of the coil results in a current i, which in turn yields a magnetic field in the air gap g between the core and the position of the plunger. The coil voltage is typically provided by a high-side driver circuit. The generated pulse-width modulated (PWM) voltage signal switches between the supply voltage v_{bat} and 0 V. Mathematically, the pulse-width modulated (PWM) voltage reads as

$$v(t) = \begin{cases} v_{\text{bat}} & \text{for} \quad kT_{\text{pwm}} < t \leq (k+\delta)T_{\text{pwm}} \\ 0\,\text{V} & \text{for} \quad (k+\delta)T_{\text{pwm}} < t \leq (k+1)T_{\text{pwm}} \end{cases} \tag{3.1}$$

for $k = 1, 2, 3, \ldots$, where $0 \leq \delta \leq 1$ is the duty cycle, and T_{pwm} is the fixed modulation period.

For the magnetic flux linkage

$$\psi = L(g, i)i \,, \tag{3.2}$$

Faraday's law yields

$$\frac{\mathrm{d}\psi(g,i)}{\mathrm{d}t} = v - Ri, \tag{3.3}$$

with the inductance $L(g, i)$ and the electrical terminal resistance R. Substituting (3.2) in (3.3) results in the current dynamics

$$\underbrace{\left(L(g,i) + \frac{\partial L(g,i)}{\partial i} i\right)}_{\bar{L}} \frac{\mathrm{d}i}{\mathrm{d}t} = v - \underbrace{\left(R + \frac{\partial L(g,i)}{\partial g}\dot{g}\right)}_{\bar{R}} i \,. \tag{3.4}$$

In practice, $\bar{L}$ and $\bar{R}$ are unknown nonlinear functions of the current i and the air gap g, which depend on the specific solenoid design. Recall that the objective of interest is to design an adaptive control strategy for (3.4) that exhibits the same closed-loop performance independent of $\bar{L}$ and $\bar{R}$. Since we do not have any information about the exact characteristics of $\bar{L}$ and $\bar{R}$, we assume for the controller design that $\bar{L}$ and $\bar{R}$ are unknown but constant. Note that this is a common assumption in the context of adaptive control in the literature, see, e.g., [54, 53] and the references therein. Thus, in the following, we focus on the

simplified controller design model

$$\bar{L}\frac{\mathrm{d}y}{\mathrm{d}t} = u - \bar{R}y \,, \tag{3.5}$$

with the average input voltage $u(t) = v_{\mathrm{bat}}\delta(t)$, the unknown constant parameters $\bar{L}$ and $\bar{R}$, and the measured output current $y(t)$, which corresponds to the current $i(t)$ averaged over one modulation period.

Remark 6. *It is worth noting that an adaptive controller that ensures stability and the desired closed-loop performance for* (3.5) *does not guarantee that this also holds true for* (3.4)*. However, in this work, an adaptive two-degrees-of-freedom control concept is presented where the feedforward part strongly predominates over the feedback part of the control input signal; see also the experimental results in Section 3.7.3. This shows that the simplified model* (3.5) *together with the proposed parameter estimation approach is able to closely capture the dynamics of the original system* (3.4)*. It is well known from the literature, see, e.g., [71, 72], that parameter estimation schemes based on least-squares concepts with exponential forgetting exhibit a certain robustness to unmodeled nonlinear dynamics and time-varying parameters.*

3.4 Adaptive Control Concept

The proposed overall adaptive control structure is depicted in Figure 3.2. The input u and the output y are filtered by the linear low-pass filter Λ_a to generate the signals for the parameter adaptation. The reference signal r, which is assumed to be two-times continuously differentiable, specifies the desired time evolution of the output current y. The estimated parameters $\boldsymbol{\theta}$ are fed back to parametrize the feedforward and feedback controller, denoted by C_{ff} and C_{fb}, respectively. No disturbances affecting the plant are considered in the setting shown in Figure 3.2.

3.4.1 Adaptation Scheme

To compute the time derivative of the current $y = x$ and to mitigate high-frequency measurement noise and unmodeled effects, (3.5) is filtered by the linear low-pass

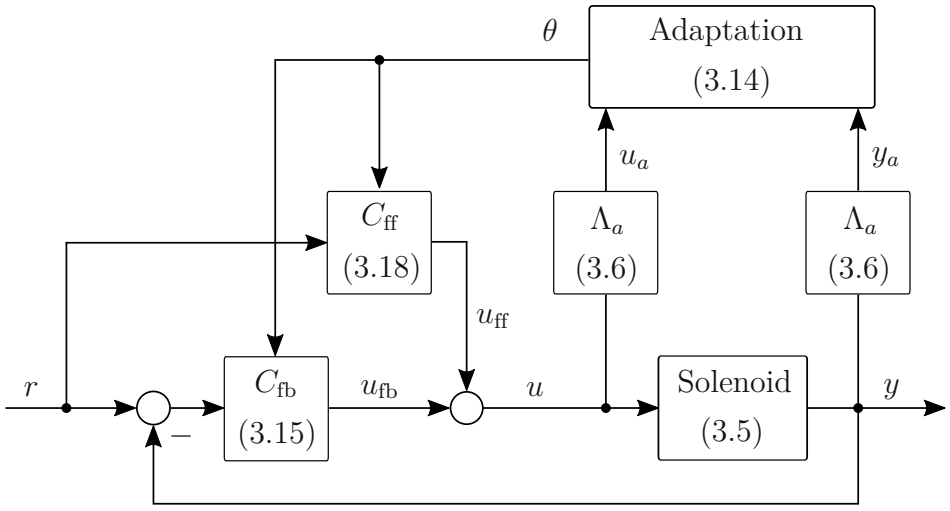

Figure 3.2: Overall adaptive control structure with the filter Λ_a, the adaptive feedfoward controller C_{ff}, and the feedback controller C_{fb}.

filter

$$\Lambda_a(s) = \frac{\lambda_a}{s + \lambda_a} \,, \tag{3.6}$$

with the Laplace variable s and the filter constant $\lambda_a > 0$. The input-output behavior of the plant is preserved by filtering both signals

$$u_a = \Lambda_a u \quad \text{and} \quad y_a = \Lambda_a y \,. \tag{3.7}$$

To apply a recursive least-squares algorithm, the model (3.5) is rewritten in the standard form with u_a as the scalar least-squares output, namely

$$u_a = \boldsymbol{\varphi}^{\mathrm{T}} \boldsymbol{\theta}^* = \begin{bmatrix} \frac{\mathrm{d}}{\mathrm{d}t} y_a & y_a \end{bmatrix} \begin{bmatrix} \bar{L} \\ \bar{R} \end{bmatrix} \,, \tag{3.8}$$

where $\boldsymbol{\theta}^* \in \mathbb{R}^2$ is the true parameter vector, and $\boldsymbol{\varphi} \in \mathbb{R}^2$ is the regression vector

$$\boldsymbol{\varphi} = \begin{bmatrix} \frac{\mathrm{d}}{\mathrm{d}t} y_a & y_a \end{bmatrix}^{\mathrm{T}} \,, \quad \boldsymbol{\theta}^* = \begin{bmatrix} \bar{L} & \bar{R} \end{bmatrix}^{\mathrm{T}} \,. \tag{3.9}$$

Remark 7. *In the classical formulation of adaptation algorithms, the highest derivative of the system is chosen as the adaptation output, i.e.,* $\boldsymbol{\varphi}^{\mathrm{T}} \boldsymbol{\theta}^* = \frac{\mathrm{d}}{\mathrm{d}t} i_a$*, see, e.g., [19, 4], which simplifies the mathematical treatment. In this case, the*

parameter vector reads as $\boldsymbol{\theta}^* = \left[1/\bar{L}, \bar{R}/\bar{L}\right]^{\mathrm{T}}$. *Practical experiments showed that the resulting coupling between the inductance and resistance parameters drastically degrades the estimation performance. Compared to other formulations, in* $\boldsymbol{\theta}^*$ *of* (3.9)*, the resistance and inductance can be estimated independently. In particular, since the resistance can be estimated in steady-state conditions, this formulation significantly improves the robustness to parameter drifts caused by low excitation. Additionally, projection methods can guarantee strict bounds on the individual parameters.*

Using the estimated parameter vector

$$\boldsymbol{\theta}^{\mathrm{T}} = \begin{bmatrix} \hat{L} & \hat{R} \end{bmatrix} , \tag{3.10}$$

the estimation error ϵ can be introduced, based on (3.8) and (3.10), as

$$\epsilon = \frac{\boldsymbol{\varphi}^{\mathrm{T}}\boldsymbol{\theta}^* - \boldsymbol{\varphi}^{\mathrm{T}}\boldsymbol{\theta}}{m^2} = \frac{u_a - \boldsymbol{\varphi}^{\mathrm{T}}\boldsymbol{\theta}}{m^2} , \tag{3.11}$$

with the normalization factor $m^2 = 1 + \boldsymbol{\varphi}^{\mathrm{T}}\boldsymbol{\varphi}$, see, e.g., [19]. Note that the normalization can be omitted if $\boldsymbol{\varphi} \in \mathcal{L}_\infty$, i.e., the vector function $\boldsymbol{\varphi}$ is essentially bounded. However, using the normalization factor m, the adaptation speed is normalized, which facilitates parameter tuning of the adaptation algorithm. In addition, projection allows to handle convex parameter constraints $\boldsymbol{\theta} \in \mathcal{S}$, to guarantee feasible limits of the parameter estimates, such as positive values for the inductance and resistance estimates. Given a convex set $\mathcal{S}$, the orthogonal projection of $\boldsymbol{\theta}$ on the set $\mathcal{S}$ is the solution of the optimization problem

$$\mathcal{P}_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \arg\min_{\mathbf{v}\in\mathcal{S}} \|\mathbf{v} - \boldsymbol{\theta}\|_2^2 . \tag{3.12}$$

One can define the projection of a vector $\mathbf{z}$ by, see [73],

$$\Pi_{\boldsymbol{\theta}}(\boldsymbol{\theta}, \mathbf{z}) = \lim_{\eta\to 0} \frac{\mathcal{P}_{\boldsymbol{\theta}}(\boldsymbol{\theta} + \eta\mathbf{z}) - \boldsymbol{\theta}}{\eta} , \tag{3.13}$$

with the convex set $\mathcal{S} = \{\boldsymbol{\theta} \in \mathbb{R}^2 | \mathbf{g}(\boldsymbol{\theta}) \leq \mathbf{0}\}$, its boundary $\delta\mathcal{S}$ and interior $\mathcal{S}^\circ$. Herein, the inequality $\mathbf{g}(\boldsymbol{\theta}) \leq \mathbf{0}$ describes the set $\mathcal{S}$ in the parameter space. To estimate the parameter vector $\boldsymbol{\theta}$, the so-called continuous-time constrained bounded-gain forgetting least-squares algorithm from [21] is augmented with the projection

algorithm described above. This algorithm was shown to have comparable performance to the BGLS algorithm from Section 2.6. It was employed here to show the practical application of an algorithm with the well-known exponential forgetting. Following similar steps to those in [1], this yields

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\theta} = \Pi_{\boldsymbol{\theta}}\left(\boldsymbol{\theta}, \mathbf{P}\boldsymbol{\varphi}\epsilon\right), \qquad \boldsymbol{\theta}(0) = \boldsymbol{\theta}_0\ , \tag{3.14a}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{P} = \Pi_{\mathbf{P}}\left(\boldsymbol{\theta}, \beta\mathbf{P} - \mathbf{P}\frac{\boldsymbol{\varphi}\boldsymbol{\varphi}^{\mathrm{T}}}{m^2}\mathbf{P}\right), \quad \mathbf{P}(0) = P_0\mathbf{I}\ , \tag{3.14b}$$

with

$$\Pi_{\mathbf{P}}(\boldsymbol{\theta}, \cdot) = \begin{cases} \cdot & \text{if } \boldsymbol{\theta} \in \mathcal{S}^{\circ} \text{or } \left(\text{if } \boldsymbol{\theta} \in \delta\mathcal{S} \text{ and } (\mathbf{P}\boldsymbol{\varphi}\epsilon)^{\mathrm{T}}\nabla\mathbf{g} \leq \mathbf{0}\right) \\ \mathbf{0} & \text{otherwise}\ . \end{cases} \tag{3.14c}$$

Herein, $\mathbf{P}$ is the positive definite gain matrix, $\boldsymbol{\theta}_0$ and $P_0\mathbf{I} > 0$ are the initial conditions, and $\mathbf{I}$ denotes the identity matrix. The (time-dependent) forgetting factor in (3.14b)

$$\beta = \beta_{\max}\left(1 - \frac{\|\mathbf{P}\|}{P_{\max}}\right)\ , \tag{3.14d}$$

with $P_{\max}$ being an arbitrary positive constant, guarantees an upper and lower bound on the gain matrix $\mathbf{P}$ and a maximum forgetting factor $\beta_{\max}$; see (2.22) and [21]. For a more detailed analysis of least-squares adaptation algorithms, see Chapter 2 and [1, 74]. A practical implementation is given in Appendix A.4. The upper bound on the norm of the gain matrix can be specified by $P_{\max} > 0$. The parameters $\beta_{\max}$, $P_{\max}$, and the filter constant λ_a in (3.6) allow for an independent tuning of the adaptation algorithm. Hence, strong filtering can be used to suppress noise and to filter unmodeled system dynamics. Analogous to a conventional discrete-time least-squares forgetting factor $\lambda \in (0, 1]$, see, e.g., [75], the continuous-time forgetting factor can be found by $\beta_{\max} = (1 - \lambda)/T_s$, with the sampling time T_s, cf. (A.40e). The maximum gain $P_{\max}$ allows limiting the gradient of the estimated parameters.

3.4.2 Feedback and Feedforward Control

Using the certainty equivalence principle, adaptive pole placement control, see, e.g., [19], allows to derive the adaptive PI-feedback controller

$$u_{\text{fb}} = \hat{k}_p e + \hat{k}_i x_c \tag{3.15a}$$

$$\dot{x}_c = e \,, \tag{3.15b}$$

with the control error

$$e = r - y \,, \tag{3.16}$$

and the known reference signal r. The adaptive feedback controller (3.15) constitutes a PI controller with time-varying proportional and integral gains parametrized by adaptive pole placement according to

$$\hat{k}_p = \hat{L}\alpha_1^* - \hat{R} \quad \text{and} \quad \hat{k}_i = \hat{L}\alpha_0^* \,, \tag{3.17}$$

with constant coefficients $\alpha_1^* > 0$ and $\alpha_0^* > 0$. To enhance the tracking performance, the adaptive feedforward controller

$$u_{\text{ff}} = \hat{L}\dot{r} + \hat{R}r \tag{3.18}$$

is introduced. Finally, the adaptive two-degrees-of-freedom control input is given by

$$u = u_{\text{ff}} + u_{\text{fb}} \,. \tag{3.19}$$

Applying (3.19), with (3.15)-(3.18), to (3.5) and assuming that the certainty equivalence holds, i.e., the estimated parameters $\hat{L}$ and $\hat{R}$ correspond to their real values $\bar{L}$ and $\bar{R}$, the closed-loop error system

$$\ddot{e} + \alpha_1^* \dot{e} + \alpha_0^* e = 0 \tag{3.20}$$

is obtained. Clearly, with the constants α_0^* and α_1^*, the closed-loop poles of the error dynamics (3.20) can be chosen to achieve an exponentially stable behavior with a desired rate of decay.

The feedforward and feedback control (3.19) with (3.15)-(3.18) is combined with the parameter adaptation algorithm (3.14) to form the overall adaptive control scheme of Figure 3.2.

3.5 Stability Proof in a Nutshell

In this section, the main points of the stability proof of the overall closed-loop system comprising adaptation, controller, and plant are outlined. The only assumptions made are that the ideal parameter vector $\boldsymbol{\theta}^*$ is constant and that the reference signal r is sufficiently smooth, i.e., $r, \dot{r}, \ddot{r} \in \mathcal{L}_\infty$. Under these assumptions, Theorem 3 can be stated, which guarantees bounds on certain signals of the adaptation algorithm.

Theorem 3. *The least-squares algorithm* (3.14) *guarantees that*

(i.) $\epsilon, \dot{\boldsymbol{\theta}}, \boldsymbol{\theta}, \epsilon m, \mathbf{P} \in \mathcal{L}_\infty$

(ii.) $\epsilon, \dot{\boldsymbol{\theta}}, \epsilon m \in \mathcal{L}_2$

(iii.) $\mathbf{g}(\boldsymbol{\theta}) \leq \mathbf{0}$,

with $\mathcal{L}_2$ *being the space of quadratically integrable functions and* $\mathcal{L}_\infty$ *the space of essentially bounded functions.*

Proof. The proof of Theorem 3 is similar to what is shown in [19] and follows by analyzing the function $V = (\boldsymbol{\theta} - \boldsymbol{\theta}^*)^{\mathrm{T}} \mathbf{P}^{-1} (\boldsymbol{\theta} - \boldsymbol{\theta}^*)$. □

Finally, Theorem 4 establishes the asymptotic stability of the overall adaptive control scheme of Figure 3.2.

Theorem 4. *For the parameter estimation algorithm presented in* (3.14)*, all signals in the closed-loop adaptive two-degrees-of-freedom control system* (3.14)-(3.19) *are uniformly bounded, and the control error* e *converges asymptotically to zero.*

Proof. The proof of Theorem 4 is performed in 4 steps:

1. The estimation error and control law are expressed as a linear time-varying (LTV) system.
2. The exponential stability of the LTV system is shown.
3. The boundedness of all signals in the closed-loop system is proven by using the Bellman-Gronwall lemma.
4. The control error convergence is proven using Barbalat's lemma.

More details of the proof of Theorem 4 are given in Appendix A.3. □

Remark 8. *Assuming persistence of excitation of the regression vector $\boldsymbol{\varphi}$, the adaptation algorithm converges exponentially to the ideal parameter vector, see Theorem 2 and [1]. However, for the convergence of the control error e neither persistence of excitation nor convergence of the parameters to the ideal parameter vector is necessary, as stated in Theorem 4.*

3.6 Benchmark Approaches from the Literature

In the following sections, two benchmark control approaches from the literature are presented, and their performance is compared with the adaptive two-degrees-of-freedom control algorithm presented in this chapter. First, in Section 3.6.1, a second-order sliding mode controller is given as an example of a robust control method commonly employed in solenoid control. Second, a model reference adaptive controller serving as a benchmark for an adaptive control method is discussed in Section 3.6.2. In industrial applications, further measures are taken to avoid practical problems like parameter drift under steady-state conditions, e.g., dead zone, dynamic normalization, or anti-windup, see, e.g., [19, chap. 8] for more details. For a fair and meaningful comparison, we refrain from implementing such measures because they can be used independently of the respective control and adaptation methods.

3.6.1 Second-Order Sliding Mode Controller

A second-order sliding mode controller with dynamic pole placement is proposed in [56]. The control input

$$u = \alpha_1 \sqrt{|\sigma(t)|}\, \operatorname{sign}(\sigma(t)) + \alpha_0 \int_0^t \sqrt[3]{|\sigma(\tau)|}\, \operatorname{sign}\big(\sigma(\tau)\big)\, \mathrm{d}\tau \,, \tag{3.21a}$$

with the constant tuning parameters $\alpha_0 > 0$ and $\alpha_1 > 0$ and the control error $e(t) = r(t) - y(t)$ is used to stabilize the sliding surface

$$\sigma\,(e) = \left(\frac{\mathrm{d}}{\mathrm{d}t} + \lambda_0 - \lambda_1\,|e|\right) e \,, \tag{3.21b}$$

with the constant tuning parameters $\lambda_1 > 0$ and $\lambda_0 > 0$. The bounds $|e| < e_{\max}$ and $\lambda_0 > \lambda_1 e_{\max}$ guarantee a stable closed-loop system.

3.6.2 Model Reference Adaptive Controller

As a benchmark for a well-known adaptive controller, the nonlinear model reference adaptive control scheme from [4], see also [48], is applied to (3.5), which yields

$$\dot{\boldsymbol{\theta}} = -\lambda \begin{bmatrix} y \\ \hat{R}y + \hat{L}(\dot{r} + K_p e) \end{bmatrix} e \tag{3.22a}$$

$$u = \hat{R}y + \hat{L}(\dot{r} + K_p e) \,, \tag{3.22b}$$

with the control error $e = r - y$, the parameter estimate vector (see Remark 7)

$$\boldsymbol{\theta}^{\mathrm{T}} = \begin{bmatrix} 1/\hat{L} & \hat{R}/\hat{L} \end{bmatrix} \,, \tag{3.23}$$

and the constant tuning coefficients $K_p > 0$ and $\lambda > 0$. The control law (3.22b) consists of a feedforward part using the time derivative of the reference signal $\dot{r}$, a static compensation of the estimated voltage caused by the resistance of the solenoid, and a proportional control term. As stated in the introduction, the control law is typically augmented by an adaptation algorithm to guarantee a decreasing Lyapunov function. Here, the commonly used quadratic functions lead to a gradient-type adaptation law. Note, however, that in this case, the adaptation (3.22a) is driven by the control error e rather than the estimation error ϵ.

3.7 Experimental Validation

In this section, experimental results of the benchmark control approaches from Section 3.6 are presented and compared with the adaptive control scheme proposed in Section 3.4. For this purpose, three different solenoids, henceforth referred to as solenoid A, B, and C, are used for the experiments, see Figure 3.3. The nominal current of solenoid A is denoted by i_A.

The three solenoids were taken from different fields of application and feature different mechanical and electromagnetic designs. In particular, solenoid A is used in a pressure control valve, solenoid B is part of a pilot valve of a hydraulic two-stage valve, and solenoid C is employed in an automatic transmission gear.

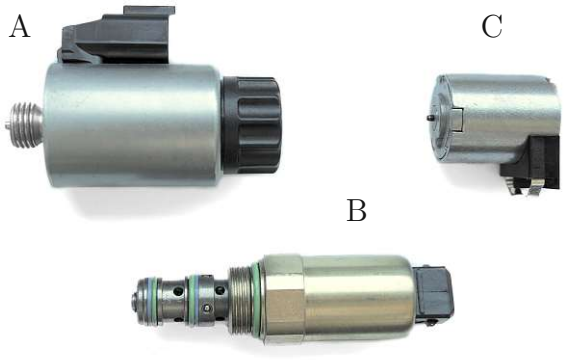

Figure 3.3: Photographs of the solenoids used for the experimental validation.

	Solenoid A	Solenoid B	Solenoid C
$\bar{R}$	R_A	$0.47R_A$	$0.82R_A$
$\bar{L}$	L_A	$0.15L_A$	$0.23L_A$

Table 3.1: Nominal parameters of the solenoids of Figure 3.3.

Hence, there are significant differences in their nominal resistance and inductance parameters. Their nominal parameter values are given in Table 3.1.
All experiments were conducted on a dSpace MicroLab Box at a sampling time of $T_s = 1\,\text{ms}$ and a modulation period of $T_{\text{pwm}} = 50\,\mu\text{s}$. The current is sampled at a rate of $10\,\mu\text{s}$ and averaged over 100 measurements in order to mitigate the effects of the current ripple caused by switching the transistor. The battery voltage v_{bat} is used with a calibrated power electronics circuit to generate the PWM voltages across the solenoid terminals.

3.7.1 Sliding Mode Control Experiments

In this section, experimental results of the sliding mode control law from [56], as outlined in Section 3.6.1, are presented as a baseline for comparing the proposed method with a common approach in solenoid control. Figure 3.4 shows the experimental results achieved by the control input (3.21a) applied to the solenoids A and B. The tuning parameters are listed in Table 3.2 for both cases. The peaks in the current error at 3.1 s and 5.7 s in Figure 3.4 result from the lack of a feedforward part in this control approach. This leads to a significant delay between the reference and the controlled current, which causes large control errors. However,

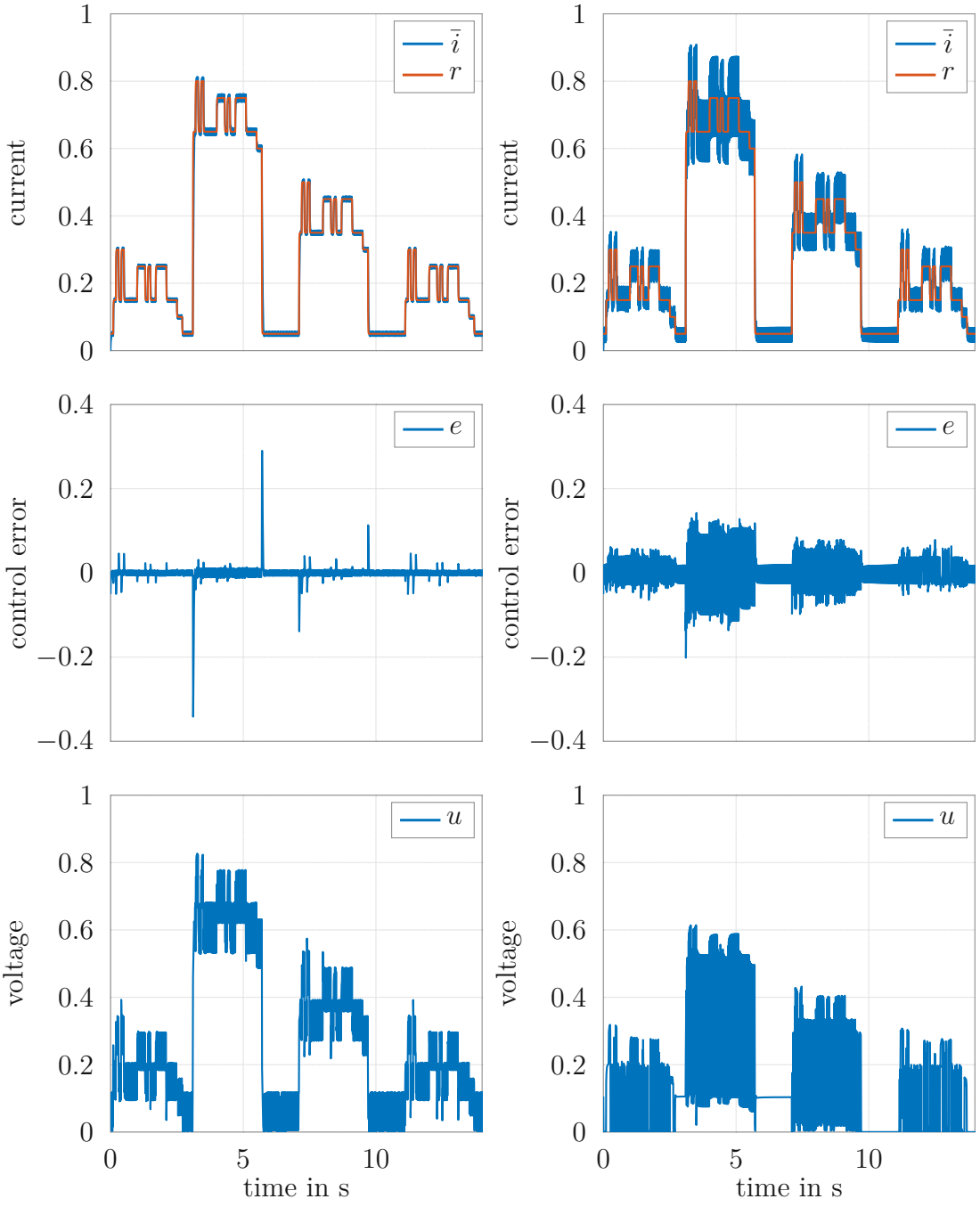

Figure 3.4: Experimental results of the sliding mode controller for solenoid A (left side) and for solenoid B (right side). The values are normalized to i_A and v_{bat}.

Control		Sliding surface	
α_1	0.35	λ_1	20
α_0	0.04	λ_0	100

Table 3.2: Parameters used for the sliding mode controller experiments.

the general performance of the well-tuned sliding mode controller for solenoid A is good. In contrast, the right plots of Figure 3.4 show the experimental results for the same sliding mode controller applied to solenoid B. Even though the sliding mode controller is a robust control approach, the control performance is severely degraded by the poor tuning for this solenoid. In particular, the smaller inductance results in overshoots and persistent oscillations of the current. Furthermore, the nonlinearity of the solenoid inductance leads to a larger control error at higher current levels. It becomes clear from Figure 3.4 that the sliding mode controller provides good results when properly tuned, but the performance may significantly degrade if retuning is not possible.

3.7.2 Model Reference Adaptive Control Experiments

In this section, experimental results of the nonlinear model reference adaptive control scheme from [4], as outlined in Section 3.6.2, are presented. The left of Figure 3.5 shows the solenoid current and the control error of the algorithm from (3.22a) and (3.22b) applied to solenoid A. The tuning parameters used in the experiment can be found in Table 3.3. The nominal parameters of the solenoid are given in Table 3.1. The reference trajectory was selected to show the performance of the algorithm for rapid setpoint changes and for periods with insufficient excitation. During these periods at about 6 s and 10 s, the reference signal is constant, hence, the inductance and the resistance cannot be identified simultaneously. Additionally, the inductance varies significantly with the different current levels of the reference signal. This current- and position-dependence of the inductance is an unmodeled nonlinear effect. The current trajectory in Figure 3.5 clearly shows that the adaptation algorithm cannot estimate the inductance and the resistance of the solenoid to achieve a satisfactory tracking performance. During periods of low excitation, the gradient-based adaptation law only converges slowly. Hence, in steady-state, the control error is slowly reduced, but the reference is not reached even after 1 s. The control error shows a large mean error with peaks over $0.2 i_A$.

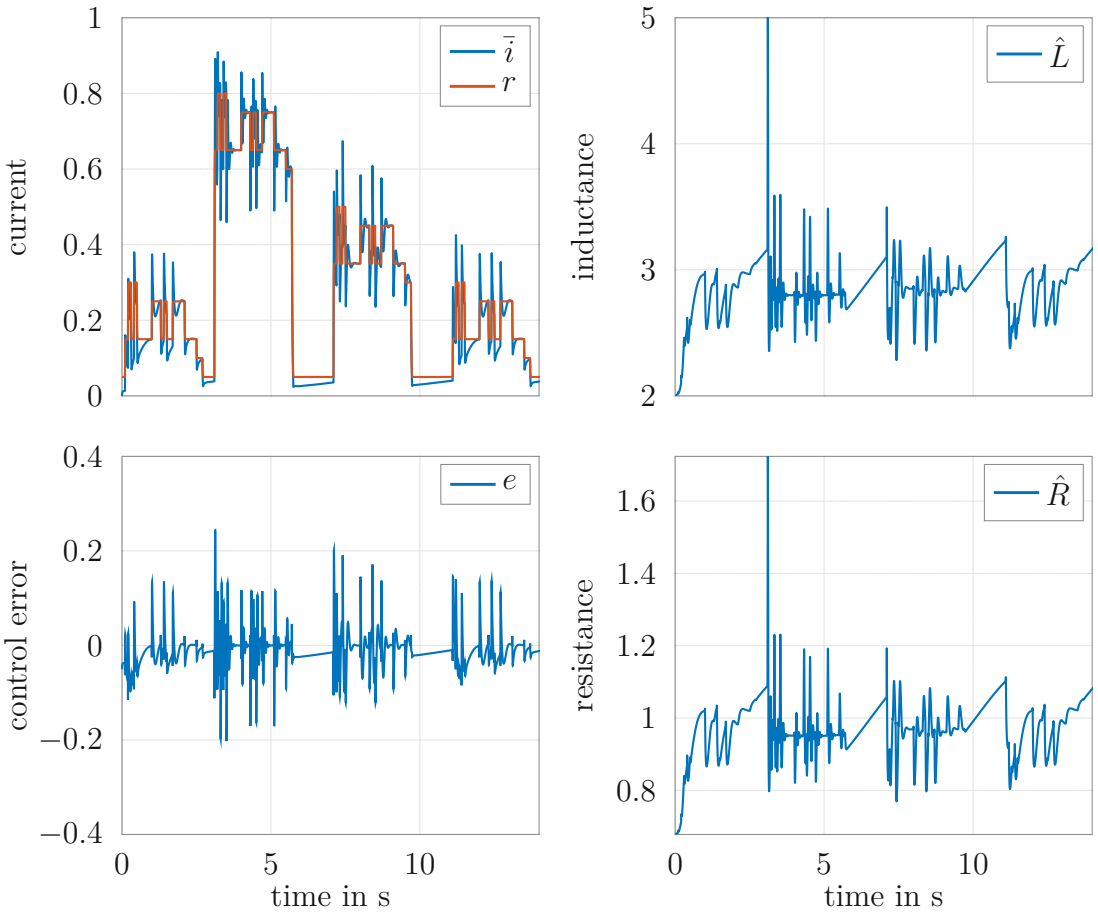

Figure 3.5: Solenoid current, control error and estimated parameters of the nonlinear model reference adaptive control scheme for solenoid A. The values are normalized to i_A, L_A, and R_A.

Control & adaptation		Initial conditions	
K_p	10	$\hat{R}_0$	$0.68R_A$
λ	20	$\hat{L}_0$	$2.0L_A$

Table 3.3: Parameters used for the nonlinear model reference adaptive control experiment.

Furthermore, the repeating reference signal at 11 s is not improved compared to the tracking performance with the initial parameters at 0.1 s. Both cases show a peak error of $0.15 i_A$.
The estimated resistance and inductance parameters of this experiment are shown on the right in Figure 3.5. Note that according to (3.22a), the parameter vector is updated proportionally to the control error. Hence, large control errors are necessary for the parameters to converge, which makes this approach sensitive to model uncertainties such as the nonlinear inductance effects. Furthermore, the parameter update has a constant gain λ. These two properties lead to a fluctuating update of the estimated parameters and rapid changes, whenever a large control error occurs. The estimated inductance values and the lack of dynamic feedback lead to high current overshoots.
As discussed in Section 3.4.1, projection bounds cannot be formulated tightly for the coupled parameter vector (3.23), which leads to estimates exceeding the desired bounds $L_{\max}$ and $R_{\max}$. At 3 s, the resistance exceeds the desired bound of $R_{\max} = 1.36 R_A$. During periods of low excitation at 6 s and 10 s, both parameters are used by the algorithm to counteract the steady-state error. However, only one independent parameter can be identified during this time, i.e., there is no persistence of excitation. Hence, the parameters drift on a one-dimensional subspace of the parameter space. This drift is caused by the loss of observability of the parameters, and methods such as the dead zone have been proposed to mitigate the drift. However, it will be shown that the drift is much slower for the proposed method. Note that the estimated parameters strongly depend on the control error and exhibit a similar trajectory.

3.7.3 Proposed Indirect Adaptive Control Scheme

In this section, the proposed indirect adaptive two-degrees-of-freedom control strategy is experimentally validated for all three investigated solenoids. To this end, the controller is initialized with the same parameters for all three solenoids depicted in Figure 3.3.
The constrained forgetting least-squares adaptation algorithm in (3.14) was discretized following [76, 77, 3], as detailed in Appendix A.4. The control parameters and initial values can be found in Table 3.4. The controller parameters α_0^* and α_1^* were chosen for a time constant of 10 ms and a damping ratio of 0.5 for the closed-loop error system (3.20). The initial parameters R_0 and L_0 were set to typical

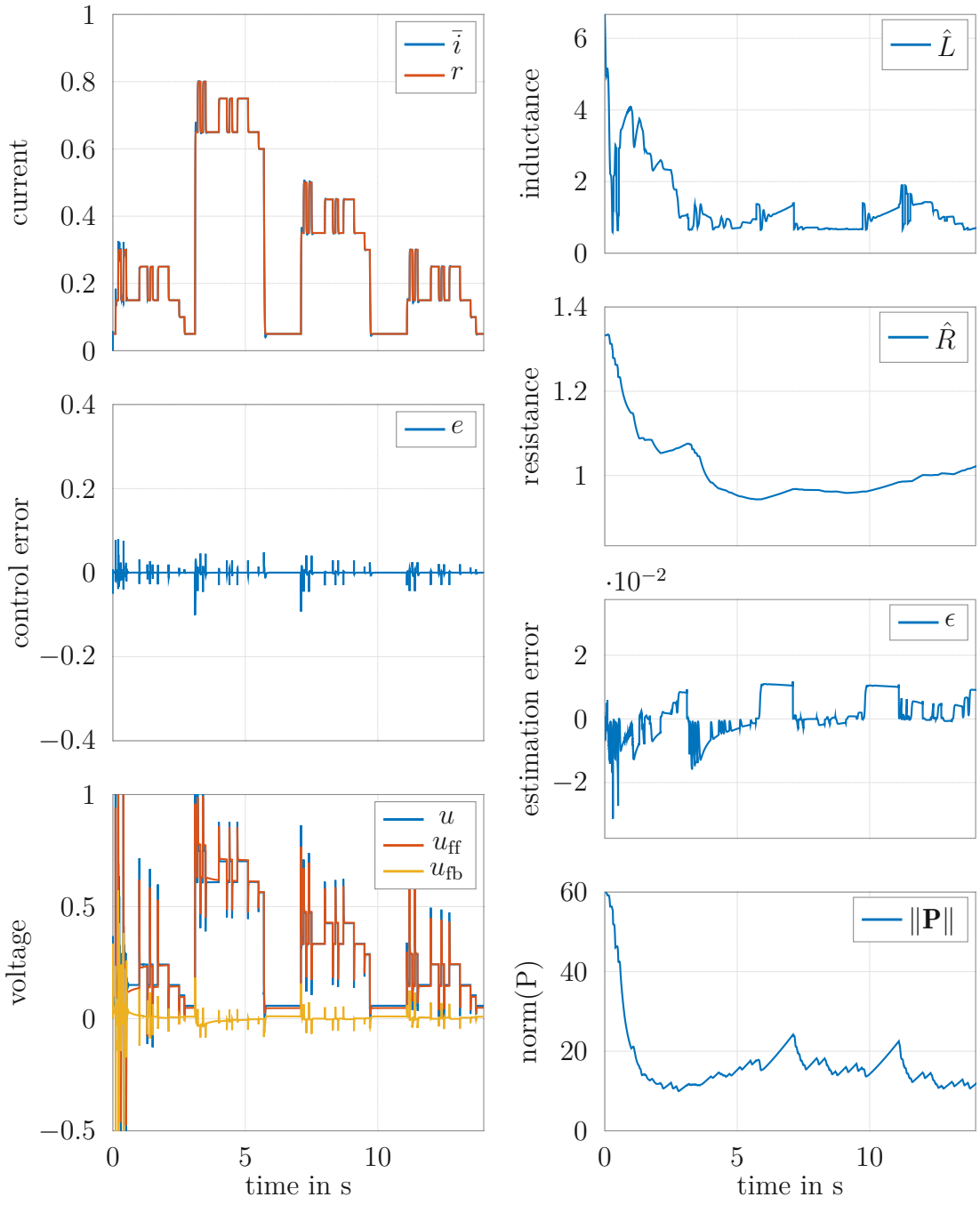

Figure 3.6: Experimental results of the indirect adaptive two-degrees-of-freedom control algorithm for solenoid A. The values are normalized to i_A, v_{bat}, L_A, and R_A.

Control		Adaptation		Projection	
$\alpha_0{}^*$	$10{,}000\,\mathrm{s}^{-2}$	L_0	$2.0L_A$	$L_{\min}$	$0.2L_A$
$\alpha_1{}^*$	$100\,\mathrm{s}^{-1}$	R_0	$0.68R_A$	$L_{\max}$	$5.0L_A$
		λ_a	$0.2\,\mathrm{s}^{-1}$	$R_{\min}$	$0.34R_A$
		P_0	60	$R_{\max}$	$1.36R_A$
		$\beta_{\max}$	0.6		
		$P_{\max}$	150		

Table 3.4: Parameters used for the experiment with the proposed indirect adaptive controller.

nominal values within the parameter range of the considered solenoids. The time constant λ_a of the low-pass filter (3.6) is essentially determined by the measurement noise when calculating the time derivative of the current y. The initial and maximum gain matrix, P_0 and $P_{\max}$, and the forgetting factor $\beta_{\max}$ were tuned according to the procedure presented in Section A.4 and can be treated similar to the classical least-squares tuning factors. The bounds for the inductance estimate, $L_{\min}$ and $L_{\max}$, and for the resistance estimate, $R_{\min}$ and $R_{\max}$, reported in Table 3.4, are selected to restrict the parameters to physically meaningful values. These bounds do not influence the transient performance of the overall algorithm.

In direct comparison with the model reference control and the sliding mode control scheme, Figure 3.6 shows experimental results of the proposed indirect adaptive control algorithm (3.14)-(3.19) applied to solenoid A. Here, the control error decays quickly after an initial convergence of the estimated plant parameters. The large contribution of the feedforward controller u_{ff} to the overall control input u suggests that the parametrized model accurately describes the physical plant. Hence, the feedback controller is used around the reference trajectory and can be tuned independently of the reference tracking control task. The repeated pattern of the reference signal at 11 s underlines the improvement of the control performance achieved by the adaptation. Here, the control performance is significantly improved compared to the reference signal controlled using the initial parameters at 0.1 s. At 3 s, the feedback controller shows an increased activity caused by the high current, which entails a decrease in the inductance. This effect is compensated by the feedback controller and does not significantly impact the control performance. Hence, the interaction between adaptation and integral feedback control combines fast convergence of the parameters with robustness to model uncertainties and un-

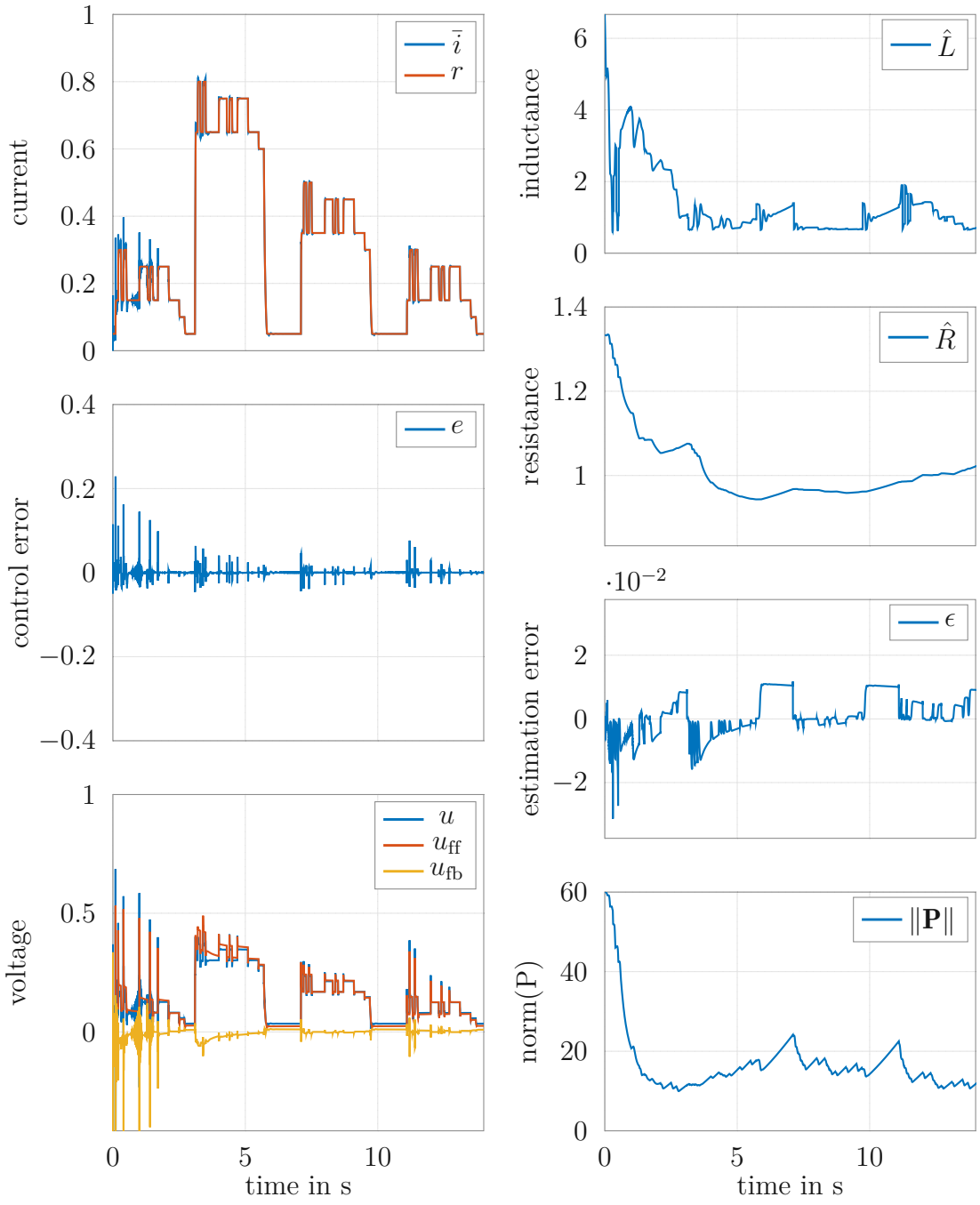

Figure 3.7: Experimental results of the indirect adaptive two-degrees-of-freedom control algorithm for solenoid B. The values are normalized to i_A, v_{bat}, L_B, and R_B.

modeled effects. Additionally, the least-squares adaptation algorithm from (3.14) uses the estimation error and can be adapted even without a control error. Thus, control errors due to disturbances are compensated by the feedback controller, whereas the parameters are updated when an estimation error occurs.

The estimated parameters, the normalized estimation error (3.11), and the norm of the gain matrix are depicted in Figure 3.6 (right side). The large estimation error and gain matrix norm in the first second of the experiment leads to a rapid convergence of the resistance and inductance estimates. The high initial gain value is used to reduce the estimation error quickly, while after this convergence phase, the gain matrix of the estimation algorithm (3.14) adapts to the current excitation. During periods of low excitation at 6 s and 10 s, the gain matrix is increased again by the exponential forgetting. Any parameter errors accumulated during this period are rapidly compensated for as soon as the parameters are excited again, as indicated by the estimation error. In contrast, a least-squares algorithm without exponential forgetting cannot neglect faulty measurements, even if new correct data is collected afterwards. Furthermore, the estimated parameters show only negligible drift in steady-state. In applications with long periods of insufficient excitation, modifications like a dead zone can be added to account for the lack of excitation in the reference signal, see, e.g., [19].

Solenoid B has approximately half the resistance and a drastically smaller inductance than solenoid A. However, Figure 3.7 shows that the adaptation algorithm of the *same* controller as the one used for solenoid A applied to solenoid B rapidly converges and establishes a small control error throughout the whole reference trajectory. Here, again, the nonlinear effect of the change in inductance at 3 s is compensated by the feedback control term u_{fb}. The feedforward part already achieves precise reference tracking after the initial convergence period. This is illustrated by the small feedback control action u_{fb} after about 5 s. After that, the control error stays well below $0.1 i_A$ even with rapid changes of the reference signal and periods of low excitation. Caused by the strong deviations of the initial conditions of the parameters from the real values, the controller shows some overshoots until the parameters have converged. Similar to the results with solenoid A, the parameters quickly converge, and after 3 s, excellent tracking performance and a low control error are achieved, see Figure 3.7 (right side).

The experimental results for solenoid C are depicted in Figure 3.8. The large initial value of the estimated inductance parameter causes overshoots during the first

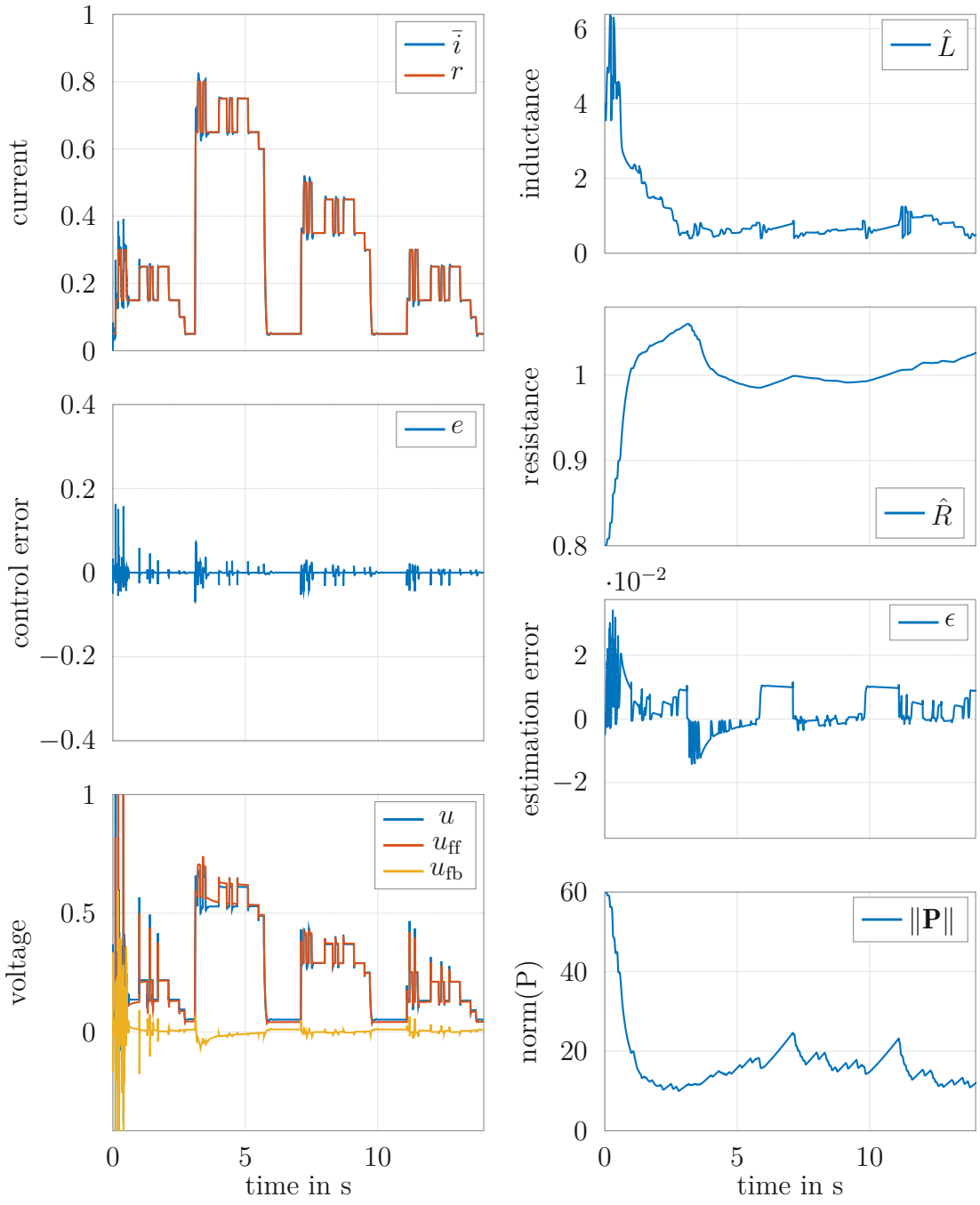

Figure 3.8: Experimental results of the indirect adaptive two-degrees-of-freedom control algorithm for solenoid C. The values are normalized to i_A, v_{bat}, L_C, and R_C.

second of the experiment and around 3 s due to the nonlinear inductance. However, from the right side of Figure 3.8, it can be seen that the inductance estimate decreases and eventually leads to excellent control performance. Furthermore, it should be noted that the estimated resistance parameter changes between 3 s and 4 s due to the excitation of the reference signal. After the initial convergence phase, the small feedback control action u_{fb}, shows a good match between the adaptively parametrized model and the controlled solenoid.

3.8 Conclusions

In this chapter, an indirect adaptive two-degrees-of-freedom control algorithm for the current control of solenoids is proposed in three parts.

First, an adaptive feedforward part extends the indirect adaptive pole placement scheme known from the literature, and the formulation is modified to improve its robustness. A thorough stability proof is provided for the overall closed-loop system comprising the plant, the constrained bounded-gain forgetting least-squares parameter estimation scheme, and the adaptive two-degrees-of-freedom control concept, described in Section 3.4. The adaptive feedforward part improves the tracking performance for rapidly changing reference trajectories, particularly for time-varying parameters. The constrained bounded-gain forgetting least-squares parameter estimation scheme ensures fast convergence of the parameters and does not suffer from excessive drift during periods of low excitation. Since the control design model does not account for the nonlinearity of the inductance and the time-varying parameters, the derived stability proof does not ensure stability for the nonlinear plant (3.4). However, the achieved closed-loop control performance achieved in the experiments justifies the proposed approach.

For practical applications, additional experiments should investigate application-specific effects, such as the characteristics of the valve opening, the fluid pressure on the solenoid plunger, and the magnetic flux of the used solenoid. Further research is to be conducted to improve the parameter convergence in situations of low excitation, which is an active field of research, see, e.g., [53]. The proposed adaptive control scheme strongly benefits from its property that the control error convergence does not rely on the convergence of the parameters. This alleviates the need for the persistence-of-excitation assumption and yields a good control performance without the persistence of excitation in the presented experiments.

Second, an experimental validation of the proposed control scheme is presented. The feasibility and good performance of the proposed approach are demonstrated by applying the control concept with one nominal controller tuning to three different solenoids from various applications with strongly differing parameters. Thus, for the whole range of different solenoids, only a single controller tuning is required, and the proposed adaptation scheme shows a robust and high-performance operation without further adjustments. This saves time and costs, particularlry during commissioning, and ensures high performance under changing loads and environmental conditions.
Third, experimental results of the performance of the proposed solution are compared with two well-known benchmark methods for solenoid control, taken from the literature, i.e., a robust second-order sliding-mode controller and a nonlinear model reference adaptive control approach. The sliding mode controller requires retuning for every solenoid, and the model reference adaptive controller exhibits a poor adaptation performance.

Chapter 4

Iterative Learning Control of a Robotic Application

Industrial robots are increasingly used in applications requiring high accuracy, such as additive manufacturing, precision machining, and medical robotics. While state-of-the-art model-based robot control strategies exhibit a high fidelity in compensating known dynamical effects, achieving sub-millimeter accuracy for dynamical movements of industrial robots remains a challenge. The main sources of errors comprise joint elasticities, the mismatched control input, friction effects, transmission errors, and kinematic errors due to tolerances, and wear, e.g., [78, 79, 80]. Compared to the uncertainties in Chapter 3, these effects are very challenging to model and depend on many parameters, such as the robot configuration and speed, dynamic load, transmission inconsistencies, and lubrication. Hence, in this case, a data-driven learning approach can leverage the observed dynamic effects, without relying on a complex dynamic model and many parameters.
Iterative learning control (ILC) can eliminate unknown, but repetitive effects by iteratively improving the accuracy of the robotic manipulator while traversing a path repeatedly. To improve the absolute accuracy of an industrial robot, a model-based controller is augmented with an ILC approach in this chapter.
Essential Parts of this chapter were accepted for publication in the author's work [81].

4.1 Literature Review

Absolute accuracy in task-space is crucial for industrial robots in demanding applications, such as advanced machining tasks, e.g., [82, 83], medical applications, and surgery robotics, e.g., [84]. Despite the importance of task-space accuracy, this topic has gained little attention in the literature compared to joint-space precision and approaches based on joint encoders, see, e.g., [85, 86]. However, it is

well known that significant dynamics of the robot links cannot be observed using motor-side encoders, see, e.g., [87]. Hence, the proposed method uses the full system knowledge in a computed-torque controller with feedforward compensation based on a mathematical robot model to achieve accurate task-space control. The controller leverages all typically available parameters of an industrial robot. Additionally, a cascaded velocity feedback controller is employed to counteract nonrepetitive errors. To further improve the accuracy of the control concept, an efficient iterative learning control (ILC) scheme is employed to compensate for unmodeled effects, such as parameter uncertainties, transmission error dynamics, and couplings between elastic and friction effects. This is in contrast to many existing works, where ILC is used to *substitute* for a model-based feedforward control, e.g., [88, 89]. Liu et al. [90] use a pure feedforward inverse-dynamics controller and a joint-space proportional ILC approach to improve precision. In this thesis, a full computed-torque is employed and since the reference path is modified by the ILC, the presented control law can be applied easily to other control architectures. A laser tracker system measures the absolute position of the tool center point, and motor encoders measure the orientation of the robot.

The transmission error dynamics, which are the main focus of this work, exhibit a nonlinear behavior and change drastically throughout the workspace. For example, imperfections of a gear tooth, can influence the end-effector motion up to a hundred times per joint revolution, see, e.g., [91]. Similarly, the coupling of gravitational load, elasticities, and friction effects causes a significant hysteresis and changes the observed damping for movements of a link with or against gravity. In contrast to [87], where unknown weights serve as a basis for a local learning model, we investigate trajectories spanning a large part of the workspace, and all available information about the process is used. Therefore, the remaining error dynamics are challenging to model, and hence, a simple data-driven proportional-derivative (PD) ILC learning law is employed to cope with the unknown dynamics.

An additional benefit of the proposed PD ILC law is that it does not depend on the complete trajectory and can be applied recursively, hence realizing an online ILC approach. In the literature, online ILC approaches like [92] use a linear error model, which does not reflect the nonlinear effects of the investigated error dynamics. In [93], an online approach for learning dynamic motion primitives is developed, and [94] presents a force learning task.

In order to achieve the desired absolute accuracy, the robot end-effector position

must be measured with high accuracy at a high rate. In the industry, laser trackers are typically used to calibrate robots to improve accuracy. Due to the high price of a laser tracker, the ILC inputs are pre-recorded and then used as a feedforward input. Therefore, the laser tracker is only needed in the initial learning phase. In order to be able to apply the pre-recorded feedforward signals to different execution speeds of the robot trajectory, the path parameter of the reference curve serves as an index variable. Different time-scaling approaches that rely on a system model have been reported in the literature, see, e.g., [95, 96, 97]. Compared to these approaches, faced with complex error dynamics, we take a simple path parameter ILC approach that allows us to scale the execution speed of the reference path. Phase indexing approaches are reported in the literature, see, e.g., [98, 99].
Combining the online ILC with the path parameter indexing alleviates two classical ILC assumptions: the requirement of fixed-length trials and the fixed starting point for each trial. In contrast to prior works, the combination of parameter indexing and online learning allows the proposed ILC to be applied to variable speed traversals, partially traversed paths, continuously executed trials, i.e., repetitive control problems, e.g., [100].

4.2 Contribution

The scientific contribution of this chapter can be summarized as follows: A simple, efficient, and flexible control scheme with learning control for the absolute accuracy of an industrial robot manipulator is presented. To fully leverage the ILC, the governing dynamics of the robot are compensated by a computed-torque and feedforward controller. The proposed ILC law can supplement the performance of existing control strategies with little requirements on the computational hardware and control structure.
In contrast to many prior investigations, this work focuses on the unknown transmission error dynamics and uses of all typically available parameters and a detailed robot model. Furthermore, the absolute accuracy of the robot, which is measured using a laser tracker, is the main focus of this work. Since the laser tracker may only be available for an initial learning phase, the efficacy of the learned ILC signals as feedforward trajectories is investigated. The proposed ILC approach is also applied to improve the tool orientation accuracy of the robot with motor encoder measurements only.

Additionally, the approach features high flexibility and ease of practical application. All calculations can be done recursively, and a continuous learning phase is possible even with variations in the reference trajectory speed. Further, the classical requirement of strictly repeating trials for the ILC is softened. Other works in the field, such as [87], use multiple filters that must be tuned after each trial. Here, we only use a single straightforward learning approach that does not require any intervention between the trials and achieves a comparable accuracy for a complex trajectory on industrial robot.

4.3 Problem Statement

This section aims to improve the absolute positioning accuracy of an industrial robot for a repeating trajectory by recursively learning unmodeled effects. The ILC approach used to learn these effects does not make assumptions about the employed control approach or the knowledge of robot parameters. To improve the absolute accuracy of the robot, a laser tracker measures the absolute position of the tool center point (TCP) of the robot. The experimental validation was done with the 6-axis industrial robot *Comau Racer 7-1.4* and the laser tracker *Leica Absolute Tracker AT960*, depicted in Figure 4.1.

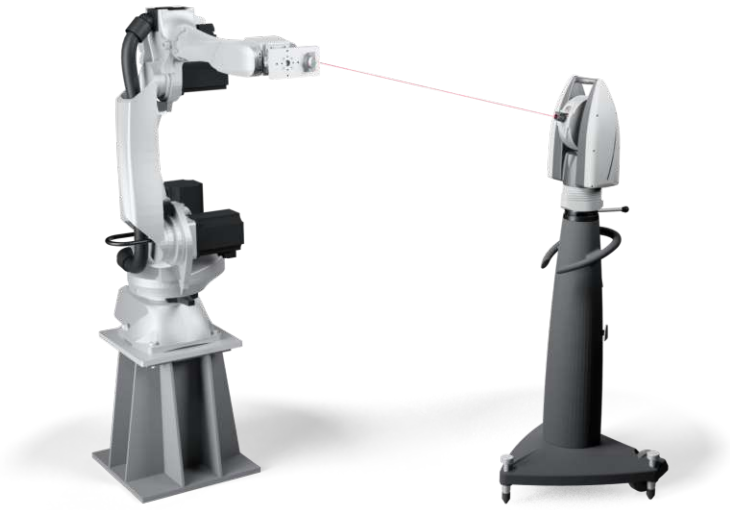

Figure 4.1: Industrial robot *Comau Racer 7-1.4* and *Leica Absolute Tracker AT960*.

4.3.1 Mathematical Model

The mathematical model of the rigid-body robot is given by

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = -\boldsymbol{\tau}_f + \boldsymbol{\tau}_c, \tag{4.1}$$

where $\mathbf{M}(\mathbf{q})$ is the mass matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$, the Coriolis matrix, and $\mathbf{g}(\mathbf{q})$ the vector of gravitational torques. Furthermore, $\boldsymbol{\tau}_f$ represents the torques caused by friction effects, and $\boldsymbol{\tau}_c$ are the control input torques on the robot joints. The mapping between the joint-space coordinates $\mathbf{q}$ and the Cartesian task-space position $\mathbf{x}$ of the TCP is given by the forward kinematics

$$\mathbf{x} = \mathbf{f}(\mathbf{q}) \ , \tag{4.2}$$

and the differential kinematics are defined using the manipulator Jacobian matrix $\mathbf{J}_v(\mathbf{q})$ by

$$\dot{\mathbf{x}} = \mathbf{J}_v\dot{\mathbf{q}} \ . \tag{4.3}$$

4.3.2 Control Concept

Let $\mathbf{q}_r, \dot{\mathbf{q}}_r$, and $\ddot{\mathbf{q}}_r$ be a joint reference trajectory and its derivatives. The control

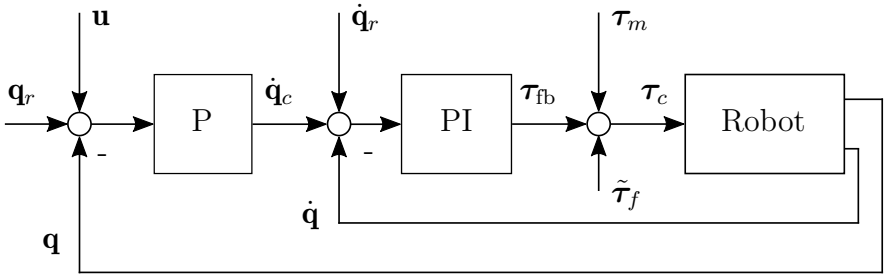

Figure 4.2: Flowchart of the robot control structure.

input $\boldsymbol{\tau}_c$ consists of three parts

$$\boldsymbol{\tau}_c = \boldsymbol{\tau}_m + \tilde{\boldsymbol{\tau}}_f + \boldsymbol{\tau}_{\text{fb}} \ , \tag{4.4}$$

with the computed-torque control

$$\boldsymbol{\tau}_m = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}_r + \mathbf{C}(\dot{\mathbf{q}}, \mathbf{q})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) \tag{4.5}$$

and the feedforward friction compensation term, e.g., [34],

$$\tilde{\boldsymbol{\tau}}_f = \mathbf{K}_v \dot{\mathbf{q}}_r + \mathbf{K}_c \text{sign}(\dot{\mathbf{q}}_r) \; . \tag{4.6}$$

The entries of the positive diagonal matrices $\mathbf{K}_v$ and $\mathbf{K}_c$ refer to the viscous and Coulomb friction coefficients in the joints, respectively, and were found from measurements.

In order to account for non-repetitive unmodeled effects, like transmission errors and elasticities, a cascaded feedback controller (the term $\boldsymbol{\tau}_{\text{fb}}$ in (4.4)) is employed with a PI velocity controller in the inner loop and a proportional position controller in the outer loop. Figure 4.2 depicts the block diagram of this state-of-the-art control concept. The PI velocity controller reads as

$$\boldsymbol{\tau}_{\text{fb}} = \mathbf{K}_{i,v} \int_0^t (\dot{\mathbf{e}}_q + \dot{\mathbf{q}}_c) \, \mathrm{d}\tilde{t} + \mathbf{K}_{p,v} \left(\dot{\mathbf{e}}_q + \dot{\mathbf{q}}_c \right) \; , \tag{4.7}$$

with the joint angle error

$$\mathbf{e}_q = \mathbf{q}_r - \mathbf{q} \tag{4.8}$$

and the positive definite controller gain diagonal matrices $\mathbf{K}_{p,v}$ and $\mathbf{K}_{i,v}$. The proportional position controller gives the output

$$\dot{\mathbf{q}}_c = \mathbf{K}_p (\mathbf{e}_q + \mathbf{u}) \; , \tag{4.9}$$

with the positive diagonal controller gain matrix $\mathbf{K}_p$. Moreover, $\mathbf{u}$ in (4.9) serves as input for the ILC scheme to be designed in the next section. The stability of computed-torque controllers is analyzed, e.g., in [34].

4.4 ILC Algorithm

In this section, the online path ILC algorithm is discussed, and the implementation in discrete-time is presented.

Let $\boldsymbol{\gamma}(\lambda(t)) : [0; 1] \mapsto \mathbb{R}^3$ denote a parametrized desired reference path in the task space of the robot. The user defines a desired execution speed by specifying the path parameter $\lambda(t) \in \mathcal{C}^2$. The tracking accuracy of the robot should be iteratively improved by the ILC each time the robot follows the trajectory $\boldsymbol{\gamma}(\lambda(t))$. Assume, without loss of generality, that the path parameter is normalized to the

interval $[0;1]$ and the path is traversed in one direction, i.e., $\dot{\lambda} \geq 0$. Thus, the reference position of the TCP is given by $\mathbf{x}_r(t) = \boldsymbol{\gamma}(\lambda(t))$. The joint-space reference trajectory $\mathbf{q}_r(t) = \mathbf{f}^{-1}(\mathbf{x}_r(t))$ can be found by using the inverse kinematics.
The ILC law uses a proportional-derivative (PD) type update for iteration i at the time t of the form

$$\boldsymbol{\alpha}_i(t) = \mathbf{u}_{i-1}(t) + \mathbf{K}_{p,ilc}\mathbf{e}_{q,i}(\lambda(t)) + \mathbf{K}_{d,ilc}\dot{\mathbf{e}}_{q,i}(\lambda(t)) \ , \tag{4.10}$$

with the constant positive diagonal matrices $\mathbf{K}_{d,ilc}$ and $\mathbf{K}_{p,ilc}$. Details on the application, robustness, and convergence of this algorithm are given e.g., in [101, 102]. The joint velocity error $\dot{\mathbf{e}}_q$, see (4.8), can be obtained from the high precision encoders of the robot. Assuming small deviations from the path, the TCP position from (4.2) can be written by the Taylor expansion

$$\mathbf{x} = \mathbf{x}_r - \left.\frac{\partial \mathbf{f}(\mathbf{q})}{\partial \mathbf{q}}\right|_{\mathbf{q}=\mathbf{q}_r} \mathbf{e}_q + \mathcal{O}(\mathbf{e}_q)^2 \ . \tag{4.11}$$

Thus, from (4.11) and (4.3), the positioning error $\mathbf{e}_q$ can be calculated from the laser tracker measurement in the form

$$\mathbf{e}_q \approx \mathbf{J}_v^{-1}(\mathbf{q}_r)\mathbf{e} \ , \tag{4.12}$$

$$\mathbf{e} = \mathbf{x}_r - \mathbf{x} \ . \tag{4.13}$$

The ILC update in (4.10) is parametrized by the path parameter $\lambda(t)$. Since we assume $\dot{\lambda} \geq 0$, the path parameter $\lambda(t)$ is a monotonic function of t and can be used to calculate the angle correction values $\mathbf{u}_i$ of the ILC law for the corresponding position along the path.

Remark 9. *The assumption $\dot{\lambda} \geq 0$ can be alleviated by splitting the trajectory into parts with common traversing direction and considering them as separate repetitive control problems.*

Next, the ILC input from (4.10) is filtered with the Q-filter

$$\mathbf{u}_i = Q * \boldsymbol{\alpha}_i \ , \tag{4.14}$$

which filters out nonrepetitive effects and high-frequency measurement noise. The

filter is realized by a Gaussian filter [103]

$$Q = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{\lambda^2}{2\sigma^2}\right) , \tag{4.15}$$

with the standard deviation

$$\sigma = \frac{\sqrt{\ln(2)}}{2\pi f} , \tag{4.16}$$

where the filter parameter f is the frequency at which the filter has a gain of -3dB.

4.4.1 Orientation Error of the Wrist Axes

Since the laser tracker cannot directly measure the orientation, the orientation of the end-effector is calculated from the motor encoder measurements. Hence, the axis-angle error of the wrist axes is

$$\mathbf{e}_{q,w} = \mathbf{q}_{w,r} - \mathbf{q}_w , \tag{4.17}$$

with the joint angles of the (wrist) axes 4, 5, and 6 $\mathbf{q}_w = [q_4, q_5, q_6]^{\mathrm{T}}$ and the corresponding reference $\mathbf{q}_{w,r}$. Analogously to (4.12), the wrist joint error $\mathbf{e}_{q,w}$ can be used in (4.10) to compensate for the orientation error. The proposed ILC concept can also be employed with a 6-dimensional measurement of the end-effector pose.

4.4.2 Online Implementation in Discrete-Time

In this section, the online implementation of the ILC law and the Q-filter is discussed. First, the parameter range $\lambda \in [0; 1]$ is split into N equidistant intervals of the length $\delta = 1/N$. Let $\lambda_k = \lambda(kT_s)$ denote the path parameter $\lambda(t)$ at the time $t = kT_s$ with the sampling time T_s and $k \in \mathbb{Z}$. Further, $l_k = \mathrm{round}(\lambda_k/\delta) \in [0; N-1]$ is the index value associated with the path parameter λ_k at the time kT_s. Subsequently, any variable x as a function of the discrete path parameter l is denoted by $x[l]$. The zero-order hold equivalent of (4.15) is given by

$$q_G[l] = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{l^2}{2\sigma^2}\right) . \tag{4.18}$$

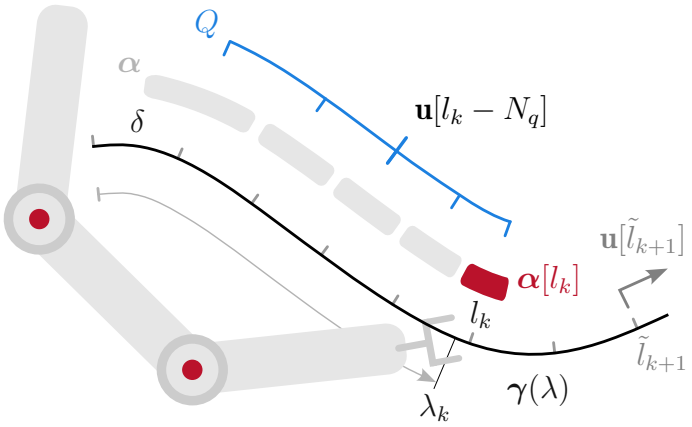

Figure 4.3: Schematic of a path $\boldsymbol{\gamma}(\lambda)$, the current robot TCP position at the path parameter λ_k, the stored update data vector $\boldsymbol{\alpha}$, the filter Q, and the ILC input $\mathbf{u}$.

This discrete-time filter is normalized over a square window with a total length of $2N_q + 1$ samples and applied to the ILC signal using the discrete-time convolution over $2N_q + 1$ path parameter indices, see also (4.14)

$$\mathbf{u}[k] = q * \boldsymbol{\alpha} = \sum_{m=-N_q}^{N_q} \frac{q_G[m]\boldsymbol{\alpha}[k-m]}{\sum_{m=-N_q}^{N_q} q_G[m]} \,. \tag{4.19}$$

The ILC algorithm is summarized in Algorithm 1, and a schematic of the algorithm is depicted in Figure 4.3.

Algorithm 1 Online path ILC algorithm

1: $l_k \leftarrow$round(λ_k/δ)
2: **if** $l_k \neq l_{k-1}$ **then** ▷ new λ interval
3: $\quad \mathbf{e} \leftarrow \mathbf{x}_r - \mathbf{x}$
4: $\quad \boldsymbol{\alpha}[l_k] \leftarrow$(4.10) ▷ save $\boldsymbol{\alpha}$ at l_k
5: $\quad \mathbf{u}[l_k - N_q] \leftarrow$(4.19) ▷ filter at $l_k - N_q$
6: **end if**
7: $l_{k+1} \leftarrow$round$((\lambda_{k+1} + t_d)/\delta)$
8: **return** $\mathbf{u}[l_{k+1}]$ ▷ return ILC input

Due to the discretization of the path parameter, an interval of the path parameter can remain active for multiple time instances k. This is particularly clear when stopping the execution, i.e., $\dot{\lambda} = 0$ and $l_{k+1} = l_k$. Here, the algorithm is updated

with the first encountered value for each interval. This is realized by updating each interval once per trial and otherwise outputting the same value repeatedly, see Line 2 of Algorithm 1. In Line 4 of Algorithm 1, the currently measured error signal is mapped to the joint-space of the robot, and the ILC law (4.10) is applied. The calculated update is stored in $\boldsymbol{\alpha}[l_k]$ at the current path parameter index l_k, see also Figure 4.3. Finally, the value at the path parameter index $l_k - N_q$ is filtered in Line 5 of Algorithm 1, which yields the ILC correction value $\mathbf{u}[l_k - N_q]$. The time delay t_d and the path parameter are used to calculate the next path parameter value $l_{k+1} = \text{round}((\lambda_{k+1} + t_d)/\delta)$. The associated joint-angle corrections $\mathbf{u}[l_{k+1}]$ serve as the ILC input signal to compensate for the time delay of the system. The benefit of this recursive algorithm lies in the flexible calculation scheme and the computational efficiency.

4.5 Experimental Validation

The 6-axis industrial robot *Comau Racer 7-1.4* and a *Hexagon AT960* laser tracker were used for the experimental validation of the proposed online path ILC approach, see Figure 4.1. It is well known that the joint-angle offsets play an essential role in the absolute accuracy of an industrial robot. In order to calibrate the joint positions, the robot was moved to a point on the reference trajectory, and a static calibration of all joints was performed by offsetting the axes to match the TCP position measured by the laser tracker. The orientation error is measured by the ZYX-Euler-angle representation $\mathbf{e}_o$ of the error between the reference and actual rotation matrix $\mathbf{R}_e = \mathbf{R}^e_{0,ref}(\mathbf{R}^e_0)^{\mathrm{T}}$. The control parameters for all experiments are summarized in Table 4.1.

Two experiments were conducted to validate the proposed ILC concept. In Experiment 1, the convergence and error caused by a variation in the execution speed of the robot is investigated. In Experiment 2, pre-recorded values are used for the feedforward control and the ILC law is applied without further usage of the laser tracker. In this experiment, the laser tracker measurement is used for validation only.

In Figure 4.4, the reference path and the deviation from the path amplified by a factor of 50 are depicted for trial 1 and trial 2 of Experiment 1. The path covers a large part of the workspace and contains segments where $\frac{\partial \gamma}{\partial \lambda}$ is small, resulting in slow movements at a constant parameter speed. Conversely, the path also features

$$
\begin{aligned}
\mathbf{K}_{p,v} &= \mathrm{diag}(1440, 1440, 500, 20, 14, 11) \\
\mathbf{K}_{i,v} &= \mathrm{diag}(1200, 2560, 1700, 104, 75, 65) \\
\mathbf{K}_{p} &= 20\,\mathbf{I} \\
\mathbf{K}_{p,ilc} &= 1\,\mathbf{I} \\
\mathbf{K}_{d,ilc} &= 0.01\,\mathbf{I} \\
N_q &= 45 \\
f &= 5\,\mathrm{Hz} \\
T_s &= 1\,\mathrm{ms} \\
N &= 10000
\end{aligned}
$$

Table 4.1: Control and ILC parameters.

trial	$\|\mathbf{e}\|_\infty$ in mm	RMS($\mathbf{e}$) in mm	$\|\mathbf{e}_o\|_\infty$ in 10^{-3} deg	RMS($\mathbf{e}_o$) in 10^{-3} deg
0	3.238	1.8599	51.92	26.09
1	0.258	0.1191	23.93	10.91
2	0.166	0.0767	21.56	10.31
3	0.749	0.3923	21.15	10.12
4	0.186	0.0564	20.40	10.19
5	3.842	2.1570	46.88	24.37
6	0.183	0.0564	21.03	10.21

Table 4.2: Position and orientation error for the seven trials of Experiment 1.

fast movements where $\frac{\partial \gamma}{\partial \lambda}$ is large. There are three reversing points, one in the direction of the gravitational acceleration. These features render the reference trajectory challenging for a high-precision application, since friction effects and joint elasticities play an essential role in this scenario.

The position error and the corresponding ILC input of Experiment 1 are given in Figure 4.5, where the gray-shaded areas mark the seven trials. Table 4.2 lists the maximum and RMS errors for each trial. In Experiment 1, the maximum error improves by 92 % after one trial. After trial 2, the path-parameter speed is increased from $\dot{\lambda} = 0.1$ to 0.12 trials/s. Despite this change in the execution speed of the robot, the maximum error of trial 3 is 76 % lower compared to trial 0. The slightly increased error due to the change in the execution speed is reduced to the level achieved earlier after learning at this speed for one trial. In trial 4, the maximum error is reduced by 94 % compared to trial 0. Finally, in trial 5, the ILC correction output and learning are deactivated to show the baseline error at

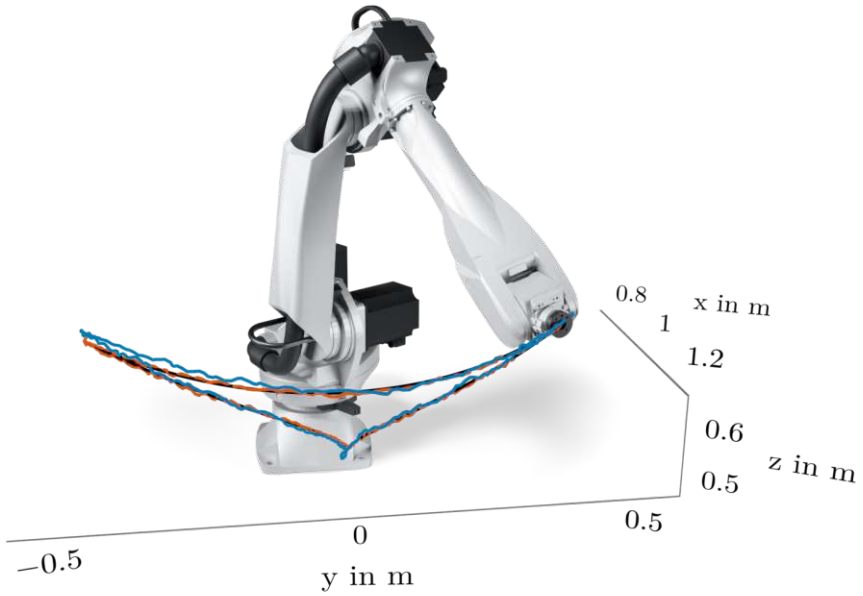

Figure 4.4: Experiment 1: Robot trajectories with 50 times amplified deviation from the path. The reference path is drawn in black, trial 1 in blue, trial 2 in orange.

this execution speed and the improvement using the ILC signals again in trial 7, where the ILC is activated. The maximum error throughout trials 2, 4, and 6 is lower than 200 µm, and the RMS error is lower than 80 µm.

In Figure 4.6, the orientation error of Experiment 1 is depicted. After the first learning trial, the error improves by more than a factor of 2. This significant improvement is retained throughout trial 3, where the execution speed is increased by 20 %. However, the orientation error does not converge further after the first trial. This is a consequence of the joint-space orientation ILC (4.17) based on the motor encoder measurements.

Experiment 2 is conducted with pre-learned ILC signals in a feedforward sense. No further learning is done in this experiment, and the laser tracker was used only to validate the accuracy that can be achieved when using the pre-learned signals as feedforward input. Figure 4.7 shows the position error and the pre-learned ILC feedforward input for the six trials of Experiment 2. During the trials 1, 2, 3, and 5, a maximum accuracy below 200 µm is achieved without updating the ILC data. The maximum position and orientation errors and RMS errors during the six trials of Experiment 2 are listed in Table 4.3. From this table, it is clear that

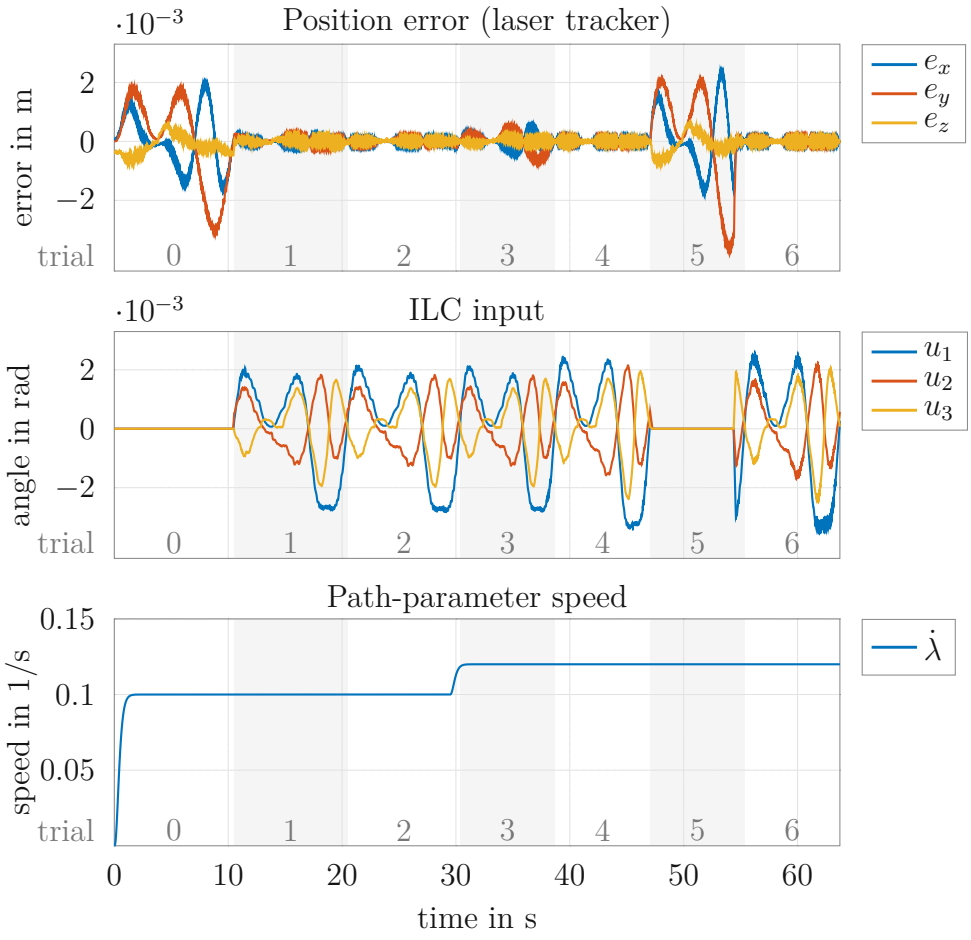

Figure 4.5: Experiment 1 (ILC active): Position error, corresponding ILC input, and path-parameter speed for seven trials marked by the gray-shaded areas (trial 3: change in the robot's execution speed; trial 5: ILC turned off).

the absolute accuracy of the robot can be drastically improved with the proposed method, even if an expensive laser tracker is only used for an initial learning phase.

In trial 4, the parameter speed is increased from $\dot{\lambda} = 0.1$ to 0.12 trials/s, but the feedforward signal was not changed to account for this higher execution speed. Here, only slightly larger errors occur. The accuracy is improved by more than a factor of 4 compared to trial 0, even with a 20 % change in the execution speed of the robot. Finally, in trial 5, the execution speed is reduced again, and the execution speed matches the speed when the feedfoward signal was learned. Here, the same absolute accuracy is achieved, as in trials 1, 2, and 3.

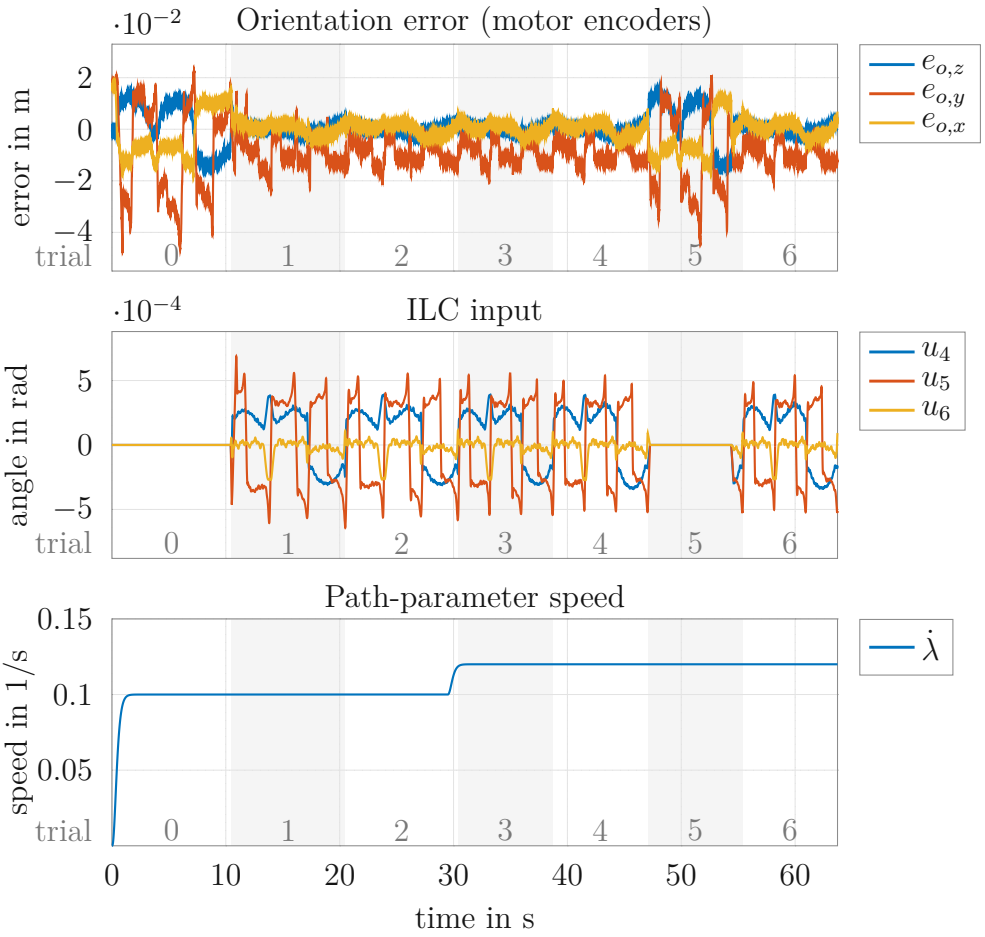

Figure 4.6: Experiment 1 (ILC active): Orientation error, corresponding ILC input, and path-parameter speed for seven trials marked by the gray-shaded areas (trial 3: change in the robot's execution speed; trial 5: ILC turned off).

4.6 Conclusion

In this chapter, a numerically efficient flexible online path iterative learning control (ILC) scheme was presented and practically validated. The practical experiments confirm a rapid convergence during continuous task execution with an accuracy improvement of 95 % after two trials. The proposed path-parameter-based ILC can be easily employed for different and even varying execution speeds where a significant increase in execution accuracy is retained. Similarly, learning from partial trials is possible with the presented approach. Adaptating the ILC input to a different execution speed can be achieved with a single trial.

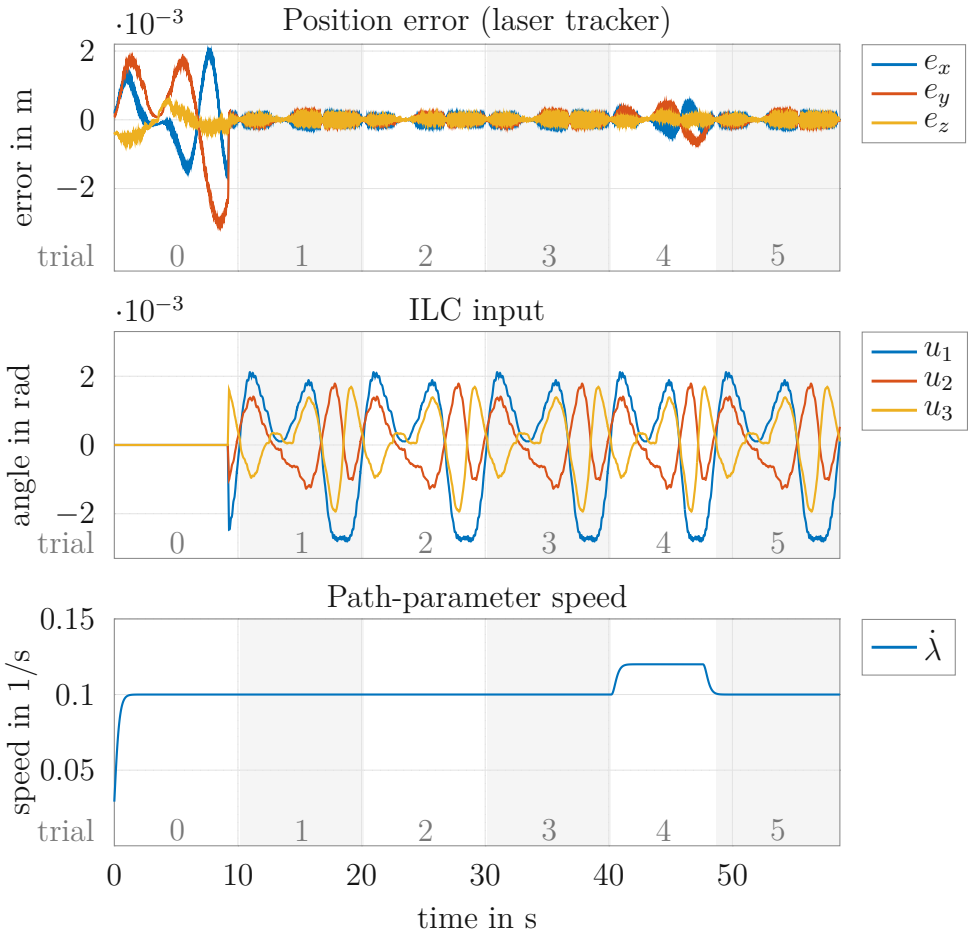

Figure 4.7: Experiment 2 (ILC only used for feedforward control): Position error, corresponding pre-trained ILC feedforward input, and path-parameter speed for six trials marked by the gray-shaded areas (trial 4: change in the execution speed of the robot).

The experiments were conducted with an external laser tracker absolute measurement system. It is shown that the presented method can significantly improve the absolute accuracy of the robot. Due to the high repeatability of industrial robots the achieved accuracy can be sustained even without this measurement system and trial-by-trial learning. The orientation was measured using the motor encoders of the robot, highlighting the adaptation capabilities of the approach to different measurement scenarios. The maximum error of the orientation was reduced by a factor of 2. The presented approach approach shows multiple practical advantages and can be easily deployed to different path tracking problems and

trial	$\|\mathbf{e}\|_\infty$ in mm	RMS($\mathbf{e}$) in mm	$\|\mathbf{e}_o\|_\infty$ in 10^{-3} deg	RMS($\mathbf{e}_o$) in 10^{-3} deg
0	3.226	1.8493	48.73	25.18
1	0.196	0.0776	19.23	9.95
2	0.192	0.0773	19.92	9.98
3	0.194	0.0782	20.22	10.04
4	0.751	0.3922	20.91	9.97
5	0.199	0.0775	20.01	10.03

Table 4.3: Position and orientation error for the six trials of Experiment 2.

robotic systems. A drawback of the presented approach is the simple ILC-law. It is an open research question to combine a model-based learning filter with the presented path-ILC framework. Moreover, improving the achieved performance of the method for variations in execution speed is a topic to be investigated. In addition, applications with contacts between the manipulator and the environment present topics for future research. Further investigations could include the generalization of pre-learned ILC-data to different paths.

Chapter 5

Conclusion and Outlook

This thesis presents an adaptive control and a learning control scheme combined with model-based controllers. The adaptive control scheme improves the performance and facilitates the deployment of the current control of solenoid valves and the learning control scheme improves the task-space accuracy of an industrial robot. Combining domain knowledge and model information with adaptive and learning strategies greatly benefits the closed-loop performance of the investigated control systems. Furthermore, the presented control approaches are computationally efficient and easy to deploy in practical scenarios.

The general least-squares (LS) framework presented in Chapter 2 recovers many methods from the literature as special cases. The interpretation of the forgetting matrix and the general least-squares cost functional proved helpful for designing problem-tailored estimation algorithms leveraging domain knowledge. In Chapter 3 it was shown how the provided proof of relevant properties of this general parameter estimation algorithm can be exploited to guarantee the stability of the overall closed-loop system comprising the plant, the estimation algorithm and the adaptive controller.

Further investigations could include the behavior of these algorithms in situations, where only limited excitation of the parameters is available for extended periods. Additionally, an extension of the proofs with time-varying parameters is an open research question.

In Chapter 3 a two-degrees-of-freedom adaptive control scheme was presented. The formulation of the adaptation problem from the literature was adapted to the given application, and a feedforward controller was added, outperforming classical benchmark methods. The proposed concept was applied to three solenoids with different designs in an experimental setup and exhibited high performance without individual tuning. This application shows that combining adaptation strategies and model-based control provides additional capabilities over pure model-based controllers.

In this context, future research topics include extending the stability proof to plants with time-varying parameters, rapidly changing, and highly nonlinear effects.
In Chapter 4, a numerically efficient online path iterative learning control (ILC) scheme for improving the absolute accuracy of an industrial robot was presented and experimentally validated. The key novelties are the online update of the PD-type ILC algorithm and the use of the path parameter for parametrizing the ILC approach. This combination facilitates the practical application of ILC and can be easily deployed to different path-tracking problems and robotic systems. The use of the ILC inputs as feedforward control signals achieves a significant increase in the absolute accuracy of the robot, even without requiring external measurements of the absolute end-effector position by a laser tracker. This application highlights that the performance of model-based controllers can be significantly improved by employing a combination of model-based and learning control.
The proposed ILC scheme can also be used with different task execution speeds, significantly improving the absolute accuracy without further learning. Generalizing the feedforward control signals trained by ILC for a specific path to different spacial trajectories is an open research question.
In summary, this thesis presents an adaptation and learning control scheme to significantly enhance the performance and practical application of pure model-based control approaches. Model-based controllers achieve a high level of performance in a wide range of applications. However, difficulties in parametrization and the challenges with modeling complex dynamical effects pose fundamental limits to this approach. While pure learning control can provide robust adaptation to uncertainties, the combination of model-based control strategies and learning control significantly improves the performance of the closed-loop system since a priori knowledge in the form of the models and available measurement data are systematically exploited.
Further, due to the included domain knowledge, even computationally efficient learning strategies can be employed to greatly enhance the performance of a control system. Similarly, the combination of learning and model-based control can augment the capabilities of the closed loop an facilitate the practical application. In conclusion, the combination of model-based and learning control methods provide significant advantages compared to a pure model-based, or pure learning control approach. The field of combined learning control holds many interesting challenges for future research.

Chapter A

Appendix

A.1 Useful Definitions and Theorems

This section contains some definitions and theorems that are used in this thesis.

Definition 1 (Persistent Excitation, see [19], p.177)**.**—A signal $\boldsymbol{\varphi}(t)$ is persistently exciting, if there exist constants $\alpha_0, \alpha_1, T_0 > 0$ such that

$$\alpha_0 \mathbf{I} \leq \frac{1}{T_0} \int_{t_0}^{t_0+T_0} \boldsymbol{\varphi}(\tau)\boldsymbol{\varphi}^{\mathrm{T}}(\tau)\,\mathrm{d}\tau \leq \alpha_1 \mathbf{I} \tag{A.1}$$

holds for all $t_0 > 0$.

Definition 2 (Uniform Complete Observability, see [19], p.90)**.**—The pair of matrices $(\mathbf{C}(t), \mathbf{A}(t))$ of the system

$$\dot{\mathbf{x}} = \mathbf{A}(t)\mathbf{x} + \mathbf{B}(t)\mathbf{u} \tag{A.2}$$

$$\mathbf{y} = \mathbf{C}(t)\mathbf{x} + \mathbf{D}(t)\mathbf{u}\,, \tag{A.3}$$

where $\mathbf{A}(t)$, $\mathbf{B}(t)$, $\mathbf{C}(t)$, and $\mathbf{D}(t)$ are bounded functions of the time t, is uniformly completely observable, if there exist constants $\beta_1, \beta_2, \nu > 0$ such that

$$\beta_1 \mathbf{I} \leq \mathcal{O}(t_0, t_0+\nu) \leq \beta_2 \mathbf{I} \tag{A.4}$$

holds for all $t_0 \geq 0$, where

$$\mathcal{O}(t_0, t_0+\nu) = \int_{t_0}^{t_0+\nu} \boldsymbol{\Phi}^{\mathrm{T}}(\tau, t_0)\mathbf{C}^{\mathrm{T}}(\tau)\mathbf{C}(\tau)\boldsymbol{\Phi}(\tau, t_0)\,\mathrm{d}\tau \tag{A.5}$$

is the observability Gramian, and $\boldsymbol{\Phi}(\tau, t_0)$ is the state transition matrix associated with $\mathbf{A}(t)$.

Theorem 5 (Stability of Linear Time-Varying Systems, see [19], p.122)**.** *A necessary and sufficient condition for uniform asymptotic stability of the equilibrium* $\mathbf{x}_e = \mathbf{0}$ *of the system* $\dot{\mathbf{x}} = \mathbf{A}(t)\mathbf{x}$ *is that there exists a positive definite matrix* $\mathbf{\Gamma}(t)$ *such that the conditions*

1. $\gamma_1 \mathbf{I} \leq \mathbf{\Gamma}(t) \leq \gamma_2 \mathbf{I}$

2. $\dot{\mathbf{\Gamma}}(t) + \mathbf{A}^{\mathrm{T}}(t)\mathbf{\Gamma}(t) + \mathbf{\Gamma}(t)\mathbf{A}(t) + \nu \mathbf{C}^{\mathrm{T}}(t)\mathbf{C}(t) \leq 0$

are satisfied for all $t \geq 0$ *and some constants* $\nu, \gamma_1, \gamma_2 > 0$ *and* $\mathbf{C}$*, is such that the pair* $(\mathbf{C}(t), \mathbf{A}(t))$ *is UCO.*

Lemma 1 (see[19], p.221)**.** *Assume that there exist constants* $\nu, k_v > 0$ *for all* $t_0 > 0$*, and* $\mathbf{K} \in \mathbb{R}^{n \times l}$ *satisfies the inequality*

$$\int_{t_0}^{t_0+\nu} \|\mathbf{K}(\tau)\|^2 \, \mathrm{d}\tau \leq k_v \,, \quad \forall t \tag{A.6}$$

then $(\mathbf{C}, \mathbf{A})$ *with* $\mathbf{C} \in \mathbb{R}^{l \times n}$ *and* $\mathbf{A} \in \mathbb{R}^{n \times n}$ *is a UCO pair if and only if* $(\mathbf{C}, \mathbf{A} + \mathbf{KC})$ *is a UCO pair.*

Lemma 2. *Given a stable proper transfer function* $W(s)$ *and two differentiable signals* $t \mapsto \mathbf{a}(t)$ *and* $t \mapsto \mathbf{b}(t)$ *such that* $\mathbf{b} \in \mathcal{L}_\infty$ *and* $\dot{\mathbf{a}} \in \mathcal{L}_\infty \bigcap \mathcal{L}_2$*, there exists a signal* $\rho \in \mathcal{L}_\infty \bigcap \mathcal{L}_2$ *such that*

$$W\{\mathbf{a}^{\mathrm{T}}\mathbf{b}\} = \mathbf{a}^{\mathrm{T}} W\{\mathbf{b}\} + \rho \,. \tag{A.7}$$

A.2 Proof of Theorem 2

The proof of Theorem 2 proceeds similarly to proofs for other methods in [19]. First, the parameter error differential equation is written in a linear form. Then, the exponential stability of the equilibrium point $\tilde{\boldsymbol{\theta}} = \mathbf{0}$ is established.

Proof. Because the ideal parameter vector $\boldsymbol{\theta}^*$ is assumend to be constant,

$$\dot{\tilde{\boldsymbol{\theta}}} = \mathbf{P}\boldsymbol{\varphi}\epsilon \,. \tag{A.8}$$

With (2.14), this equation can be rewritten in the form

$$\dot{\tilde{\boldsymbol{\theta}}} = \mathbf{A}\tilde{\boldsymbol{\theta}}\,, \qquad \mathbf{A} = -\mathbf{P}\boldsymbol{\varphi}\boldsymbol{\varphi}^{\mathrm{T}} \tag{A.9a}$$

$$\epsilon = \mathbf{C}\tilde{\boldsymbol{\theta}}\,, \qquad \mathbf{C} = -\boldsymbol{\varphi}^{\mathrm{T}}, \tag{A.9b}$$

which is a linear time-varying system. From (2.15), and since $\mathbf{F} \geq \mathbf{0}$, the derivative of the Lyapunov-like function (2.8) yields

$$2\dot{\mathcal{V}} = -\epsilon^2 - \tilde{\boldsymbol{\theta}}^{\mathrm{T}}\mathbf{F}\tilde{\boldsymbol{\theta}} \leq -\epsilon^2\,. \tag{A.10}$$

Combining this equation with (2.11) and (A.9) yields

$$-\tilde{\boldsymbol{\theta}}^{\mathrm{T}}\mathbf{F}\tilde{\boldsymbol{\theta}} = \tilde{\boldsymbol{\theta}}^{\mathrm{T}}\left(\mathbf{A}^{\mathrm{T}}\mathbf{R} + \dot{\mathbf{R}} + \mathbf{R}\mathbf{A} + \mathbf{C}^{\mathrm{T}}\mathbf{C}\right)\tilde{\boldsymbol{\theta}} \leq 0\,. \tag{A.11}$$

This implies that the matrix $\mathbf{A}^{\mathrm{T}}\mathbf{R} + \dot{\mathbf{R}} + \mathbf{R}\mathbf{A} + \mathbf{C}^{\mathrm{T}}\mathbf{C}$ is negative semidefinite. Theorem 5 in Appendix A.1 with $\boldsymbol{\Gamma} = \mathbf{R}$, $\gamma_1 = P_{\max}^{-1}$, $\gamma_2 = P_{\min}^{-1}$, and $\nu = 1$ guarantees exponential stability of the system (A.9), if $(\mathbf{C}, \mathbf{A})$ is uniformly completely observable (UCO), see Definition 2 in Appendix A.1. With Lemma 1 in Appendix A.1, the UCO pair can be established by comparison to a modified system. In particular, the choice $\mathbf{K} = -\mathbf{P}\boldsymbol{\varphi} \in \mathbb{R}^{n\times 1}$ yields a trivial system. Using the upper bounds $\|\mathbf{P}\| \leq P_{\max}$ and $\|\boldsymbol{\varphi}\| \leq 1$ reveals that

$$\int_{t_0}^{t_0+\nu} \|\mathbf{K}(\tau)\|^2\,\mathrm{d}\tau \leq P_{\max}^2\,\nu =: k_v \tag{A.12}$$

and, thus, the requirements to apply Lemma 1 are satisfied for this choice of $\mathbf{K}$. The shifted system takes the form

$$\dot{\mathbf{Y}} = \mathbf{A} + \mathbf{K}\mathbf{C} = \mathbf{0} \tag{A.13}$$

$$y = \mathbf{C}\mathbf{Y} = -\boldsymbol{\varphi}^{\mathrm{T}}\mathbf{Y} \tag{A.14}$$

and the associated observability Gramian (A.5) reads as

$$\mathcal{O}(t_0, t_0+\nu) = \int_{t_0}^{t_0+\nu} \boldsymbol{\varphi}(\tau)\boldsymbol{\varphi}^{\mathrm{T}}(\tau)\,\mathrm{d}\tau\,. \tag{A.15}$$

Because the regression vector $\boldsymbol{\varphi}$ is persistently exciting, see Definition 1 in Ap-

pendix A.1, there exist constants $\alpha_0, \alpha_1, \nu > 0$ such that

$$\alpha_0 \nu \mathbf{I} \leq \int_{t_0}^{t_0+\nu} \boldsymbol{\varphi}(\tau)\boldsymbol{\varphi}^{\mathrm{T}}(\tau)\,\mathrm{d}\tau \leq \alpha_1 \nu \mathbf{I}\,,\ \forall t_0 > 0\,. \tag{A.16}$$

Equation (A.16) limits the observability Gramian (A.15), and $(\mathbf{C}, \mathbf{A} + \mathbf{KC})$ is UCO. According to Lemma 1 in Appendix A.1, also the pair $(\mathbf{C}, \mathbf{A})$ is UCO, and Theorem 5 guarantees uniform asymptotic stability and, by linear system theory, uniform exponential stability of the equilibrium $\tilde{\boldsymbol{\theta}} = \mathbf{0}$. Thus, the parameters $\boldsymbol{\theta}(t)$ converge exponentially to their ideal values $\boldsymbol{\theta}^*$.

□

A.3 Proof of Theorem 4

The proof proceeds similarly to the proof presented in [19, p. 471]. However, there are essential differences from the original proof, such as the integral feedback path, the feedforward controller, and the formulation of the least-squares problem. Therefore, in the following, the main aspects of the proof are sketched.
The filtered system input and output can be written with (3.7) as

$$\dot{u}_a = -\lambda_a u_a + \lambda_a u\,, \quad u_a(0) = u_0\,, \tag{A.17a}$$
$$\dot{y}_a = -\lambda_a y_a + \lambda_a y\,, \quad y_a(0) = y_0\,, \tag{A.17b}$$

with initial conditions u_0 and y_0.
Consequently, the least-squares estimation error is given by, cf. (3.11),

$$\epsilon = \frac{u_a - \hat{L}\dot{y}_a - \hat{R}y_a}{m^2}\,, \tag{A.18}$$

with the normalization factor $m^2 = 1 + y_a^2 + \dot{y}_a^2$ and the estimated parameters $\hat{L}$ and $\hat{R}$. In addition, the control input from (3.19) can be written as

$$u = \hat{k}_p(r - y) + \hat{k}_i x_c + \hat{L}\dot{r} + \hat{R}r\,, \tag{A.19}$$

with the integral control error x_c from (3.15b). The reference trajectory must be chosen such that $r, \dot{r} \in \mathcal{L}_\infty$, which is satisfied by the assumptions in Section 3.5.
The proof of Theorem 4 is performed in four steps, as listed in Section 3.5. Rear-

rangement of (A.17), (A.18), and (A.19) yields the system

$$\dot{\boldsymbol{\psi}} = \mathbf{A}(t)\boldsymbol{\psi} + \mathbf{b}_1(t)\epsilon m^2 + \mathbf{b}_2(t)r + \mathbf{b}_3(t)\dot{r} \ , \tag{A.20}$$

where

$$\boldsymbol{\psi} = \begin{bmatrix} u_a \\ y_a \\ x_c \end{bmatrix} , \ \mathbf{b}_1(t) = \frac{1}{\hat{L}} \begin{bmatrix} \hat{k}_p \\ -1 \\ \frac{1}{\lambda_a} \end{bmatrix} , \tag{A.21}$$

$$\mathbf{b}_2(t) = \begin{bmatrix} \lambda_a(\hat{k}_p + \hat{R}) \\ 0 \\ 1 \end{bmatrix} , \ \mathbf{b}_3(t) = \begin{bmatrix} \lambda_a \hat{L} \\ 0 \\ 0 \end{bmatrix} \tag{A.22}$$

and

$$\mathbf{A}(t) = \begin{bmatrix} -\lambda_a - \frac{\hat{k}_p}{\hat{L}} & -\lambda_a \hat{k}_p + \frac{\hat{R}}{\hat{L}}\hat{k}_p & \lambda_a \hat{k}_i \\ \frac{1}{\hat{L}} & -\frac{\hat{R}}{\hat{L}} & 0 \\ -\frac{1}{\lambda_a \hat{L}} & \frac{\hat{R}}{\hat{L}\lambda_a} - 1 & 0 \end{bmatrix} . \tag{A.23}$$

The input and output of the plant can be written as an output of this system by substituting (A.20) into (A.17a), and (A.17b), yielding

$$\begin{bmatrix} u \\ y \end{bmatrix} = \mathbf{C}(t)\boldsymbol{\psi} + \mathbf{d}_1(t)\epsilon m^2 + \mathbf{d}_2(t)r + \mathbf{d}_3(t)\dot{r} \ , \tag{A.24}$$

with the output matrix and vectors

$$\mathbf{C}(t) = \begin{bmatrix} -\frac{\hat{k}_p}{\lambda_a \hat{L}} & \frac{\hat{k}_p \hat{R}}{\lambda_a \hat{L}} - \hat{k}_p & \hat{k}_i \\ \frac{1}{\lambda_a \hat{L}} & 1 - \frac{\hat{R}}{\lambda_a \hat{L}} & 0 \end{bmatrix} , \ \mathbf{d}_1(t) = \begin{bmatrix} \frac{\hat{k}_p}{\lambda_a \hat{L}} \\ -\frac{1}{\lambda_a \hat{L}} \end{bmatrix} \tag{A.25}$$

$$\mathbf{d}_2(t) = \begin{bmatrix} \hat{R} + \hat{k}_p \\ 0 \end{bmatrix} , \ \mathbf{d}_3(t) = \begin{bmatrix} \hat{L} \\ 0 \end{bmatrix} . \tag{A.26}$$

Due to the projection, the adaptation algorithm (3.14) ensures that $\hat{R}$ and $\hat{L}$ are bounded from below and above. In particular, $0 < L_{\min} \leq \hat{L}$, which guarantees that $\mathbf{A}(t)$, $\mathbf{b}_i(t)$, $\mathbf{C}(t)$, and $\mathbf{d}_i(t)$ $i = 1, 2, 3$ are bounded.

Next, it will be shown that the homogeneous part of (A.20) is exponentially stable.

This will be done by showing that the eigenvalues of $\mathbf{A}(t)$, are negative for all times t, and the induced norm $\|\dot{\mathbf{A}}(t)\| \in \mathcal{L}_2$. The characteristic polynomial of $\mathbf{A}(t)$ reads as

$$\det(\mathbf{A}(t) - s\mathbf{I}) = (s + \lambda_a)(s^2 + \alpha_1^* s + \alpha_0^*) \; . \tag{A.27}$$

Thus, the first pole of the system is determined by the filter of the adaptation algorithm and the remaining two poles by the desired closed-loop dynamics. If the poles of (3.20) are chosen to be in the open left half plane and $\lambda_a > 0$, then the eigenvalues of $\mathbf{A}(t)$ have a negative real part for all times t.

According to Theorem 1 in Section 2.5.2, $\hat{L}, \dot{\hat{L}}, \hat{R}, \dot{\hat{R}} \in \mathcal{L}_\infty$ and $\dot{\hat{L}}, \dot{\hat{R}} \in \mathcal{L}_2$. This together with the bound $0 < L_{\min} \leq \hat{L}$, which is guaranteed by the projection (3.14), implies that $\|\dot{\mathbf{A}}(t)\| \in \mathcal{L}_\infty \bigcap \mathcal{L}_2$. Thus, based on [19, Theorem 3.4.11, p. 124], the homogeneous part of (A.20) is exponentially stable.

In the next step, these results are used to establish boundedness of the system signals using the truncated exponentially weighted $\mathcal{L}_{2\delta}$ norm and the Bellman-Gronwall Lemma. Here, the procedure is similar to what is shown in [19, p.472]. Thus, by applying the Bellmann-Gronwall Lemma [19, Lemma 3.3.9, p. 103], we conclude that $m, y_a, \dot{y}_a \in \mathcal{L}_\infty$, for all times $t > 0$. Substituting into (A.18) and using $\epsilon \in \mathcal{L}_\infty$ (by Theorem 3) leads to $u_a \in \mathcal{L}_\infty$. It then follows that $\boldsymbol{\psi}, \dot{\boldsymbol{\psi}}, y, u \in \mathcal{L}_\infty$.

In the last step, the convergence of the control error will be addressed. Here, the parameter estimator properties, the boundedness of the system signals, and the plant dynamics are used to prove convergence of the control error by employing Barbalat's lemma. Given a vector signal $t \mapsto \mathbf{a}(t) \in \mathbb{R}^n$ filtered componentwise by an LTI filter with the transfer function $W(s)$, we denote by $W\{\mathbf{a}\}$ the corresponding output signal. With this notation, Lemma 2 is a corollary of the swapping lemma [19, Lemma A.1, p 774].

Using the aforementioned assumptions and theorems the estimation error equation will now be bounded. Rearranging (A.18) and taking the time derivative results in

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}(\epsilon m^2) &= \dot{u}_a - \frac{\mathrm{d}}{\mathrm{d}t}\left(\hat{L}\dot{y}_a + \hat{R}y_a\right) \\ &= \dot{u}_a - \hat{L}\ddot{y}_a - \hat{R}\dot{y}_a + \rho_1 \; , \end{aligned} \tag{A.28}$$

with the rest term $\rho_1 \in \mathcal{L}_\infty \bigcap \mathcal{L}_2$. Application of the filter

$$W = \frac{\lambda_a s}{s + \lambda_a} \tag{A.29}$$

to (A.19) yields

$$\dot{u}_a = W\left\{ \begin{bmatrix} \hat{L} & \hat{R} \end{bmatrix} \begin{bmatrix} \dot{r} \\ r \end{bmatrix} \right\} + W\left\{ \begin{bmatrix} \hat{k}_p & \hat{k}_i \end{bmatrix} \begin{bmatrix} e \\ x_c \end{bmatrix} \right\} . \tag{A.30}$$

Rewriting this expression using Lemma 2 yields

$$\dot{u}_a = \begin{bmatrix} \hat{L} & \hat{R} \end{bmatrix} W\left\{ \begin{bmatrix} \dot{r} \\ r \end{bmatrix} \right\} + \begin{bmatrix} \hat{k}_p & \hat{k}_i \end{bmatrix} W\left\{ \begin{bmatrix} e \\ x_c \end{bmatrix} \right\} + \rho_2 + \rho_3 \, , \tag{A.31}$$

with $\rho_2, \rho_3 \in \mathcal{L}_\infty \bigcap \mathcal{L}_2$. Substituting (A.31) into (A.28) and using (3.17) gives

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\epsilon m^2\right) = \hat{L} A^* \Lambda_a e + \bar{\rho}, \tag{A.32}$$

where $A^* = \frac{\mathrm{d}}{\mathrm{d}t}^2 + \alpha_1^* \frac{\mathrm{d}}{\mathrm{d}t} + \alpha_0^*$ refers to the desired pole-placement polynomial, see (3.20), and $\bar{\rho} = \sum_{i=1}^{3} \rho_i \in \mathcal{L}_\infty \bigcap \mathcal{L}_2$. Rearranging for the control error e and using the product rule yields

$$e = \frac{1}{\Lambda_a A^*} \left(\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{1}{\hat{L}} \epsilon m^2\right) + \frac{\dot{\hat{L}}}{\hat{L}^2} \epsilon m^2 - \frac{\bar{\rho}}{\hat{L}} \right) . \tag{A.33}$$

Since $\hat{L} \in \mathcal{L}_\infty$ as well as $\dot{\hat{L}}, \epsilon m^2 \in \mathcal{L}_\infty \bigcap \mathcal{L}_2$, and $A^*(s)$ is a Hurwitz polynomial by design, it follows that $e \in \mathcal{L}_\infty \bigcap \mathcal{L}_2$. Additionally, from a special case of Barbalat's lemma [19, Lemma 3.2.5, p.76], it follows that $\dot{e} \in \mathcal{L}_\infty$ and

$$\lim_{t \to \infty} e(t) = 0 \, . \tag{A.34}$$

We will now show that the parameter rates converge to zero. Eq. (A.28) can be expanded to

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\epsilon m^2\right) = \dot{u}_a - \dot{\hat{L}} \dot{y}_a - \hat{L} \ddot{y}_a - \dot{\hat{R}} y_a - \hat{R} \dot{y}_a \, . \tag{A.35}$$

Due to (3.6), $\ddot{y}_a \in \mathcal{L}_\infty$ holds, and since $\boldsymbol{\psi}, \dot{\boldsymbol{\psi}}, \hat{R}, \hat{L}, \dot{\hat{R}}, \dot{\hat{L}} \in \mathcal{L}_\infty$, it can be concluded

that $\frac{\mathrm{d}}{\mathrm{dt}}(\epsilon m^2) \in \mathcal{L}_\infty$. This together with $\epsilon m^2 \in \mathcal{L}_\infty \bigcap \mathcal{L}_2$ and the uniform continuity of (A.35) leads via Barbalat's Lemma to $\epsilon m^2 \to 0$, as $t \to \infty$. Because $m^2 \geq 1$, it can be further concluded that $\epsilon \to 0$ as $t \to \infty$. Using the structure of the estimator in (3.14), and since the gain matrix $\mathbf{P} \in \mathcal{L}_\infty$, it can be concluded that $\dot{\hat{R}}, \dot{\hat{L}} \to 0$ as $t \to \infty$. Moreover, (3.17) implies that $\dot{\hat{k}}_i \to 0$ and $\dot{\hat{k}}_p \to 0$ as $t \to \infty$. This concludes the proof.

Remark 10. *It is guaranteed that the estimation error ϵ and the plant and control parameter rates $\dot{\hat{R}}, \dot{\hat{L}}, \dot{\hat{k}}_p, \dot{\hat{k}}_i$ converge to zero. It is not guaranteed that the plant and control parameters $\hat{R}, \hat{L}, \hat{k}_p, \hat{k}_i$ will converge to the true values of $\bar{R}, \bar{L}, \bar{k}_p, \bar{k}_i$. Indeed, it is not ensured that the signals used in the estimation algorithm are persistently exciting.*

A.4 Discrete-Time Constrained Bounded-Gain Forgetting Least-Squares Algorithm

In this section, the discrete-time implementation of the constrained bounded-gain forgetting least-squares algorithm from Section 3.4.1 is summarized. We apply a time discretization for $t = kT_s$ with the sampling time T_s and $k = 1, 2, \ldots, N$. Subsequently, the index k refers to the sampling instant at time kT_s, i.e., $f_k = f(kT_s)$. For the given application, we consider box constraints of the form

$$\mathcal{S} = [L_{\min}, L_{\max}] \times [R_{\min}, R_{\max}] \,, \tag{A.36}$$

with lower limits $L_{\min}$ and $R_{\min}$ and upper limits $L_{\max}$ and $R_{\max}$. In this case, an analytical solution to the orthogonal projection of (3.12) is given by

$$\mathcal{P}_\theta(\boldsymbol{\theta}) = \begin{bmatrix} \mathcal{P}_L(\hat{L}) \\ \mathcal{P}_R(\hat{R}) \end{bmatrix} \,, \tag{A.37}$$

with

$$\mathcal{P}_L(\hat{L}_{k+1}) = \begin{cases} L_{\min} & \text{if } \hat{L}_{k+1} < L_{\min} \\ L_{\max} & \text{if } \hat{L}_{k+1} > L_{\max} \\ \hat{L}_{k+1} & \text{if } L_{\min} \leq \hat{L}_{k+1} \leq L_{\max} \end{cases} \tag{A.38}$$

and

$$\mathcal{P}_R(\hat{R}_{k+1}) = \begin{cases} R_{\min} & \text{if } \hat{R}_{k+1} < R_{\min} \\ R_{\max} & \text{if } \hat{R}_{k+1} > R_{\max} \\ \hat{R}_{k+1} & \text{if } R_{\min} \leq \hat{R}_{k+1} \leq R_{\max} \, . \end{cases} \tag{A.39}$$

Hence, the parameters are constrained by using the Goldstein-Levitin-Polyak projection algorithm, see, e.g., [77]. The discrete-time constrained bounded-gain forgetting least-squares algorithm, with $\boldsymbol{\theta}_k = \boldsymbol{\theta}(kT_s)$ and $\mathbf{P}_k = \mathbf{P}(kT_s)$, reads as [76, page 365] and [3, Chapter 3.7, page 91]

$$\mathbf{L}_k = \frac{\mathbf{P}_{k-1}\boldsymbol{\varphi}_k}{\lambda_k + \boldsymbol{\varphi}_k^{\mathrm{T}}\mathbf{P}_{k-1}\boldsymbol{\varphi}_k} \tag{A.40a}$$

$$\mathbf{P}'_k = \frac{1}{\lambda_k}\left(\mathbf{P}_{k-1} - \mathbf{L}_k\boldsymbol{\varphi}_k^{\mathrm{T}}\mathbf{P}_{k-1}\right) \tag{A.40b}$$

$$\boldsymbol{\theta}_k = \mathcal{P}_\theta\left(\boldsymbol{\theta}_{k-1} + \mathbf{L}_k\left(z_k - \boldsymbol{\varphi}_k^{\mathrm{T}}\boldsymbol{\theta}_{k-1}\right)\right) \tag{A.40c}$$

$$\mathbf{P}_k = \Pi_p\left(\boldsymbol{\theta}_k, \mathbf{P}'_k\right) \, , \tag{A.40d}$$

with the discrete-time forgetting factor

$$\lambda_k = 1 - T_s\beta_{\max}\left(1 - \frac{\|\mathbf{P}_k\|}{P_{\max}}\right) \tag{A.40e}$$

and the gain matrix projection operator

$$\Pi_p(\boldsymbol{\theta}_k, \cdot) = \begin{cases} \mathbf{P}'_k & \text{if } \boldsymbol{\theta}_k \in \mathcal{S} \\ \mathbf{P}_{k-1} & \text{otherwise} \, . \end{cases} \tag{A.40f}$$

Bibliography

[1] V. Shaferman, M. Schwegel, T. Glück, and A. Kugi, "Continuous-Time Least-Squares Forgetting Algorithms for Indirect Adaptive Control," *European Journal of Control*, vol. 62, pp. 105–112, 2021.

[2] R. M. Johnstone, R. C. Johnson, R. R. Bitmead, and Brian. O. Anderson, "Exponential convergence of recursive least squares with exponential forgetting factor," *Systems & Control Letters*, vol. 2, no. 2, pp. 77–82, 1982.

[3] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control*. New York, USA: Dover Publications, 1984.

[4] S. Sastry and M. Bodson, *Adaptive Control: Stability, Convergence and Robustness*. New Jersey, USA: Prentice Hall, 1989.

[5] H. Zengin, N. Zengin, B. Fidan, and A. Khajepour, "Blending based multiple-model adaptive control of multivariable systems with application to lateral vehicle motion control," *European Journal of Control*, vol. 58, pp. 1–10, 2021.

[6] M. Aguado-Rojas, P. Maya-Ortiz, and G. Espinosa-Pérez, "On-line estimation of switched reluctance motor parameters," *International Journal of Adaptive Control and Signal Processing*, vol. 32, no. 6, pp. 950–966, 2018.

[7] C. Jiang, H. Qian, Y. Pan, and T. Chai, "The research of superheated steam temperature control based on generalized predictive control algorithm and adaptive forgetting factor," *International Journal of Adaptive Control and Signal Processing*, vol. 34, no. 1, pp. 15–31, 2020.

[8] I. Constantin, J. Constantin, and A. Bigand, "A new kernel RLS algorithm for systems with bounded noise," *International Journal of Adaptive Control and Signal Processing*, vol. 32, no. 2, pp. 344–361, 2018.

[9] B. Bercu and V. Vázquez, "On the almost sure central limit theorem for ARX processes in adaptive tracking," *International Journal of Adaptive Control and Signal Processing*, vol. 33, no. 12, pp. 1901–1911, 2019.

[10] F. Fraccaroli, A. Peruffo, and M. Zorzi, "A new recursive least squares method with multiple forgetting schemes," in *IEEE Conference on Decision and Control*, 2015, pp. 3367–3372.

[11] R. Kulhavý and M. B. Zarrop, "On a general concept of forgetting," *International Journal of Control*, vol. 58, no. 4, pp. 905–924, 1993.

[12] S. Dong, T. Liu, and F. Chen, "Output Error Model Identification Against Unexpected Load Disturbance," *IFAC-PapersOnLine*, vol. 49, no. 7, pp. 863–868, 2016.

[13] G. Hardier, "An Extended U-D Algorithm with Multiple Forgetting Factors for RLS Estimation of Model Parameters," *IFAC-PapersOnLine*, vol. 48, no. 21, pp. 200–207, 2015.

[14] M. Beza and M. Bongiorno, "A Modified RLS Algorithm for Online Estimation of Low-Frequency Oscillations in Power Systems," *IEEE Transactions on Power Systems*, vol. 31, no. 3, pp. 1703–1714, 2016.

[15] I. Karafyllis and M. Krstic, "Adaptive certainty-equivalence control with regulation-triggered finite-time least-squares identification," *Transactions on Automatic Control*, vol. 63, no. 10, pp. 3261–3275, 2018.

[16] D. Wu, J. Song, and Y. Shen, "Variable forgetting factor identification algorithm for fault diagnosis of wind turbines," in *Chinese Control and Decision Conference*, 2016, pp. 1895–1900.

[17] M. Xu, P. Pinson, Z. Lu, Y. Qiao, and Y. Min, "Adaptive robust polynomial regression for power curve modeling with application to wind power forecasting," *Wind Energy*, vol. 19, no. 12, pp. 2321–2336, 2016.

[18] N. H. Nguyen, K. Doğançay, and E. E. Kuruoğlu, "An Iteratively Reweighted Instrumental-Variable Estimator for Robust 3-D AOA Localization in Impulsive Noise," *IEEE Transactions on Signal Processing*, vol. 67, no. 18, pp. 4795–4808, 2019.

[19] P. A. Ioannou and J. Sun, *Robust Adaptive Control*. New York, USA: Dover Publications, 2012.

[20] M. E. Salgado, G. C. Goodwin, and R. H. Middletton, "Modified least squares algorithm incorporating exponential resetting and forgetting," *International Journal of Control*, vol. 47, no. 2, pp. 477–491, 1988.

[21] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*. New Jersey, USA: Prentice-Hall, 1991.

[22] J. Li, Y. Wang, Y. Li, and W. Luo, "Reference Trajectory Modification Based on Spatial Iterative Learning for Contour Control of Two-Axis NC Systems," *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 3, pp. 1266–1275, 2020.

[23] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, Massachusetts, USA: MIT Press, 2008.

[24] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "Autoaugment: Learning Augmentation Strategies From Data," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 113–123.

[25] Z. Yang, C. Wang, Z. Zhang, and J. Li, "Mini-batch algorithms with online step size," *Knowledge-Based Systems*, vol. 165, pp. 228–240, 2019.

[26] P. Zheng, T. Askham, S. L. Brunton, J. N. Kutz, and A. Y. Aravkin, "A Unified Framework for Sparse Relaxed Regularized Regression: SR3," *IEEE Access*, vol. 7, pp. 1404–1423, 2018.

[27] L. Cao and H. Schwartz, "A directional forgetting algorithm based on the decomposition of the information matrix," *Automatica*, vol. 36, no. 11, pp. 1725–1731, 2000.

[28] S. Brüggemann and R. R. Bitmead, "Exponential convergence of recursive least squares with forgetting factor for multiple-output systems," *Automatica*, vol. 124, no. 109389, pp. 1–4, 2021.

[29] B. Lai and D. S. Bernstein, "Generalized Forgetting Recursive Least Squares: Stability and Robustness Guarantees," *http://arxiv.org/abs/2308.04259*, pp. 1–16, 2023.

[30] M. Bin, "Generalized recursive least squares: Stability, robustness, and excitation," *Systems & Control Letters*, vol. 161, pp. 1–10, 2022.

[31] J. Parkum, N. Poulsen, and J. Holst, "Recursive forgetting algorithms," *International Journal of Control*, vol. 55, no. 1, pp. 109–128, 1992.

[32] S. Gunnarsson, "Combining tracking and regularization in recursive least squares identification," in *IEEE Conference on Decision and Control*, 1996, pp. 2551–2552.

[33] H.-S. Shin and H.-I. Lee, "A New Exponential Forgetting Algorithm for Recursive Least-Squares Parameter Estimation," *https://arxiv.org/abs/2004.03910*, pp. 1–12, 2020.

[34] R. Kelly, V. Santibáñez, and A. Loría, *Control of Robot Manipulators in Joint Space*. London, England: Springer, 2005.

[35] S. Bittanti, A. J. Laub, and J. C. Willems, *The Riccati Equation*. Berlin, Germany: Springer Science & Business Media, 2012.

[36] L. Ljung, *System Identification: Theory for the User*. London, England: Prentice-Hall, 1987.

[37] L. Ljung and S. Gunnarsson, "Adaptation and tracking in system identification - A survey," *Automatica*, vol. 26, no. 1, pp. 7–21, 1990.

[38] R. Kulhavý, "Restricted Exponential Forgetting in Real-time Identification," *Automatica*, vol. 23, no. 5, pp. 589–600, 1987.

[39] A. Vahidi, A. Stefanopoulou, and H. Peng, "Recursive least squares with forgetting for online estimation of vehicle mass and road grade: Theory and experiments," *Vehicle System Dynamics*, vol. 43, no. 1, pp. 31–55, 2005.

[40] X. Zhao, L. Li, J. Song, C. Li, and X. Gao, "Linear Control of Switching Valve in Vehicle Hydraulic Control Unit Based on Sensorless Solenoid Position Estimation," *Transactions on Industrial Electronics*, vol. 63, no. 7, pp. 4073–4085, 2016.

[41] L. Zhang, L. Liu, X. Zhu, and Z. Xu, "An electric load simulator for engine camless valvetrains," *Applied Sciences*, vol. 9, no. 8, pp. 1–15, 2019.

[42] J. R. M. van Dam, B. L. J. Gyser, M. Roes, and E. A. Lomonova, "Comparison of soft-landing position control and energy minimization performance of two fluid-control solenoid valves," in *European Conference on Power Electronics and Applications*, 2018, pp. 1–9.

[43] B. Zardin, M. Borghi, G. Cillo, C. A. Rinaldini, and E. Mattarelli, "Design of two-stage On/Off cartridge valves for mobile applications," *Energy Procedia*, vol. 126, pp. 1123–1130, 2017.

[44] K. Ma, D. Sun, G. Sun, Y. Kan, and J. Shi, "Design and efficiency analysis of wet dual clutch transmission decentralised pump-controlled hydraulic system," *Mechanism and Machine Theory*, vol. 154, pp. 1–24, 2020.

[45] M. Schwegel, T. Glück, V. Shaferman, L. Zaccarian, and A. Kugi, "Adaptive Two-Degrees-of-Freedom Current Control for Solenoids: Theoretical Investigation and Practical Application," *IEEE Transactions on Control Systems Technology*, vol. 31, no. 3, pp. 1078–1091, 2023.

[46] R. Sutton, A. Barto, and R. J. Williams, "Reinforcement Learning is Direct Adaptive Optimal Control," *IEEE Control Systems Magazine*, vol. 12, no. 2, pp. 19–22, 1992.

[47] H. Whitaker, J. Yamron, and A. Kezer, *Design of Model-reference Adaptive Control Systems for Aircraft*. Massachusetts, USA: Massachusetts Institute of Technology Instrumentation Laboratory, 1958.

[48] K. J. Åström and B. Wittenmark, *Adaptive Control*, 2nd ed. Mineola, New York, USA: Dover Publications, 2008.

[49] K. S. Narendra and A. M. Annaswamy, *Stable Adaptive Systems*. Mineola, New York, USA: Dover Publications, 2005.

[50] I. D. Landau, "A survey of model reference adaptive techniques-Theory and applications," *Automatica*, vol. 10, no. 4, pp. 353–379, 1974.

[51] M. Krstić, I. Kanellakopoulos, and P. Kokotović, *Nonlinear and Adaptive Control Design*. New York, USA: Wiley, 1995.

[52] A. M. Annaswamy and A. L. Fradkov, "A historical perspective of adaptive control and learning," *Annual Reviews in Control*, vol. 52, pp. 18–41, 2021.

[53] R. Ortega, J. G. Romero, and S. Aranovskiy, "A new least squares parameter estimator for nonlinear regression equations with relaxed excitation conditions and forgetting factor," *Systems & Control Letters*, vol. 169, pp. 1–12, 2022.

[54] I. Karafyllis, M. Kontorinaki, and M. Krstic, "Adaptive Control By Regulation-Triggered Batch Least Squares," *IEEE Transactions on Automatic Control*, vol. 65, no. 7, pp. 2842–2855, 2020.

[55] W. Li and M. Krstić, "Filterless Least-Squares Based Adaptive Stochastic Continuous-Time Nonlinear Control," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 2171–2176, 2020.

[56] C. Krimpmann, G. Schoppel, I. Glowatzky, and T. Bertram, "Performance evaluation of nonlinear surfaces for sliding mode control of a hydraulic valve," in *IEEE Conference on Control Applications*, 2015, pp. 822–827.

[57] K. Laib, A. R. Meghnous, M. T. Pham, and X. Lin-Shi, "Averaged state model and sliding mode observer for on/off solenoid valve pneumatic actuators," in *American Control Conference*, 2016, pp. 4569–4574.

[58] S. Hodgson, M. Tavakoli, M. T. Pham, and A. Leleve, "Nonlinear discontinuous dynamics averaging and PWM-based sliding control of solenoid-valve pneumatic actuators," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 2, pp. 876–888, 2014.

[59] T. Braun, F. Straußberger, and J. Reuter, "State estimation for fast-switching solenoid valves: A study on practical nonlinear observers and new experimental results," in *International Conference on Methods and Models in Automation and Robotics*, 2015, pp. 862–867.

[60] P. Ioannou, *Adaptive Control Tutorial (Advances in Design and Control)*. Philadelphia, USA: Society for Industrial and Applied Mathematics, 2006.

[61] M. Bin, L. Marconi, and A. R. Teel, "Results on adaptive output regulation for linear systems by least-squares identifiers," in *IEEE Conference on Decision and Control*, 2018, pp. 1391–1396.

[62] M. S. Boroujeni, G. A. Markadeh, and J. Soltani, "Adaptive Input-output feedback linearization control of Brushless DC Motor with arbitrary current reference using Voltage Source Inverter," in *Power Electronics, Drive Systems & Technologies Conference*, 2017, pp. 537–542.

[63] G. Pillonetto, A. Carè, and M. C. Campi, "Kernel-based SPS," in *Symposium on System Identification*, 2018, pp. 31–36.

[64] Y. Sun, S. Oymak, and M. Fazel, "Finite Sample System Identification: Improved Rates and the Role of Regularization," in *Machine Learning Research*, 2020, pp. 1–10.

[65] S. Oymak and N. Ozay, "Non-asymptotic identification of LTI systems from a single trajectory," in *American Control Conference*, 2019, pp. 5655–5661.

[66] B. C. Csáji, M. C. Campi, and E. Weyer, "Sign-Perturbed Sums: A new system identification approach for constructing exact non-asymptotic confidence regions in linear regression models," *IEEE Transactions on Signal Processing*, vol. 63, no. 1, pp. 169–181, 2014.

[67] A. Carè, B. C. Csáji, M. C. Campi, and E. Weyer, "Finite-sample system identification: An overview and a new correlation method," *IEEE Control Systems Letters*, vol. 2, no. 1, pp. 61–66, 2017.

[68] M. Simchowitz, R. Boczar, and B. Recht, "Learning Linear Dynamical Systems with Semi-Parametric Least Squares," in *Conference on Learning Theory*, 2019, pp. 2714–2802.

[69] G. Chowdhary and E. Johnson, "Least squares based modification for adaptive control," in *IEEE Conference on Decision and Control*, 2010, pp. 1767–1772.

[70] M. Kanamori and K. Iwagami, "Finite-gain L2 stability for position/force control of robot manipulators with constraint uncertainties," in *International Conference on Methods and Models in Automation and Robotics*, 2015, pp. 1104–1109.

[71] C. Desoer and M. Vidyasagar, "IV - Linear Systems," in *Feedback Systems: Input–Output Properties*. Massachusetts, USA: Academic Press, 1975, pp. 56–135.

[72] B. D. O. Anderson and R. M. Johnstone, "Robust Lyapunov stability results and adaptive systems," in *IEEE Conference on Decision and Control Including the Symposium on Adaptive Processes*, 1981, pp. 510–515.

[73] A. Nagurney and D. Zhang, *Projected Dynamical Systems and Variational Inequalities with Applications*. Massachusetts, USA: Kluwer, 1996.

[74] J. Parkum, N. Poulsen, and J. Holst, "Selective Forgetting in Adaptive Procedures," *IFAC Proceedings Volumes*, vol. 23, no. 8, Part 2, pp. 137–142, 1990.

[75] J. Love, *Least Squares Identification. In: Process Automation Handbook: A Guide to Theory and Practice*. London, England: Springer, 2007.

[76] L. Ljung and T. Söderström, *Theory and Practice of Recursive Identification*. Massachusetts, USA: MIT Press, 1983.

[77] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*. Massachusetts, USA: Academic press, 1996.

[78] N. Schillreff, M. Nykolaychuk, and F. Ortmeier, "Towards High Accuracy Robot-Assisted Surgery," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 5666–5671, 2017.

[79] M. Garstenauer, C. Mittermayer, M. Reyhani-Masouleh, and M. Schwegel, "Shopfloor-Ready High Accuracy Robotics," in *International Symposium on Robotics Europe*, 2022, pp. 1–8.

[80] J. Na, Q. Chen, and X. Ren, "Chapter 6 - Adaptive Control for Manipulation Systems With Discontinuous Piecewise Parametric Friction Model," in *Adaptive Identification and Control of Uncertain Systems with Non-smooth Dynamics*. Massachusetts, USA: Academic Press, 2018, pp. 93–105.

[81] M. Schwegel and A. Kugi, "A Simple Computationally Efficient Path ILC for Industrial Robotic Manipulators," in *[Accepted] IEEE Conference on Robotics and Automation*, 2024.

[82] C. Hu, S. Lin, Z. Wang, and Y. Zhu, "Task Space Contouring Error Estimation and Precision Iterative Control of Robotic Manipulators," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7826–7833, 2022.

[83] L. Wang, T. Chai, and C. Yang, "Neural-Network-Based Contouring Control for Robotic Manipulators in Operational Space," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 4, pp. 1073–1080, 2012.

[84] F. Fujii, T. Nonomura, and T. Shiinoki, "Implementation of six degree-of-freedom high-precision robotic phantom on commercial industrial robotic manipulator," *Biomedical Physics & Engineering Express*, vol. 7, no. 5, pp. 1–8, 2021.

[85] J. Wu, B. Zhang, L. Wang, and G. Yu, "An iterative learning method for realizing accurate dynamic feedforward control of an industrial hybrid robot," *Science China Technological Sciences*, vol. 64, no. 6, pp. 1177–1188, 2021.

[86] R. Lee, L. Sun, Z. Wang, and M. Tomizuka, "Adaptive Iterative Learning Control of Robot Manipulators for Friction Compensation," *IFAC-PapersOnLine*, vol. 52, no. 15, pp. 175–180, 2019.

[87] Y.-H. Lee, S.-C. Hsu, T.-Y. Chi, Y.-Y. Du, J.-S. Hu, and T.-C. Tsao, "Industrial robot accurate trajectory generation by nested loop iterative learning control," *Mechatronics*, vol. 74, pp. 1–11, 2021.

[88] M. Hofer, L. Spannagl, and R. D'Andrea, "Iterative Learning Control for Fast and Accurate Position Tracking with an Articulated Soft Robotic Arm," in *IEEE International Conference on Intelligent Robots and Systems*, 2019, pp. 6602–6607.

[89] G. Sebastian, Z. Li, Y. Tan, and D. Oetomo, "On implementation of feedback-based PD-type iterative learning control for robotic manipulators with hard input constraints," in *IEEE International Conference on Control and Automation*, 2019, pp. 43–48.

[90] C. Liu, M. Wang, X. Li, and S. Ratchev, "Feedforward Enhancement through Iterative Learning Control for Robotic Manipulator," in *IEEE International Conference on Automation Science and Engineering*, 2021, pp. 1067–1072.

[91] G. Litak and M. I. Friswell, "Vibration in gear systems," *Chaos, Solitons & Fractals*, vol. 16, no. 5, pp. 795–800, 2003.

[92] O. Koç, G. Maeda, and J. Peters, "Optimizing the Execution of Dynamic Robot Movements With Learning Control," *IEEE Transactions on Robotics*, vol. 35, no. 4, pp. 909–924, 2019.

[93] B. Nemec, T. Petrič, and A. Ude, "Force adaptation with recursive regression Iterative Learning Controller," in *IEEE International Conference on Intelligent Robots and Systems*, 2015, pp. 2835–2841.

[94] J. Xia, Y. Li, L. Yang, and D. Huang, "Spatial Repetitive Learning Control for Trajectory Learning in Human-Robot Collaboration," in *IEEE Conference on Decision and Control*, 2019, pp. 5568–5573.

[95] N. Tanimoto, M. Sekimoto, S. Kawamura, and H. Kimura, "Generation of feedforward torque by reuse of ILC torque for three-joint robot arm in gravity," in *Conference of the Society of Instrument and Control Engineers of Japan*, 2017, pp. 927–931.

[96] S. Kawamura and N. Fukao, "A time-scale interpolation for input torque patterns obtained through learning control on constrained robot motions," in *IEEE International Conference on Robotics and Automation*, 1995, pp. 2156–2161.

[97] F. Boeren, A. Bareja, T. Kok, and T. Oomen, "Frequency-Domain ILC Approach for Repeating and Varying Tasks: With Application to Semiconductor Bonding Equipment," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 6, pp. 2716–2727, 2016.

[98] J. Liu, X. Dong, D. Huang, and M. Yu, "Composite Energy Function-Based Spatial Iterative Learning Control in Motion Systems," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 5, pp. 1834–1841, 2018.

[99] F. H. Kong, A. M. Boudali, and I. R. Manchester, "Phase-indexed ILC for control of underactuated walking robots," in *IEEE Conference on Control Applications*, 2015, pp. 1467–1472.

[100] Y. Wang, F. Gao, and F. J. Doyle, "Survey on iterative learning control, repetitive control, and run-to-run control," *Journal of Process Control*, vol. 19, no. 10, pp. 1589–1600, 2009.

[101] R. W. Longman, "Iterative learning control and repetitive control for engineering practice," *International Journal of Control*, vol. 73, no. 10, pp. 930–954, 2000.

[102] Z. Bien and J.-X. Xu, *Iterative Learning Control - Analysis, Design, Integration and Applications*. Massachusetts, USA: Springer, 1998.

[103] W. Messner, R. Horowitz, W.-W. Kao, and M. Boals, "A new adaptive learning rule," *IEEE Transactions on Automatic Control*, vol. 36, no. 2, pp. 188–197, 1991.