University of Rhode Island

DigitalCommons@URI

Open Access Master's Theses

2020

FRICITION COMPENSATION USING LINEAR CONTROL METHODS

Florian Meiners

University of Rhode Island, florian.meiners@coach-zu-recht.de

Follow this and additional works at: <https://digitalcommons.uri.edu/theses>

Recommended Citation

Meiners, Florian, "FRICITION COMPENSATION USING LINEAR CONTROL METHODS" (2020). *Open Access Master's Theses*. Paper 1870.

<https://digitalcommons.uri.edu/theses/1870>

This Thesis is brought to you for free and open access by DigitalCommons@URI. It has been accepted for inclusion in Open Access Master's Theses by an authorized administrator of DigitalCommons@URI. For more information, please contact digitalcommons-group@uri.edu.

FRICTION COMPENSATION USING LINEAR CONTROL METHODS

BY

FLORIAN MEINERS

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

ELECTRICAL ENGINEERING

UNIVERSITY OF RHODE ISLAND

2020

MASTER OF SCIENCE THESIS
OF
FLORIAN MEINERS

APPROVED:

Thesis Committee:

Major Professor Richard Vaccaro

Paolo Stegagno

Musa Jouaneh

Brenton DeBoef

DEAN OF THE GRADUATE SCHOOL

UNIVERSITY OF RHODE ISLAND

2020

ABSTRACT

Friction is one of the most frequently appearing sources of problems in mechanical systems. While several controller designs exist that successfully mitigate or compensate the friction effects in an application, they usually rely on a thoroughly identified model. The focus of this work is on verifying the suitability of a linear dual-loop controller for friction compensation in an arbitrary system. It does not require detailed estimation of a friction model for the design. The controller is complemented by a Luenberger observer, to improve the quality of state feedback.

Based on different model reduction approaches, the control architecture is effectively applied to a number of system descriptions employing various friction models. Through simulation, the versatility of the idea is reinforced. By comparing the control quality with that of approbated designs taken from recent literature, we show that the concept is competitive with model-based friction compensators. The advance with respect to robustness and disturbance rejection is shown. In preparation for an implementation, the control algorithm is discretized. To predict the practical performance of the controller, it is shown that added noise does not deteriorate the excellent results in the digital realm.

ACKNOWLEDGMENTS

I would like to pay my special gratitude to my thesis supervisor, Richard Vaccaro. His heartfelt interest and enthusiasm for my work encouraged me to accomplish this final achievement of my academic career. Our discussions throughout my research were a constant inspiration for improvements and motivated me to strive for excellence. I am deeply thankful for this memorable cooperation.

For my thesis committee members, Paolo Stegagno and Musa Jouaneh, I want to express my sincere appreciation. Their unhesitant willingness to be a part of this process allowed for a fluent commencement of my work and their helpful suggestions were invaluable.

My parents deserve my most affectionate gratefulness. Not only by constantly supporting me throughout my life, but also by motivating me to pursue my studies in Germany and Rhode Island, they allowed me to accomplish something that fills me with pride and, most importantly, joy.

Finally, I want to thank my friends, family and girlfriend for their warmhearted and reliable backing. In these unprecedented times and throughout personal hardship they were with me and provided encouragement.

Contents

ABSTRACT	ii
ACKNOWLEDGMENTS	iii
Contents	iv
List of Figures	vi
Chapter	
1 Current Status of Technology	1
1.1 Friction Phenomena and their Model Description	1
1.2 Friction Compensation	5
2 Idea and Design Procedure of the Control Strategy	10
3 Applying the Controller Design to Other Systems	20
3.1 Different Classes of Systems under Consideration	20
3.2 Conditions for the Applicability	21
3.2.1 Friction Modeling	21
3.2.2 Controllability	22
3.3 Reduction of the LuGre Model	27
4 Application to Specific System Models	36
4.1 Simple Single-Mass System	36
4.1.1 Characteristic Curve-Based Model	37
4.1.2 LuGre Friction Model	45
4.2 More Complex System Dynamics	51

	Page
4.3 Plants with Position Feedback	56
4.4 Stability of the Control Loop	63
5 Comparison with Nonlinear Control Concepts	65
5.1 Output Feedback Controller with Nonlinear Friction Compensator	65
5.1.1 Controller Design	65
5.1.2 Comparison with the Novel Control Architecture	70
5.2 Friction Observer and State Feedback Control	78
5.2.1 Controller Design	78
5.2.2 Comparison with the Novel Control Architecture	81
5.3 Input-Output Linearization Controller	86
6 Discretization of the Control Algorithm	97
6.1 Design of the Digital Controller	97
6.2 Performance of the Discrete Controller	104
6.2.1 Impact of Sampling on the Control Behavior	106
6.2.2 Influence of Noise on the Tracking Quality	111
7 Summary and Conclusion	115
8 Future Work on the Topic	117
LIST OF REFERENCES	120
APPENDIX	
A Appendix A: Structural Analysis of Invariant Zeros	123
B Appendix B: MATLAB Code for the Controller Design	126
BIBLIOGRAPHY	131

List of Figures

Figure		Page
1.1	Friction Force Depending on the Velocity	2
1.2	Operating Points of the LuGre Friction Model: Sticking (left) and Motion (right)	5
1.3	General Structure of a Friction Observer	6
2.1	Plot of the Nonlinear Friction Effect in [1]	11
2.2	Inner Loop of the Nested Control Structure	12
2.3	Outer Loop of the Nested Control Structure	13
2.4	Control System with Observer in [1]	15
3.1	Pole Positions of the Linearized LuGre Model Depending on the System Mass	31
4.1	Single-Mass System with Nonlinear Friction	37
4.2	Output Progression (orange) and Reference Output (blue) for the Model in [2]	43
4.3	Tracking Error without Observer (Model in [2])	44
4.4	Tracking Error Using the Observer (Model in [2])	45
4.5	Output Progression (orange) and Reference Output (blue) for the System with LuGre Model in [3]	49
4.6	Tracking Error without Observer (Model in [3])	50
4.7	Tracking Error Using the Observer (Model in [3])	51
4.8	Tracking Error without Observer (Model in [4])	54
4.9	Tracking Error Using the Observer (Model in [4])	55
4.10	Output Progression (orange) and Reference Output (blue) for the LuGre Model in [4]	55

Figure		Page
4.11	Output Progression (blue) and Reference Output (orange) for the Model in [5]	61
4.12	Output Progression (blue) and Reference Output (orange) without Bias Torque (Model in [5])	62
4.13	Output Progression (blue) and Reference Output (orange) with Subpar Settling Time (Model in [5])	63
5.1	Controller Structure Proposed in [6]	68
5.2	Output Progression for the Controller in 5.11	71
5.3	Difference between Actual System Output and Unperturbed Reference System (Model in [6])	72
5.4	Tracking Error in System with Linear Controller (Model in [6])	73
5.5	Tracking Performance of Nonlinear Controller in Eq. 5.11 Subject to Parameter Uncertainties (blue) Compared with Reference (orange)	75
5.6	Nonlinear Controller Subject to Disturbance (blue) Compared with Reference (orange) (Model in [6])	76
5.7	Tracking Error of Linear Controller Subject to Disturbance (Model in [6])	77
5.8	Tracking Error of Linear Controller Compared to Reference Output (Model in [7])	82
5.9	Tracking Error of Observer-Based Controller with Parameter Uncertainties (Model in [7])	84
5.10	True Output Value with Observer-Based Controller (blue) Subject to Disturbance and Reference Output (orange) (Model in [7])	85
5.11	True Output Value with Linear Controller (blue) Subject to Disturbance and Reference Output (orange) (Model in [7]) . . .	86
5.12	Output Progression with Linearizing Controller (blue) Subject to Modeling Uncertainties and Reference Output (orange) . . .	95
6.1	Discrete Control System with Observer	105

Figure		Page
6.2	Output Progression (blue) and Continuous Reference Output (orange) for System with Discrete Controller	107
6.3	Output Progression (blue) and Continuous Reference Output (orange) for System with Discrete Controller and Shorter Sampling Interval	108
6.4	Control Variable Progression for Discrete System, $T_s = 0.2\text{ s}$. .	109
6.5	Control Variable Progression for Discrete System, $T_s = 0.1\text{ s}$. .	109
6.6	Output Progression (blue) and Continuous Reference Output (orange) for Disturbed and Uncertain System	110
6.7	Output Progression (blue) and Continuous Reference Output (orange) for the System Subject to Noise	112
6.8	Output Progression (blue) and Continuous Reference Output (orange) for the System Subject to Noise of Higher Amplitude .	113
A.1	Structural Graph of Linearized Single-Mass System with LuGre Friction Model	124

CHAPTER 1

Current Status of Technology

1.1 Friction Phenomena and their Model Description

Friction effects are among the most commonly encountered nonlinearities in mechanical dynamic systems. The treatment, mitigation and compensation of these effects has gained a lot of attention in research over the last decades. Not only the description of friction in the form of models but also the measures of coping with friction as a disruptive nonlinearity in system applications, have evolved.

While, generally speaking, the term friction may refer to phenomena that involve fluids, in the context of classical mechanical systems theory it describes the force counteracting the relative movement of two solid bodies whose surfaces are in contact. This force depends on various aspects like the condition of the contact surface, the mass of the moving objects or the relative velocity (see [8]). There are different friction effects that can be distinguished.

A fundamental insight characterized by Richard Stribeck almost 120 years ago, is that the occurring friction force at a constant relative velocity is highly nonlinear for small velocities. While for relative movements close to the standstill the impact of friction is very large and for higher velocities it depends almost linearly on the velocity (viscous friction), there is a very distinct recession of the friction force for small velocities. This is referred to as the Stribeck effect (see [9]).

Over the years, dynamic aspects of friction have been described in more and more detail. The most obvious friction phenomenon encountered in any practical application is that the force required to overcome the standstill and start moving is much higher than the force that has to be applied to maintain a constant velocity when sliding. Ernest Rabinowicz investigated this circumstance in 1951 and discovered that the friction force reaches its maximum for small relative displace-

Figure 1.1: Friction Force Depending on the Velocity

ments (see [10]). This is known as static friction or sticking and it sparked the idea of friction being modeled as a function of the relative position of the bodies. The subsequent developments in research led to the evolution of a number of friction models with varying focuses on the emulation of certain phenomena.

All of the first models were only static and tried to describe the basic dependence of the friction force on the velocity or the displacement. These models focus on depicting the aforementioned effects and to this end employ a characteristic curve, which usually captures the relationship between friction force and velocity. This characteristic curve imposes a nonlinearity on the model of a mechanical system. An example for such a curve is shown in Figure 1.1. It illustrates the Stribeck effect as well as the high force to overcome static friction and the linear behavior at higher velocities.

Often the decisive factor for the improvement of a friction model is the accuracy with which the nonlinear effect of friction in a system is described. Since static models fail to capture the intricacies of friction phenomena in detail, dynamic characterization of friction has shifted to the center of attention. The model developed by Phil Dahl in 1968 laid the foundation for future dynamic models (see [11]). It

has the general form

$$\frac{dF}{dx} = \sigma \left(1 - \frac{F}{F_c} \text{sgn}(v) \right)^\alpha. \quad (1.1)$$

Its main advantage over previous models is that it takes the surface deformation of the bodies subject to friction into account, an idea that was inspired by the characteristics of elastic solids. This observation allows for the dynamics to accurately represent dynamic friction effects not only when sticking is predominant. In particular, hysteresis or frictional lag can be described, a phenomenon that occurs because the friction force changes at different rates depending on whether the velocity decreases or increases.

Obviously, the Dahl model is dependent on the relative displacement and only the sign of the velocity. This property, which is also referred to as rate independence, makes it impossible to include the Stribeck effect or the large force necessary to overcome sticking, in the model. Accordingly, it was refined and modified over the years. There are two more recent types of models that are descendants of the Dahl approach.

The models developed by Bliman and Sorine in the nineties essentially combine multiple conventional Dahl models with different parameters to capture the static friction and Stribeck phenomenon next to the effects of surface deformation (see [12]). Their dynamic nature covers hysteretic phenomena of friction as well. Another idea focuses on the idea of the deformed surface being modeled as an elastic solid.

Before sliding between two bodies can occur and actual friction effects are relevant, the force is treated as the force acting on a spring under tension. The respective model developed by DeWit, Olsson, Åström and Lischinsky is known as the LuGre friction model and it is widely used in engineering and science (see [13]). Its basic concept is to think of the material surface as if it was covered with fine

bristles, which act like springs. The deflection z of these springs can be described by the nonlinear differential equation

$$\dot{z} = v - \frac{|v|}{g(v)}z, \quad (1.2)$$

where $g(v)$ is used to describe the complicated effects at low velocities:

$$g(v) = \frac{1}{\sigma_0} \left(F_C + (F_S - F_C) e^{-\left(\frac{v}{v_s}\right)^2} \right). \quad (1.3)$$

Therein F_C is the Coulomb force, that models the friction force present for the bodies in relative motion. F_S describes the static friction, the force that has to be overcome for the relative motion to transcend from surface deformation to sliding. Lastly v_s is the Stribeck velocity, the relative velocity for which the influence of the Stribeck effect is the strongest. The friction force then can be calculated as follows:

$$F_f = \sigma_0 z + \sigma_1 \dot{z} + \sigma_2 v. \quad (1.4)$$

The parameter σ_0 is the spring stiffness, σ_1 their damping coefficient. σ_2 models viscous friction for the case of sliding. In the illustration in Figure 1.2 the two different states that the model wants to reproduce are depicted. Sticking happens as long as the relative displacement and the rate of change of the force is small enough to mainly deform the material, whereas sliding, and therefore viscous friction, happens if the breakaway force is overcome. In the first case, the force results from the deflection of the bristles that act like springs. Once the relative velocity is high enough to cause sliding, the actual friction effect entails the counteracting force.

One of the main benefits of the LuGre model in the context of engineering and control in particular, is that the model is relatively simple while covering most effects that are crucial for high precision feedback control. Those include the

Figure 1.2: Operating Points of the LuGre Friction Model: Sticking (left) and Motion (right)

Stribeck effect and the breakaway force at low velocities, as well as the hysteretic behavior of systems subject to friction. For higher velocities the model captures viscous friction too (see [13]). An excellent overview of friction effects and the development of friction models is provided in [8].

Obviously, there are a number of different phenomena that are covered by the general term friction. All these phenomena have a more or less substantial impact on the proper operation of mechanical engineering applications. Particularly when high precision in control systems is required, as it is often crucial in the context of positioning and tracking, the nonlinearity that arises from the presence of friction calls for intricate measures to deal with the effects described above.

1.2 Friction Compensation

There are numerous different approaches to the problem of friction compensation in feedback control. While all of them strive to reduce the disturbing influence of the nonlinearity on the behavior of the control circuit, they do exhibit differences depending on the respective focus on robustness, efficiency, accuracy and the like. In most cases, the goal is to mitigate the influence of friction or sometimes even suppress it to an extent where the controlled system can be treated as linear. These objectives translate, for example, to the requirement of a small steady-state error or the prevention of limit cycles in dynamic terms.

A very first control strategy that has been used for almost 80 years and in

Figure 1.3: General Structure of a Friction Observer

modified forms is still encountered to this day, is the dither (see [8]). A dither is a high frequency oscillation that is applied to the controlled plant to “knock it loose”. The intention of this is to overcome the static friction level at any given time of a movement, so the actual motion of the system is only governed by viscous friction, which is linear as a good approximation. Besides the obvious problem of unpredictability and the energy consumption of the dither signal, there are more flaws to this approach, like the amount of modification to the system or acoustic noise, which make it rather unattractive for many applications.

Some of the more recent methods focus more on the actual model that describes the friction effects. Above all, there are friction observers or, more precisely, nonlinear friction observers (see [7] and [13],[14],[15]). Strictly speaking, an observer is not a controller. However, since a number of friction compensation approaches employ an observer, the concept is outlined here.

Various observer designs have been proposed throughout the years and even though they are wide-ranging in structure and determine the friction estimate in different ways, they all share the same idea. It is shown in Figure 1.3.

Analogously to the traditional idea of observers, they simulate the friction behavior based on the plant input and output. As a result, the observer estimates

the current value of the friction force to be used in the control procedure. The necessity of this simulation-based approach becomes particularly apparent when dealing with dynamic friction models like the LuGre model. Since the additional dynamic state that describes the surface deflection is intrinsically impossible to measure, a simulation is the only way of getting a grasp of its magnitude.

For friction observers that are based on a dynamic model, the estimation of this state is paramount. When recapitulating the LuGre model, we immediately notice that the nonlinearity directly affects the bristle deformation z . Therefore, if this state is to be calculated, the observer is consequently nonlinear itself, which may entail common problems with nonlinear systems, like difficulty of stability analysis and controller design, limit cycles or unpredictable behavior.

The development of modern computers gave rise to a new class of nonlinear controllers, which implement more complicated algorithms with a higher demand on fast and precise computation. Although nonlinear controllers are still rather exotic in most fields, there are several examples of nonlinear design strategies being applied for friction compensation, like for example feedback linearization (see [4]).

The idea here is to transform the system equations in a way that allows for a nonlinear controller to be created that achieves the closed-loop behavior of a linear system. That controller naturally has to be nonlinear itself. While the linear overall system has the obvious benefit of being suitable for a large number of well studied control approaches and is easy to predict and assess, the technique is prone to errors. Imprecise parameter evaluation or an inaccurate model can quickly lead to undesired behavior or even instability (see [16]).

Another nonlinear control procedure is called backstepping (see [17]). It is based on systems in strict-feedback form, namely

$$\dot{x}_1 = f_1(x_1) + h_1(x_1)x_2,$$

$$\begin{aligned}
\dot{x}_2 &= f_2(x_1, x_2) + h_2(x_1, x_2)x_3, \\
\dot{x}_3 &= f_3(x_1, x_2, x_3) + h_3(x_1, x_2, x_3)x_4, \dots, \\
\dot{x}_k &= f_k(x_1, x_2, \dots, x_k) + h_k(x_1, x_2, \dots, x_k)u.
\end{aligned} \tag{1.5}$$

Starting from the first equation and treating x_2 as its input, a nonlinear controller is designed that yields an asymptotically stable equilibrium for the respective subsystem. After plugging in the stabilizing input, this procedure is repeated from the inside out, hence the name backstepping, until $\dot{x}_k = f_k(x_1, x_2, \dots, x_k) + h_k(x_1, x_2, \dots, x_k)u$ is stabilized. The advantage of this technique is the systematic design procedure, while the drawback is the limited predictability of, and influence on the control quality (see [16]).

A large class of methods is united under the name adaptive control, where the general concept is to modify the controller parameters or even its structure based on the current circumstances. Either due to unknown disturbances or to inaccurate modeling and parameter estimation, the initial controller values may be adjusted during operation. Different reasons can cause a change of plant parameters over time, like mechanical wear or temperature variation, which can also be alleviated by adapting the controller accordingly.

For the purpose of friction compensation, the most common use of adaptive control is made in the form of an algorithm with variable parameters that are updated online based on an observer estimation of the friction model parameters. The strategy bypasses the need of a time intensive and laborious experimental determination of these values. Examples for adaptive friction control can be found in [2], [18], [19] and [20].

The last category of design methods that shall be discussed here uses machine learning procedures to approach the matter. Rather than trying to eradicate friction completely based on a fixed model, neural networks can be used to identify

the friction effects. In [21] and [22] exemplary procedures are presented. The general idea of the control algorithm yields a similar feedback structure as in the case of a conventional controller, where the reference value and plant output are fed to the controller to calculate the control signal. However, instead of having predetermined values for the control algorithm, these values are established and updated based on the input and output of the closed loop during operation.

All control methods discussed above have proven to yield satisfactory if not outstanding performance for the purpose of friction compensation. Especially for the respective situations which the controllers are designed for, be it robustness in the presence of parameter uncertainties, high precision requirements or disturbance rejection, multiple approaches exist, which successfully deal with unwanted friction effects in mechanical systems.

However, there are some gaps in the theory of friction compensation control. One fundamental issue with most technologies is that they intrinsically rely on a friction model to work, since this model is used in the design of observers or nonlinear control methods. No model is capable of describing the complicated phenomena of friction perfectly and there will always be differences between the actual physical behavior of a system and its mathematical description. Discrepancies like that may be fatal for the proper functioning of a control system or at least cause a smaller or larger deterioration of quality.

Furthermore, there does not seem to be a generally applicable theory for friction compensation yet, which is capable of dealing with the nonlinearity in an arbitrary physical plant. The goal of this work is therefore to come up with a control structure that compensates friction in any mechanical systems and that is not based on rigorous modeling of friction effects.

CHAPTER 2

Idea and Design Procedure of the Control Strategy

As we discussed in Chapter 1, there are some fundamental flaws to model-based friction compensation strategies. Almost no control algorithm works with an entirely unknown plant model and even nonlinear subsystems and processes like friction have to be approximated to enable the controller design. However, the amount of estimation and experimental system identification for nonlinear observer topologies and similar control approaches may involve a lot of simplification and guesswork and is never immune to errors.

Take, for example, a nonlinear observer that is based on the LuGre friction model. For the design of such an observer all parameters of the dynamic model typically have to be determined and its simplest form,

$$\begin{aligned}\dot{z} &= v - \frac{|v|}{g(v)}z \\ g(v) &= \frac{1}{\sigma_0} \left(F_C + (F_C - F_S) e^{-\left(\frac{v}{v_s}\right)^2} \right) \\ F &= \sigma_0 z + \sigma_1 \dot{z} + \sigma_2 v,\end{aligned}\tag{2.1}$$

already requires the estimation of six different values, namely the Coulomb force and sticking force, the Stribeck velocity, the bristle stiffness and damping and the viscous friction coefficient. Although there are methods for the identification of those parameters as in [23], [24] and [25], the values are subject to a number of external influences, can change over time and are often state-dependent. The more an applied control strategy relies on a thoroughly identified friction model, the bigger of an obstacle this poses for the performance or even stability of the closed loop control.

Figure 2.1: Plot of the Nonlinear Friction Effect in [1]

It is therefore desirable to develop a control method to enable friction compensation without such a detailed knowledge about a friction model. This is the main objective of the present work. The central idea of the approach studied throughout this thesis was first considered in [1] to increase accuracy in a specific second-order tracking control system. In particular, the system that was dealt with is a belt driven translational positioning system with the control affine nonlinear state space description

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= -F(x_2) + \beta u,\end{aligned}\tag{2.2}$$

where x_1 is the position, x_2 is the velocity and $F(x_2)$ describes the velocity-dependent nonlinear friction force. It can in turn be visualized by the characteristic curve in Figure 2.1.

As the control architecture a structure similar to cascade control, with an inner velocity loop and an outer position loop, is chosen. However, the construction is referred to as “nested”, because rather than being cascaded the velocity loop is embedded in the position loop and the velocity state and the controller states of

Figure 2.2: Inner Loop of the Nested Control Structure

the inner loop are used for state feedback control in the outer loop. In the physical system, the simple single-mass system in Eq. 2.2, its nonlinearity is approximated by a linear viscous friction coefficient:

$$\dot{v} = -\alpha v + \beta u. \quad (2.3)$$

This gives the inner velocity loop the structure illustrated in Figure 2.2. The inner loop is subsequently complemented by an integrator to receive the position signal.

Summarizing the closed inner loop and the position integrator, the outer loop takes the form of the block diagram illustrated in Figure 2.3, wherein the state space representation of the closed inner loop is

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 0 & -(\alpha + \beta k_{1v}) & \beta k_{2v} \\ 0 & -b_a & A_a \end{bmatrix}}_{\mathbf{A}_1} \mathbf{x}_1 + \underbrace{\begin{bmatrix} 0 \\ 0 \\ b_a \end{bmatrix}}_{\mathbf{b}_1} u_1 \\ y &= \underbrace{\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}}_{\mathbf{c}_1} \mathbf{x}_1. \end{aligned} \quad (2.4)$$

As can be seen from the block diagrams, both loops contain a state feedback controller and an additional dynamic part for output feedback. They are designed based on the linearized plant model in Eq. 2.3 via pole placement. This is the procedure we later want to apply to the control of an arbitrary mechanical system subject to friction.

Figure 2.3: Outer Loop of the Nested Control Structure

As the additional dynamic parts for both loops a simple integrator is used. This gives the additional dynamic blocks the mathematical form

$$\dot{x}_a = \underbrace{0}_{A_a} x_a + \underbrace{1}_{b_a} e. \quad (2.5)$$

To facilitate pole placement for both the inner and the outer loop, a design model is constructed. For this purpose, the additional dynamic state of the controller is included in the state vector, yielding the overall open-loop system description

$$\mathbf{A}_d = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ b_a \mathbf{c} & A_a \end{bmatrix}, \quad \mathbf{b}_d = \begin{bmatrix} \mathbf{b} \\ 0 \end{bmatrix}, \quad (2.6)$$

which is used to place the poles of the matrix $\mathbf{A}_d - \mathbf{b}_d \mathbf{K}_d$. $(\mathbf{A}, \mathbf{b}, \mathbf{c})$ therein functions as a placeholder for either the plant or the augmented inner loop, $(\mathbf{A}_1, \mathbf{b}_1, \mathbf{c}_1)$. The resulting feedback vector $\mathbf{K}_d$ then only has to be broken up into the state feedback vector $\mathbf{k}_1$ and the output matrix $\mathbf{k}_2$ of the additional dynamic block. In our case the latter is just a scalar, since the additional dynamics are of first order, therefore k_2 or k_{2v} .

Applying only this linearization-based control design to the friction compensation problem results in relatively poor performance, the obvious reason being that a linear dependence of the friction force can not describe the complex nature

of the occurring phenomena properly. The novelty of the strategy presented in [1] consists of the addition of a linear observer to improve the control quality. Its rationale is as follows.

If in a mechanical feedback control system subject to friction the friction effect is assumed to be linear, the design procedure is based on a model that differs from reality. This discrepancy will cause the closed loop to behave unexpectedly when the controller is applied in the physical plant. The reason for that is that when the output, typically the position, is compared to the reference input, the controller will try to apply a control signal to the plant which is substantially too small to overcome the high sticking force, because linearity is assumed.

The difference between reference and output will consequently start to build up, in an effort to overcome sticking. Once the system “breaks free”, the force exerted on the plant by the controller is too large, which entails an undesired acceleration. The control deviation has to be decomposed, an effect which is especially tragic if the controller contains an integrator. This process is encountered at low velocities, since nonlinear friction effects are predominant in this region of operation.

Recapitulating the additional dynamic part of the inner loop control structure discussed above, we remember that it contains an integrator. Despite its excellent influence on the dynamic behavior of the closed loop, it does introduce the downside of having a memory. It accumulates the high control deviation, causing the dynamic state x_a to build up, and applies a force to the plant which is larger than necessary. In the control system described above, the dynamic state of the velocity control loop is used for state feedback as it is, resulting in the aforementioned problem. The proposed linear observer is now used to suppress this effect.

It is a Luenberger observer, which is designed based on the linearized system

Figure 2.4: Control System with Observer in [1]

description and which provides a simulation-based estimate of the additional dynamic state x_a to be fed to the state feedback controller. Since it is based on the assumption of a viscous friction force, which linearly depends on the velocity, the estimate $\hat{x}_a$ will be closer to the actually required magnitude of x_a . This mitigates the influence of the nonlinear friction phenomenon on the control system by inhibiting the impact of the drastically increased integrator state on state feedback. The control system with observer is shown in Figure 2.4, where the observer is simply designed based on the linear design model of the inner loop,

$$\begin{aligned} \begin{bmatrix} \dot{x}_2 \\ \dot{x}_a \end{bmatrix} &= \underbrace{\begin{bmatrix} -(\alpha + \beta k_{1v}) & \beta k_{2v} \\ -b_a & A_a \end{bmatrix}}_{\mathbf{A}_o} \begin{bmatrix} x_2 \\ x_a \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ b_a \end{bmatrix}}_{\mathbf{b}_o} u \\ y &= \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_{\mathbf{c}_o} \begin{bmatrix} x_2 \\ x_a \end{bmatrix}, \end{aligned} \quad (2.7)$$

by performing pole placement of the matrix $\mathbf{A}_o - \mathbf{L}\mathbf{c}_o$. As the state space representation of the observer

$$\begin{aligned} \dot{\hat{\mathbf{x}}} &= (\mathbf{A}_o - \mathbf{L}\mathbf{c}_o) \hat{\mathbf{x}} + \mathbf{b}_o u + \mathbf{L}x_2 \\ \hat{x}_a &= \begin{bmatrix} 0 & 1 \end{bmatrix} \hat{\mathbf{x}} \end{aligned} \quad (2.8)$$

follows, with the observer matrix $\mathbf{L}$. The paper also proposes a reduced observer, which only simulates the integrator state. Its design procedure follows the standard methodology of a reduced Luenberger observer and can, for example, be found in [26].

Rather than simulating the complete inner loop, one could argue that including the velocity is in fact unnecessary, since its estimate is not used anyway. This rationale is the onset of the theory of reduced-order observers. Starting from the linear state space model $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ the output matrix is partitioned based on a division of the state vector $\mathbf{x}$ into the states $\mathbf{x}_2$ to be simulated by the observer and the remaining states $\mathbf{x}_1$. This gives us the form

$$\mathbf{y} = [\mathbf{C}_1 \quad \mathbf{C}_2] \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \quad (2.9)$$

and, in addition to that, the regular coordinate transformation

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{C}_1 & \mathbf{C}_2 \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \triangleq \mathbf{T}\mathbf{x}. \quad (2.10)$$

By means of this transformation the system is transferred to the form

$$\begin{aligned} \begin{bmatrix} \dot{\mathbf{y}} \\ \dot{\mathbf{x}}_2 \end{bmatrix} &= \underbrace{\begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix}}_{\mathbf{TAT}^{-1}} \begin{bmatrix} \mathbf{y} \\ \mathbf{x}_2 \end{bmatrix} + \underbrace{\begin{bmatrix} \mathbf{B}_1 \\ \mathbf{B}_2 \end{bmatrix}}_{\mathbf{TB}} \mathbf{u} \\ \mathbf{y} &= \underbrace{\begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix}}_{\mathbf{CT}^{-1}} \begin{bmatrix} \mathbf{y} \\ \mathbf{x}_2 \end{bmatrix}, \end{aligned} \quad (2.11)$$

which is the basis for the observer design. Equation 2.11 can be rewritten as

$$\begin{aligned} \dot{\mathbf{x}}_2 &= \mathbf{A}_{22}\mathbf{x}_2 + \mathbf{A}_{21}\mathbf{y} + \mathbf{B}_2\mathbf{u} \\ \dot{\mathbf{y}} &= \mathbf{A}_{11}\mathbf{y} + \mathbf{A}_{12}\mathbf{x}_2 + \mathbf{B}_1\mathbf{u}. \end{aligned} \quad (2.12)$$

With the definition of the virtual variables

$$\mathbf{u}_r \triangleq \mathbf{A}_{21}\mathbf{y} + \mathbf{B}_2\mathbf{u} \quad \text{and} \quad \mathbf{y}_r \triangleq \dot{\mathbf{y}} - \mathbf{A}_{11}\mathbf{y} - \mathbf{B}_1\mathbf{u} \quad (2.13)$$

and the matrices $\mathbf{A}_r \triangleq \mathbf{A}_{22}$, $\mathbf{B}_r \triangleq \mathbf{I}$ and $\mathbf{C}_r \triangleq \mathbf{A}_{12}$ we can formulate the state space model

$$\begin{aligned}\dot{\mathbf{x}}_2 &= \mathbf{A}_r \mathbf{x}_2 + \mathbf{B}_r \mathbf{u} \\ \mathbf{y}_r &= \mathbf{C}_r \mathbf{x}_2\end{aligned}\tag{2.14}$$

as the system to be simulated by the reduced observer. Designing a Luenberger observer for the system in Eq. 2.14 under consideration of the definitions in Eq. 2.13 yields the equation

$$\dot{\hat{\mathbf{x}}}_2 = (\mathbf{A}_{22} - \mathbf{L}\mathbf{A}_{12}) \hat{\mathbf{x}}_2 + (\mathbf{A}_{12}\mathbf{y} + \mathbf{B}_2\mathbf{u}) + \mathbf{L}(\dot{\mathbf{y}} - \mathbf{A}_{11}\mathbf{y} - \mathbf{B}_1\mathbf{u}).\tag{2.15}$$

The final step consists of the construction of the observer state vector $\mathbf{x}_b \triangleq \hat{\mathbf{x}}_2 - \mathbf{L}\mathbf{y}$, which gives us the state space representation of the reduced observer. It has the form

$$\dot{\mathbf{x}}_b = (\mathbf{A}_{22} - \mathbf{L}\mathbf{A}_{12}) \mathbf{x}_b + (\mathbf{B}_2 - \mathbf{L}\mathbf{B}_1) \mathbf{u} + ((\mathbf{A}_{22} - \mathbf{L}\mathbf{A}_{12})\mathbf{L} + \mathbf{A}_{21} - \mathbf{L}\mathbf{A}_{11}) \mathbf{y}\tag{2.16}$$

and the estimated states can be reconstructed via

$$\hat{\mathbf{x}}_2 = \mathbf{x}_b + \mathbf{L}\mathbf{y}.\tag{2.17}$$

Identically to the case of the full state observer the dynamics of the reduced observer are influenced by modifying the poles of the matrix $(\mathbf{A}_{22} - \mathbf{L}\mathbf{A}_{12})$. This is done by performing pole placement or equivalent procedures.

What looks complicated at first glance is genuinely easily comprehensible, if the design procedure is thoroughly studied step by step. The equations of the reduced observer are straight-forward to implement and in this particular case also simple to derive. Our closed inner loop in Eq. 2.7 does not require transformation, because the states are already in the correct order. From the matrices of this state

space model the partitioning

$$\begin{aligned} A_{11} &= -(\alpha + k_{1v}), A_{12} = \alpha k_{2v}, A_{21} = -b_a, A_{22} = A_a, \\ B_1 &= 0, B_2 = b_a, C_1 = 1 \text{ and } C_2 = 0 \end{aligned} \quad (2.18)$$

follows immediately and concludes the design of the reduced observer upon plugging into Eq. 2.16 and specifying the observer dynamics via pole placement. Note that, since we are dealing with a first-order reduced observer, all matrices are actually scalar values.

By use of the nested control architecture and the linear observer, the tracking accuracy not only in simulation, but also during operation in the physical system were improved. In a direct comparison with a linear reference model, the peak error was significantly reduced, besides reaching a considerable enhancement with respect to a figure of merit introduced in the paper. This indicator is defined by

$$F = 100 \left(1 - \frac{\int_P (p_{ref}(t) - p(t))^2 dt}{\int_P p_{ref}^2(t) dt} \right) \quad (2.19)$$

and is obviously given in percent, where $F = 100\%$ is interpreted as perfect tracking. The integrals are taken over one period P of the reference trajectory.

Employing the reduced observer improved the control quality even further, although the advantage of the reduced observer over the full state observer was way smaller than that of the full observer over the nested controller alone. Recreating the simulation documented in [1] reveals that the peak error using the reduced observer is only eight percent smaller than that of the full state observer.

Note that the goal of this concept is to achieve a behavior of the closed control loop which is approximately linear. Without the use of an appropriate prefilter the controller will only reach stationary accuracy as a response to a step input but will not track any desired reference trajectory perfectly. This has to be kept in mind for the simulations in the following.

The great performance of the controller in friction compensation indicates that the strategy could be applied to other mechanical systems subject to friction effects and that it is not limited to a particular model description. Furthermore, since the construction of the controller only requires a single viscous friction coefficient, there is room for hope that the design procedure does not rely on a specific form of friction model, but can be utilized to deal with other static and dynamic effects encountered in practice.

The focus of this work is on investigating the applicability of the control method described above to an arbitrary mechanical system to achieve friction compensation and to examine whether it is competitive with nonlinear control methods. Of particular interest is the comparison with model-based strategies, to determine whether the forgoing of a completely identified friction model is worthwhile.

In light of the fact that a different dynamic controller may be used later and that the plant models of other physical systems will have state space representation differing from that presented in this section, the design procedure outlined here is held relatively universal. The overall approach to other problems will be identical, while certain intricacies and complications are going to be discussed along the way.

CHAPTER 3

Applying the Controller Design to Other Systems

3.1 Different Classes of Systems under Consideration

As already foreshadowed in Chapter 2, there is justified hope in the potential of the strategy proposed in [1]. It might be suitable for the control of an arbitrary mechanical system in which friction is to be compensated, if that system can be approximated by a linear or linearized state space representation. We are therefore trying to find meaningful examples of system models in the literature, which were successfully used in designing friction compensation controllers or friction force estimates. As mentioned before, the vast majority of approaches uses a model for the description of the relevant friction phenomena. A very general differentiation between the two main types of systems under consideration can be made.

Firstly, there are models which include friction based on a characteristic curve, much like that used for the derivations in Chapter 2. Another example for such a model is given in [2], where the characteristic curve is characterized by

$$F_f = \alpha_1 (\tanh(\beta_1 v) - \tanh(\beta_2 v)) + \alpha_2 \tanh(\beta_3 v) + \alpha_3 v, \quad (3.1)$$

$\alpha_1, \alpha_2, \alpha_3, \beta_1, \beta_2$ and β_3 being positive constants. This yields a characteristic curve that is very similar to that encountered in [1], but that is, moreover, continuously differentiable. As can easily be seen upon examination of the system equations, curve-based models usually describe the dependence of the friction force on the velocity. While the modeled friction force can depend on more than one state at once, obviously complicating the system dynamics considerably, or can be based on other system states altogether, the velocity dependent friction force model is the most common one.

Secondly, and most importantly, there are friction compensation control

schemes based on dynamic friction models. Since its development the LuGre friction model and its alterations have replaced almost all other approaches to friction modeling. Its superiority when it comes to grasping the complicated dynamic effects of friction, combined with the accurate reproduction of static friction effects, make it a highly capable model. There are some issues with the application of the proposed control architecture to systems employing LuGre friction models in their description that shall be discussed in the subsequent sections.

3.2 Conditions for the Applicability

3.2.1 Friction Modeling

The general idea of the cascaded dual loop is neither new nor particularly complex. While theoretically the linear plant model the controller is constructed for is arbitrary, there are some fundamental structural limitations to the applicability of the design procedure to a system.

First and foremost, the notion of compensating friction without using a friction model is slightly misleading. In the following, we will see that the controller design is not possible without any idea of the generation of friction in the respective system at hand and a rough idea about its magnitude. The advantage of the approach over traditional model-based control strategies is much rather that the friction model can be a lot less accurate, without deteriorating the control quality dramatically. Moreover, the design is performed using one of the most common methods in linear control theory, namely pole placement of a linear state space model.

So, as a requirement for the physical system for which the controller is to be designed, we can conclude that the nonlinearity caused by the friction phenomenon has to be reducible to a linear state space representation sufficiently well. We will later see that, particularly when dealing with the dynamic LuGre model, this condition is by no means given, but can be achieved by taking a detour, for example

over model reduction. In general however, and especially for friction models based on characteristic curves, it is fairly simple to find a suitable linear approximation of the curve.

Recapitulating the procedure in Chapter 2, we remember that the characteristic curve in the nonlinear state space model was simplified to a viscous friction coefficient corresponding to the slope of the almost linear section of the curve. In doing so, the identification of the parameters of the nonlinear curve was reduced to the estimation of a single constant value.

3.2.2 Controllability

The second condition a model has to fulfill is controllability. Illustratively, this means that a dynamic system can be transferred to any given state, starting from any given state, by proper choice of the control variable. We will see that, when dealing with the LuGre model, this requirement is critically intertwined with the first one.

It is no surprise that controllability is a prerequisite for the state space representation of a plant. Not only is it necessary mathematically, for pole placement to be possible, but also crucial from a practical standpoint, because the dynamics of the system have to be influenced for the control quality to be satisfactory.

In order for the imperfections of the model and the approximation of nonlinear phenomena by a linear representation to be compensated, the poles of the linearized design model have to be shifted left in the complex plane. It is worth noting here that it is not sufficient for the system to be stabilizable, because the poles of the linearized plant model considered throughout this work are typically too close to the imaginary axis or occur in very weakly damped pairs. Even if the corresponding non-controllable “eigenmovement” of the open loop is stable, it is not fast enough.

This self-evidently means that the linear plant model has to be fully control-

lable. However, just as importantly in our case, it means that the design model used for pole placement with the additional dynamics incorporated in the state space model,

$$\mathbf{A}_d = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ b_a \mathbf{c} & \mathbf{A}_a \end{bmatrix}, \quad \mathbf{b}_d = \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix}, \quad (3.2)$$

has to be fully controllable. According to Kalman's controllability criterion (see [26]), this is equivalent with saying that the controllability matrix

$$\mathbf{M}_c = [\mathbf{b} \quad \mathbf{A}\mathbf{b} \quad \mathbf{A}^2\mathbf{b} \quad \dots \quad \mathbf{A}^{n-1}\mathbf{b}], \quad (3.3)$$

n being the system order, has full rank. For the system discussed in Chapter 2 and for similar approaches to friction modeling, controllability is practically guaranteed. Even if the dynamics of the inner loop should consider the state space representation of e.g. an electric motor, this condition is almost always given, which is readily shown using the aforementioned Kalman criterion.

If we take a look back at system representations including the LuGre friction model, the situation is different. Since it is dynamic and introduces an additional state to the state space system, problems with the controllability arise as follows. Using the nonlinear differential equations of the LuGre model presented and explained in Chapter 1; we can easily derive its formal linearization: the obvious equilibrium of the model is $[v, z] = \mathbf{0}$. Linearizing the model equations yields

$$\begin{aligned} \begin{bmatrix} \Delta \dot{v} \\ \Delta \dot{z} \end{bmatrix} &= \begin{bmatrix} -\frac{\sigma_1 + \sigma_2}{m} & -\frac{\sigma_0}{m} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta v \\ \Delta z \end{bmatrix} + \begin{bmatrix} \frac{1}{m} \\ 0 \end{bmatrix} u \\ \Delta y = \Delta v &= [1 \quad 0] \begin{bmatrix} \Delta v \\ \Delta z \end{bmatrix}, \end{aligned} \quad (3.4)$$

where m is the system's mass. As in the previous case of friction models based on characteristic curves, we can use the Kalman criterion to show that the plant model in and of itself is controllable, since

$$\mathbf{M}_c = [\mathbf{b} \quad \mathbf{A}\mathbf{b}] = \begin{bmatrix} \frac{1}{m} & -\frac{\sigma_1 + \sigma_2}{m^2} \\ 0 & \frac{1}{m} \end{bmatrix} \quad (3.5)$$

obviously has two linearly independent column vectors. On the other hand, considering the additional dynamics occurring in the construction of the design model of the inner loop, controllability is no longer given. We can verify this by constructing the controllability matrix of the open loop design model

$$\begin{bmatrix} \dot{v} \\ \dot{z} \\ \dot{x}_a \end{bmatrix} = \begin{bmatrix} -\frac{\sigma_1+\sigma_2}{m} & -\frac{\sigma_0}{m} & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v \\ z \\ x_a \end{bmatrix} + \begin{bmatrix} \frac{1}{m} \\ 0 \\ 0 \end{bmatrix} u$$

$$y = [1 \quad 0 \quad 0] \mathbf{x}, \quad (3.6)$$

which is given by

$$\mathbf{M}_c = \begin{bmatrix} \frac{1}{m} & -\frac{\sigma_1+\sigma_2}{m^2} & \frac{(\sigma_1+\sigma_2)^2}{m^3} - \frac{\sigma_0}{m^2} \\ 0 & \frac{1}{m} & -\frac{\sigma_1+\sigma_2}{m^2} \\ 0 & \frac{1}{m} & -\frac{\sigma_1+\sigma_2}{m^2} \end{bmatrix}. \quad (3.7)$$

Since the second and third row of this matrix are identical, the controllability matrix evidently has a rank deficiency and the design model is therefore not controllable. This poses a significant obstacle for the design procedure and has to be examined further.

A first step in dealing with this issue is determining how the problematic eigenvalue becomes non-controllable. For this purpose, we have to deal with the invariant zeros of the system. Invariant zeros as defined in [27] are those complex values λ that cause a rank deficiency in the Rosenbrock system matrix given by

$$\mathbf{Z}_e(\lambda) = \begin{bmatrix} \lambda \mathbf{I} - \mathbf{A} & -\mathbf{b} \\ \mathbf{c} & \mathbf{0} \end{bmatrix}. \quad (3.8)$$

In SISO systems of order n this rank deficiency can result from

$$\text{rank} \begin{bmatrix} \lambda \mathbf{I} - \mathbf{A} \\ \mathbf{c} \end{bmatrix} < n, \quad (3.9)$$

if the system is not fully observable. Observability of a system is described analogously to controllability in [26] and means that the state of a system can be reconstructed unambiguously from the knowledge of the input and output signal

in a finite time interval. In accordance to this a rank deficiency in the Rosenbrock matrix can also stem from

$$\text{rank} [\lambda \mathbf{I} - \mathbf{A} \quad -\mathbf{b}] < n, \quad (3.10)$$

if an eigenvalue is not controllable, which is the case in our present situation. Again illustratively speaking, invariant zeros are those complex input frequencies for which a dynamic system blocks transmission, i.e. the output is equal to zero. The term “invariant” refers to the fact that these system zeros can not be influenced by use of linear feedback or state transformations.

Both non-controllable and non-observable eigenvalues result in pole-zero cancellation with an invariant zero and do not show in a frequency domain transfer function due to the fact that their contribution to the system behavior is not influenceable from the input and/or not noticeable at the output. This is particularly dangerous if the affected eigenvalue is unstable because of the internal dynamic effect it entails. Note that an invariant zero can be a pure transmission zero as well, which does not cancel with an eigenvalue of the system.

It is of particular interest to determine where the problem of non-controllability arises for systems of this specific structure, since most single-mass systems have that same exact form. As shown in Appendix A, it is easy to prove that the issue at hand is indeed of structural nature and related to an unwanted pole-zero cancellation. How exactly this comes about, shall be discussed in the following.

Upon inspection of the determinant of the Rosenbrock matrix

$$\det \begin{bmatrix} \lambda \mathbf{I} - \mathbf{A} & -\mathbf{b} \\ \mathbf{c} & \mathbf{0} \end{bmatrix} = \det \begin{bmatrix} \lambda + \frac{\sigma_1 + \sigma_2}{m} & \frac{\sigma_0}{m} & 0 & -\frac{1}{m} \\ -1 & \lambda & 0 & 0 \\ -1 & 0 & \lambda & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} = \frac{\lambda^2}{m} \quad (3.11)$$

we can immediately see that the invariant zeros of the system are at $\lambda_{1,2} = 0$. Since $\lambda = 0$ causes the characteristic polynomial of the Rosenbrock matrix to be

zero, this value is the invariant zero of multiplicity two. This reveals the source of the problem we are facing here: since the linearization of the plant has two invariant zeros in the origin, adding a pole at zero will inevitably cause this pole to be canceled with one of the zeros.

Apart from the proof that systems of the form described above will always exhibit two invariant zeros, which was conducted by performing structural analysis, we still have to determine of what nature the zeros are. Computing the transfer function of the design model in the frequency domain results in

$$G(s) = \mathbf{c} (s\mathbf{I} - \mathbf{A})^{-1} \mathbf{b} = \frac{\frac{1}{m}s^2}{s \left(s^2 + \frac{\sigma_1 + \sigma_2}{m}s + \frac{\sigma_0}{m} \right)}. \quad (3.12)$$

As can be seen right away, one of the zeros remains as a pure transmission zero of the system. It does not cause any further issues. The other invariant zero, more specifically the one resulting from the dynamic modeling of friction, however, cancels with the pole of the system in the origin. This is the non-controllable eigenvalue traced by the examination of the controllability matrix of the design model and it is particularly problematic, because the canceled eigenvalue is not stable.

The non-controllability of the eigenvalue in zero is caused by the introduction of the additional state z in the LuGre friction model, which describes the surface deformation. As long as the linearization of the LuGre model contains this second state next to the velocity, the resulting design model will never be fully controllable. After discovering the root of the problem, we can try to systematically come up with a solution for it.

An attempt to modify the design procedure or the additional dynamics of the controller fails. As the only option to deal with the issues of the dynamic friction model, a reconstruction of the linearized state space representation remains. Luckily, such a modification of the system is possible by performing model reduction.

This approach shall be discussed in Section 3.3.

As a final remark in this part of the introduction, the following has to be stated clearly. Both the LuGre friction model and the aforementioned systems incorporating characteristic curve-based friction descriptions are nonlinear. Terms used in the statements and derivations of this chapter are almost entirely limited to the description of linear systems. The notion of an “eigenvalue”, for example, strictly speaking, does not make any sense in the context of a nonlinear system.

However, when dealing with the linearization of nonlinear differential equations and when examining small movements of the states around the equilibrium the linearization is performed around, the language of linear control theory is adequate. Not only does it give us a very good and visual idea of the system behavior but also approximates it decently well quantitatively.

3.3 Reduction of the LuGre Model

We discussed in Section 3.2.2 that the LuGre friction modeling philosophy requires further treatment for our control approach to even be applicable in its presence and the design procedure to be feasible. We also discovered that including the additional state of this dynamic friction model in a conventional linearization will not result in a controllable design model, rendering this intuitive approach insufficient.

Another issue that has to be mentioned here is that our ultimate goal is to come up with a control method that foregoes an accurate modeling of the friction effect to be mitigated. The identification of a relatively complicated and detailed dynamic friction model just to then reduce it to its linearization would contradict our initial line of approach and essentially ridicule every effort that is made in finding a simple control algorithm.

However, the inspection of the LuGre model gives us important insights about

the dynamic behavior of friction effects and the nonlinear model will later be an important building block and guide for every simulation that is performed to verify the performance of our controller. For that reason, we will dive deeper into the conversion of the LuGre model into a linear representation, qualified for the controller design. To that end, we are, as mentioned in the preceding section, going to employ model reduction techniques.

Before introducing the successful idea of using model reduction to obtain not only a controllable design model but also satisfactory control quality thereupon, some remarks about previous trials shall be made. As a very first and naive approach, the equilibrium around which the linearization was performed was slightly shifted from the origin of the state space. In a similar fashion, the zero entries in the state space matrices of the model were tweaked, to artificially produce a controllable system. The consequence in both cases, however, was that the pole placement algorithm used in the design procedure excessively exploited the falsely nonzero elements in the matrices, to achieve the desired pole location and the subsequent simulation showed unstable behavior.

Model reduction was already used very early on, the idea being that there has to be some way of depicting the friction effect as a single viscous friction parameter, since it was already possible for models based on characteristic curves. Even though the approximation by a single constant might be significantly worse in the presence of a dynamic model, it should still theoretically be feasible. The resulting first order model has to give us a controllable design model, because a model of order one with nonzero input matrix can not be non-controllable.

Upon inspection of the Hankel singular values, which can be interpreted as a measure for the influence of the single eigenvalues of a system, it became apparent right away that such a reduction was not justified with respect to properly

representing the system dynamics. However, using the `balred` function for model reduction in MATLAB yielded a design model with which a stable and satisfactory controller could be constructed.

On the other hand, using this built-in function on a LuGre model with different parametrization did not even result in a stable control system. This difference showed to be explicable by the very different dynamic behaviors of different LuGre models, depending on their respective parameter values. We therefore want to determine the origin of this discrepancy and how the reduction of the linearization can be performed in order to achieve desirable control quality for all dynamic cases.

A first step in examining the linearization of the LuGre model is to calculate its eigenvalues. Looking back at the system matrix, we can determine the characteristic polynomial

$$|\lambda \mathbf{I} - \mathbf{A}| = \lambda^2 + \frac{\sigma_1 + \sigma_2}{m} \lambda + \frac{\sigma_0}{m} \quad (3.13)$$

and its roots, which are given by

$$\lambda_{1,2} = -\frac{\sigma_1 + \sigma_2}{2m} \pm \sqrt{\left(\frac{\sigma_1 + \sigma_2}{2m}\right)^2 - \frac{\sigma_0}{m}}. \quad (3.14)$$

Without further knowledge of the parameter values, no quantitative statement about the pole locations is possible. As a condition for the existence of a pole pair with an imaginary part, we quickly want to establish a relation between the parameters. For the roots of the characteristic polynomial to not be real its discriminant has to be negative:

$$\left(\frac{\sigma_1 + \sigma_2}{2m}\right)^2 - \frac{\sigma_0}{m} < 0 \Leftrightarrow \frac{(\sigma_1 + \sigma_2)^2}{4\sigma_0} < m. \quad (3.15)$$

This gives us a rough idea of how large the mass of the system has to be relative to the parameters of the friction model in order for the linearization to have a complex pole pair capable of causing vibration.

The next step is to visualize the eigenvalue locations of the system. Since the structure of the linearized LuGre model is always the same, no matter what parameter values are chosen, we can display the pole locations dependent on the system's mass (or the inertia of a rotatory system) using the root locus. This gives us a qualitative understanding of what pole constellations are possible. For the construction of the root locus we reformulate the characteristic polynomial in the following way:

$$\lambda^2 + \frac{\sigma_1 + \sigma_2}{m}\lambda + \frac{\sigma_0}{m} = \left(\frac{\sigma_1 + \sigma_2}{m}\lambda + \frac{\sigma_0}{m} \right) + \lambda^2. \quad (3.16)$$

The right hand side of this equation can be interpreted as the characteristic polynomial of a fictitious system with the open loop transfer function

$$G_{fict}(s) = \frac{1}{m} \frac{(\sigma_1 + \sigma_2)\lambda + \sigma_0}{\lambda^2}. \quad (3.17)$$

Consequently, the root locus of this fictitious system can be plotted, where the reciprocal of the mass is treated as the variable gain of a P-controller. Obviously, the system has a zero at $-\frac{\sigma_0}{\sigma_1 + \sigma_2}$ and a pole of multiplicity two in the origin. The result is depicted in Figure 3.1.

For $1/m \rightarrow 0$ or $m \rightarrow \infty$ both poles lie in the origin and, analogously, for $m \rightarrow 0$ the linearization has two real eigenvalues which strive towards $-\frac{\sigma_0}{\sigma_1 + \sigma_2}$ and $-\infty$. We can immediately see that the eigenvalues of the systems are always stable, excluding the extreme case of infinite mass. A second insight is that, for a larger mass, the linearized system obtains an oscillatory eigenvalue pair, which can also be weakly dampened.

This elaborate procedure of examining possible eigenvalue locations serves the purpose of introducing an important differentiation between cases of LuGre model parameterizations, which require different treatment to ensure the best possible control quality. As already stated above, the built-in model reduction algorithm

Figure 3.1: Pole Positions of the Linearized LuGre Model Depending on the System Mass

of MATLAB lends itself to the modification of some LuGre models. The inspection of a variety of different parameter sets from the literature shows that this approach seems to work really well if the linearization has eigenvalues close to the intersection point of the root locus branches. This can be the case for a strongly dampened complex pole pair, encountered, for example, in the linearization using the parameter set from [7]. The parameters could also yield eigenvalues that lie relatively close together on the real axis for this function to be applicable.

Unfortunately, this method only covers a very small portion of the root locus curve. We want to come up with an idea that includes the entirety of possible pole locations, to develop a closed theory. An option for real eigenvalues is the application of a formal model reduction algorithm. Such eigenvalue constellations are, for example, reached for the parameterizations in [28], [29] or [30]. For the reduction the modal technique proposed by Litz is used, which is described in [31]. It essentially splits the system dynamics up in a dynamically dominant part, which

is typically slower, and a faster part, that is to be neglected.

Starting from an arbitrary linear state space system

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{b}u \\ y &= \mathbf{c}\mathbf{x},\end{aligned}\tag{3.18}$$

the procedure works as follows. The first step is the transformation of the state space model to its modal form

$$\begin{aligned}\dot{\mathbf{z}} &= \mathbf{\Lambda}\mathbf{z} + \mathbf{b}^*u \\ y &= \mathbf{c}^*\mathbf{z}\end{aligned}\tag{3.19}$$

using the linear transformation $\mathbf{x} = \mathbf{V}\mathbf{z}$. In the following, we determine the set of dominant eigenvalues by use of certain measures for the influence of the single modes. Since in our case we just want to take the eigenvalue closer to the imaginary axis along, this step is very straight-forward. After breaking the modal coordinates $\mathbf{z}$ up into the states to be maintained for the reduced model, $\mathbf{z}_1$, and states $\mathbf{z}_2$ to be dropped, we can formulate the state equations for the relevant part as follows:

$$\dot{\mathbf{z}}_1 = \mathbf{\Lambda}_1\mathbf{z}_1 + \mathbf{b}_1^*u\tag{3.20}$$

and the equations for the omitted states analogously:

$$\dot{\mathbf{z}}_2 = \mathbf{\Lambda}_2\mathbf{z}_2 + \mathbf{b}_2^*u.\tag{3.21}$$

The most involved part of the reduction algorithm is the recreation of the influence of the neglected states. It is achieved by introducing the reconstruction matrix

$$\mathbf{E} = \mathbf{\Lambda}_2^{-1} \left(\mathbf{b}_{21} + (\mathbf{b}_2^* - \mathbf{b}_{21}\mathbf{b}_{11}^{-1}\mathbf{b}_1^*) \left(\bar{\mathbf{b}}_1^{*\text{T}} \mathbf{b}_{11}^{-1} \mathbf{b}_1^* \right)^{-1} \bar{\mathbf{b}}_1^{*\text{T}} \right) \mathbf{b}_{11}^{-1} \mathbf{\Lambda}_1,\tag{3.22}$$

where the bar indicates the complex conjugate of the respective matrix and the matrices $\mathbf{b}_{11}$ and $\mathbf{b}_{21}$ are defined element-wise by

$$\begin{aligned} [\mathbf{b}_{11}]_{i,j} &= -\frac{[\mathbf{b}_1^* \mathbf{Q}_u \bar{\mathbf{b}}_1^{*T}]_{i,j}}{\lambda_i + \bar{\lambda}_j}, \quad i = 1, \dots, m; \quad j = 1, \dots, m, \quad \text{and} \\ [\mathbf{b}_{21}]_{i,j} &= -\frac{[\mathbf{b}_2^* \mathbf{Q}_u \bar{\mathbf{b}}_1^{*T}]_{i,j}}{\lambda_{m+1} + \bar{\lambda}_j}, \quad i = 1, \dots, n-m; \quad j = 1, \dots, m. \end{aligned} \quad (3.23)$$

In these expressions, m is the number of eigenvalues taken into account in the reduced system and λ_i indicates the single eigenvalues. The matrix $\mathbf{Q}_u$ is the diagonal matrix of squared control variable amplitudes, which are chosen as typical values of the control signal. By partitioning the Transformation matrix as

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_{11} & \mathbf{V}_{12} \\ \mathbf{V}_{21} & \mathbf{V}_{22} \end{bmatrix}, \quad (3.24)$$

wherein the top block of columns is associated with the conserved modal states $\mathbf{z}_1$, we can reconstruct the reduced model in the original coordinates:

$$\tilde{\mathbf{x}}_1 = \mathbf{V}_{11}\mathbf{z}_1 + \mathbf{V}_{12}\mathbf{E}\mathbf{z}_1 \triangleq \mathbf{F}\mathbf{z}_1. \quad (3.25)$$

The fully reduced and reconstructed state space system is then given by the differential equations

$$\dot{\tilde{\mathbf{x}}}_1 = \underbrace{\mathbf{F}\mathbf{\Lambda}\mathbf{F}^{-1}}_{\tilde{\mathbf{A}}} \tilde{\mathbf{x}}_1 + \underbrace{\mathbf{F}\mathbf{b}_1^*}_{\tilde{\mathbf{b}}} u \quad (3.26)$$

and to obtain the approximation of the controlled variable the part of the output matrix associated with the preserved states, $\mathbf{c}_1$, is used:

$$\tilde{y} = \mathbf{c}_1 \tilde{\mathbf{x}}. \quad (3.27)$$

By using the procedure described above, we again obtain a first-order system which approximates the behavior of a linearized LuGre model and which yields an intrinsically controllable design model. As a final remark, it has to be said that the

quality of the model reduction is of course better if the eigenvalues are far apart on the real axis. However, it has shown in simulation that even for pole positions relatively close to each other, the subsequent control quality was satisfactory.

The last case we have to consider is that of a weakly dampened eigenvalue pair in the linearization of the LuGre model, as it can be found in [6], [19] or [32]. In Figure 3.1 these locations lie outside the dashed lines in the complex plane. We refer to poles above or below the enclosed area as weakly dampened. It is intuitive and by no means surprising that the oscillatory behavior caused by a complex pole pair can not be emulated by a system of order one. So far neither MATLAB's **balred**-function nor the modal method of Litz reduce the linearized model well enough to result in a closed loop control of sufficient quality.

Thankfully, there is a very simple remedy to this that is also easy to implement. If we recall that we have already successfully approximated the friction effect of models based on characteristic curves by its behavior in the range of viscous friction, it seems natural to chose a similar approach to the LuGre model. To that end, we just drop the nonlinear dynamics of the model completely, resulting in the simple first order approximation

$$\dot{\tilde{v}} = -\frac{\sigma_2}{m}\tilde{v} + \frac{1}{m}u. \quad (3.28)$$

The remaining factor σ_2 is literally the viscous friction parameter of the model. With this approximation in the design procedure, satisfactory tracking control in simulation is achieved. Checking back on the linearized models that were already successfully reduced, reveals that this idea seems to be able to replace the **balred** approach in that it results in an even better control behavior.

However, if the poles of the system are real and relatively far apart, formal model reduction is inevitable. In these cases using the modal procedure of Litz is preferable, even though considering only viscous friction does yield a stable control

circuit. We can therewith cover all possible pole constellation of the linearized LuGre model and receive a reduced model, adequate for the design procedure.

It has to be said that we obviously do not have a friction model at hand before beginning the controller design and that we, frankly, do not want to use one in the strict sense. But the inspection of the different relevant cases for the dynamics of the LuGre model and the successive experiments based on simulations reveal two important facts.

Firstly, the approximation of a relevant friction model from literature, whether it is dynamic or not, by a first order model is absolutely sufficient for the design of an effective controller following our scheme. Since the LuGre model is known to describe friction effects very well, a simulatory control of this dynamic model with our linear controller is expected to yield meaningful results.

Secondly, and more importantly, it will become apparent quickly that the accuracy of the parameter values used in the linearization and reduction does not have to be high. It is therefore possible to compare both approaches in a physical system to be controlled based on a fairly rough identification of the three friction parameters σ_0 , σ_1 and σ_2 . After estimating these parameters, the performance of a controller based on a thoroughly reduced model and that of a controller based only on the viscous friction coefficient σ_2 can easily be compared.

This allows us to think of the parameter estimation not as the identification of a friction model, but as the identification of a simple linear state space representation of the controlled system, which is one of the main objectives of the idea.

CHAPTER 4

Application to Specific System Models

Throughout the research leading up to this thesis, a number of different models from recent literature were researched and the proposed system models underwent treatment with our control approach. All considered models were developed within the last 20 years, with a considerable portion being less than ten years old. While a number of the examples contain a single-mass system like the one we have already encountered in Chapter 2, there are some that deal with more complicated dynamic models. Some of the models explicitly take motor dynamics into account, yielding an inner loop model of higher order.

There are also mechanical system models subject to friction in which the position signal is fed back to the calculation of the acceleration. This system structure renders the proposed nested control with an inner velocity loop that is designed separately, impossible to apply. It is shown, however, that in a similar fashion, we can design a dual position loop for these systems, which yields satisfactory control quality.

Finally, there are systems where the modeling of nonlinearities is not limited to friction effects. We will see that our control strategy lends itself to the mitigation of backlash in one particular case, which in turn sparks hope that the idea of a nested control architecture might not be limited to friction compensation.

4.1 Simple Single-Mass System

The most elementary system that could be thought of, is an either translatory or rotatory single-mass system with mass m or inertia J . Its general structure is given by the block diagram in Figure 4.1. The nonlinear friction block can either contain a characteristic curve-based nonlinearity or a dynamic friction model like

Figure 4.1: Single-Mass System with Nonlinear Friction

the LuGre approach, which introduces an additional dynamic state.

4.1.1 Characteristic Curve-Based Model

For the model description in [2], which already functioned as an example for friction modeling using a characteristic curve, the complete nonlinear dynamics are given by

$$\begin{aligned}
 \dot{x}_1 &= x_2 \\
 \dot{x}_2 &= \frac{1}{J} (K_1 u - K_2 x_2 - T_f) \\
 T_f &= \alpha_1 (\tanh(\beta_1 x_2) - \tanh(\beta_2 x_2)) + \alpha_2 \tanh(\beta_3 x_2) + \alpha_3 x_2.
 \end{aligned} \tag{4.1}$$

Therein, the state x_1 is the angle and x_2 denotes the angular velocity. In contrast to the considerations in the paper, disturbances and uncertainties are neglected for the time being, since the purpose of this chapter is to present an overview over the design procedure rather than a thorough assessment of the control performance.

To illustrate the application of the nested structure to an arbitrary system, we want to perform the design procedure step by step. Consequently, we first reduce the differential equation to its linear part, which contains only a viscous friction coefficient. We directly obtain the linear state space model

$$\begin{aligned}
 \dot{\mathbf{x}} &= \begin{bmatrix} 0 & 1 \\ 0 & \frac{-K_2 - \alpha_3}{J} \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ \frac{K_1}{J} \end{bmatrix} u \\
 y &= \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x},
 \end{aligned} \tag{4.2}$$

wherein the equation for the angular velocity is given by

$$\dot{x}_2 = \underbrace{\frac{-K_2 - \alpha_3}{J}}_A x_2 + \underbrace{\frac{K_1}{J}}_b u, \quad (4.3)$$

ready for the construction of the inner loop. Plugging these values into our design model for the inner loop and using a simple integrator as the additional dynamics, yields essentially the same matrices as for the model treated in [1]:

$$\mathbf{A}_{di} = \begin{bmatrix} \frac{-K_2 - \alpha_3}{J} & 0 \\ 1 & 0 \end{bmatrix}, \mathbf{b}_{di} = \begin{bmatrix} \frac{K_1}{J} \\ 0 \end{bmatrix}. \quad (4.4)$$

As stated before, pole placement of the design model is readily carried out by modifying the poles of the matrix $\mathbf{A}_{di} - \mathbf{b}_{ki}\mathbf{K}_v$. The index v of the feedback vector indicates that we are placing the poles of the inner loop, which is meant to control the velocity or, in this case, the angular velocity. Rather than calculating the control vector on paper, we employ the built-in MATLAB function `place(A,b,poles)`, which calculates the gain vector $\mathbf{K}_v$ based on the desired pole location specified in `poles`. As the poles of the linear closed loop model, we choose zeros of the normalized second-order normalized Bessel polynomial as proposed in the original paper (see [1]), namely $\hat{s}_{1,2} = -4.053 \pm 2.34j$. They are scaled by the reciprocal of the required settling time for the velocity loop, $T_{sv} = 0.125$ s, to obtain the inner loop poles

$$\mathcal{S}_i = \{-32.424 \pm 18.72j\}. \quad (4.5)$$

Partitioning the resulting feedback vector into its two components gives us the scalar $k_{v,1} = [\mathbf{K}_v]_1$, used in state feedback of the velocity in the inner loop, and $k_{v,2} = [\mathbf{K}_v]_2$, the output matrix of the additional dynamic controller. The closed inner loop is then given by the state space model

$$\begin{bmatrix} \dot{x}_2 \\ \dot{x}_{ai} \end{bmatrix} = \begin{bmatrix} \frac{-K_2 - \alpha_3}{J} - \frac{K_1}{J}k_{v,1} & \frac{K_1}{J}k_{v,2} \\ -1 & 0 \end{bmatrix} \begin{bmatrix} x_2 \\ x_{ai} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_v$$

$$x_2 = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_2 \\ x_{ai} \end{bmatrix}, \quad (4.6)$$

where x_{ai} is the additional dynamic state of the controller. u_v indicates the virtual control variable of the inner loop. We can therewith construct the model of the outer loop, comprised of the closed inner loop and an integrator to calculate the angle. This results in the model

$$\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_{ai} \end{bmatrix} &= \underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 0 & \frac{-K_2-\alpha_3}{J} - \frac{K_1}{J}k_{v,1} & \frac{K_1}{J}k_{v,2} \\ 0 & -1 & 0 \end{bmatrix}}_{\mathbf{A}_1} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_{ai} \end{bmatrix}}_{\mathbf{x}_1} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}_{\mathbf{b}_1} u_v \\ \theta = x_1 &= \underbrace{\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}}_{\mathbf{c}_1} \mathbf{x}_1 \end{aligned} \quad (4.7)$$

for the augmented inner loop. A design model of the outer loop is constructed analogously to that of the inner loop:

$$\mathbf{A}_{do} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{-K_2-\alpha_3}{J} - \frac{K_1}{J}k_{v,1} & \frac{K_1}{J}k_{v,2} & 0 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{b}_{do} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}. \quad (4.8)$$

The eigenvalues of this matrix are now shifted via pole placement in the same fashion as for the inner loop, where the desired eigenvalue locations of the inner loop stay unaffected and the remaining two poles are chosen as the zeros of the normalized Bessel polynomial scaled by the reciprocal of the prescribed settling time $T_s = 0.5$ s, therefore

$$\mathcal{S}_o = \{-8.106 \pm 4.68 j, -32.424 \pm 18.72 j\}. \quad (4.9)$$

This yields the outer loop feedback vector $\mathbf{K}_p$, where the index p again indicates that we are dealing with the position loop, in this case the angular position or just the angle. It only has to be broken up into the state feedback vector of the outer loop, $\mathbf{k}_{p,1}$, consisting of the first three entries in $\mathbf{K}_p$ and the scalar output gain of the additional integrator of the outer loop, $k_{p,2}$, which is just the last

entry. The controller design is therewith completed and we can formulate a linear reference model for the entire closed control loop:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_{ai} \\ \dot{x}_{ao} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{A}_1 - \mathbf{b}_1 \mathbf{k}_{p,1} & \mathbf{b}_1 k_{p,2} \\ -b_a \mathbf{c}_1 & A_a \end{bmatrix}}_{\mathbf{A}_r} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_{ai} \\ x_{ao} \end{bmatrix}}_{\mathbf{x}_r} + \underbrace{\begin{bmatrix} \mathbf{0} \\ b_a \end{bmatrix}}_{\mathbf{b}_r} r$$

$$\theta = x_1 = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{c}_r} \mathbf{x}_r. \quad (4.10)$$

Herein, the reference input is symbolized by r and the controlled variable x_1 is renamed θ , the angle of the rotatory system. The state space representation of the additional dynamics are given by $A_a = 0$ and $b_a = 1$ for the outer and inner loop alike.

This reference model derived above is used in the subsequent simulation to assess the ability of the linear controller to mitigate friction in the system to an extent where the closed loop can be treated like a linear system as a good approximation. Because the matching is obviously not going to be perfect due to the nonlinearity that has been neglected throughout the design process, we want to examine how good the tracking quality can be.

The last step in the design process is the construction of the linear observer, which estimates the states of the inner loop, the angular velocity and the additional integrator state, based on the output of the inner loop and its input, the virtual control variable u_c . As we remember, the observer is designed according to the linear model of the inner loop, i.e. the nonlinear friction effect is neglected.

The idea of an observer in the conventional sense is to estimate the state of a dynamic system by modeling the system dynamics in an online simulation. Since the state variables are often difficult and expensive to measure or not measurable at all, this procedure can provide a reliable approximation of said variable values and therewith facilitate state feedback control. For our purpose, however, the

observer is added to the control loop for the reason described in Chapter 2. It is designed by use of pole placement as well, modifying the eigenvalue locations of the matrix $(\mathbf{A}_o - \mathbf{L}\mathbf{c}_o)$, and has the structure of a Luenberger observer. As the system matrices $\mathbf{A}_o$, $\mathbf{b}_o$ and $\mathbf{c}_o$ the matrices of the linear closed inner loop are chosen, namely

$$\mathbf{A}_o = \begin{bmatrix} \frac{-K_2 - \alpha_3}{J} - \frac{K_1}{J}k_{v,1} & \frac{K_1}{J}k_{v,2} \\ -1 & 0 \end{bmatrix}, \mathbf{b}_o = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \text{ and } \mathbf{c}_o = [1 \ 0]. \quad (4.11)$$

The observer vector $\mathbf{L}$ is readily obtained using pole placement. As the observer poles, we choose

$$\mathcal{S}_{obs} = \{-97.272 \pm 56.16 j\}, \quad (4.12)$$

the scaled Bessel poles that are shifted in the left complex half plane by a factor of three with respect to the poles of the inner loop. This is done so the observer error decreases much faster than control actions due to reference variable changes or disturbances are executed.

Instead of using the complete state of the inner loop for control, only the estimate of the additional integrator state $\hat{x}_{ai}$ is fed back into the loop. With the controller and observer complete, we can turn towards the simulation, which is set up as follows.

In analogy with the paper which the friction and system model stem from, the reference signal is a sine wave of low amplitude. The rationale behind this is that we want to challenge the control system with a reference trajectory which causes the velocity to be near zero, where the influence of the nonlinear friction effect is most prominent, for a significant portion of the simulation time. Unfortunately, the parameter values used in the paper were documented insufficiently, so that we have to partially rely on sensible numbers taken from similar application examples found in literature. Despite drastic variation of these parameters the controller still yielded satisfactory results.

For the parameterization of the friction curve the values

$$\alpha_1 = 0.25 \frac{\text{kg m}^3}{\text{s}^2}, \alpha_2 = 0.5 \frac{\text{kg m}^3}{\text{s}^2}, \alpha_3 = 0.01 \frac{\text{kg m}^3}{\text{s}} \quad (4.13)$$

as well as

$$\beta_1 = 100 \text{ s}, \beta_2 = 1 \text{ s and } \beta_3 = 100 \text{ s} \quad (4.14)$$

are adopted from the paper. The gain parameters $K_1 = \frac{K_t}{R_a}$ and $K_2 = \frac{K_t K_e}{R_a}$ require an educated guess about the magnitude of the individual parameters used to approximate the behavior of the electric motor in the system and are therefore chosen as

$$K_t = 0.8 \frac{\text{kg m}^3}{\text{s}^2 \text{ A}}, K_e = 4 \text{ and } R_a = 0.9 \Omega. \quad (4.15)$$

Furthermore, we assume that the system has unit moment of inertia, therefore $J = 1 \text{ kg m}^2$.

The experiments are performed in SIMULINK using the ode45 solver with automatic step size selection, where the block diagrams from Chapter 2 were directly transferred to the simulation environment. Because we are lacking a physical system, the control performance is studied based on a plant model which incorporates the respective nonlinear friction model from the literature source the parameters were gathered from. This is also the approach of other upcoming simulations.

With a sine wave of amplitude 0.01 rad at a frequency of 1Hz a first test is to be made as to how well the control circuit performs in tracking the reference signal. To that end, the observer is omitted for the time being and the additional dynamic state is used for state feedback in the simulation. That yields the output value and reference model output progression displayed in the plot in Figure 4.2.

We can directly see that the tracking performance is fairly good. Even though a thorough comparison with the control quality presented in the paper is inadequate due to the missing parameter values, we can conclude that, at first glance,

Figure 4.2: Output Progression (orange) and Reference Output (blue) for the Model in [2]

the tracking accuracy seems at least competitive and the required energy in form of control variable consumption is reasonable. This becomes particularly apparent upon inspection of the tracking error in Figure 4.3.

When speaking of the tracking error in this and the following simulations, the deviation of the output of the control system with the nonlinear plant from the output of the linear reference model is meant. This comparison is more meaningful than contrasting the controlled variable to the system input because there is, naturally, a delay between the input and output and the controller still lacks a prefilter, which could ensure the correct amplitude for an arbitrary frequency. Bear in mind that we are using the term “reference” for both the desired trajectory the system is supposed to track and the benchmark system the closed control loop is compared to.

The error reaches its peak right after the zero crossing of the velocity and exhibits a maximum of eight percent of the reference signal amplitude. In general though and especially given the fact that we are requiring extremely small movements from the system, the tracking quality is good. This particularly shows when

Figure 4.3: Tracking Error without Observer (Model in [2])

examining the figure of merit in Eq. 2.19, which yields a value of $F = 99.3\%$. As a next step, we want to examine the influence of the linear observer on the performance. In the following simulation, the observer is used to estimate the additional dynamic state of the inner loop integrator x_{ai} , which yields the tracking error in Figure 4.4.

Right off the bat, it can be seen that by use of the estimated integrator state during operation, the peak error is reduced to less than five percent of the input amplitude. Moreover, the figure of merit value is now up to $F = 99.8\%$. These results prove what we have initially assumed: the control concept is not only applicable to other systems from literature, but also yields promising tracking quality in simulation.

Two final remarks have to be made. Firstly, the impact the observer has on the control performance depends critically on the system description and friction model at hand. We will see later that, in particular when dynamic friction models are studied, the effect of the observer is typically much more substantial and can indeed reduce the peak error by a factor of two to three. Secondly, tuning the

Figure 4.4: Tracking Error Using the Observer (Model in [2])

required settling time in the design procedure can further improve the tracking quality at the expense of larger control energy. This will also be addressed later, since we will see that the correct choice of settling time can be paramount for satisfying results.

4.1.2 LuGre Friction Model

The logical next step in our investigation is answering the question whether the control strategy is suitable to replace model-based controller designs based on dynamic friction models. In recent publications on friction compensation, the predominant choice is the LuGre model, hence why it gets so much attention in this work. As we can easily see from Section 3.3, the design procedure for LuGre-based approaches is more laborious than the ones discussed so far. However, we will find out quickly that for the most part the construction of the controller is executed identically.

We want to use the nonlinear system model described in [3] and design a linear controller in the same way as in Section 4.1.1. Therefore, in a first step, we need to linearize the friction model itself and determine what model reduction approach

will result in the most ideal control quality. Linearizing the LuGre model results in the state equations

$$\begin{aligned} \begin{bmatrix} \Delta \dot{v} \\ \Delta \dot{z} \end{bmatrix} &= \begin{bmatrix} -\frac{\sigma_1 + \sigma_2}{m} & -\frac{\sigma_0}{m} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta v \\ \Delta z \end{bmatrix} + \begin{bmatrix} \frac{1}{m} \\ 0 \end{bmatrix} u = \begin{bmatrix} -211.95 & -15291 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta v \\ \Delta z \end{bmatrix} + \begin{bmatrix} 0.3058 \\ 0 \end{bmatrix} u \\ \Delta y &= \Delta v = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta v \\ \Delta z \end{bmatrix}, \end{aligned} \quad (4.16)$$

where the model parameters are $m = 3.27 \text{ kg}$, $\sigma_0 = 50,000 \frac{\text{N}}{\text{m}}$, $\sigma_1 = 600 \frac{\text{kg}}{\text{m}}$ and $\sigma_2 = 93.09 \frac{\text{kg}}{\text{m}}$. To determine whether a traditional model reduction approach is preferable or considering only the viscous friction parameter is sufficient, we calculate the eigenvalues which turn out to be

$$\lambda_{1,2} = -105.98 \pm 63.71 j. \quad (4.17)$$

Therefore, we go down the path of simply dropping the nonlinear dynamics and the additional state z from the linearization to receive our controllable design model.

This design model is constructed in analogy with the procedure described in Section 4.1.1. With our model description

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{1}{m} (-F_f + K_a K_t u) \\ F_f &= \sigma_0 z + \sigma_1 \dot{z} + \sigma_2 x_2 \\ \dot{z} &= x_2 - \sigma_0 \frac{|x_2|}{g(x_2)} z, \end{aligned} \quad (4.18)$$

the standard LuGre model description in the context of a single-mass system and the nonlinear function $g(x_2)$ described in Chapter 1, we can perform the controller design. Again, the state variables x_1 and x_2 signify the position and velocity, respectively. To receive our design model, the nonlinear system dynamics are reduced to the linear model

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 \\ 0 & -\frac{\sigma_2}{m} \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ \frac{K_a K_t}{m} \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}, \quad (4.19)$$

which then, combined with an integrator as the additional dynamics, yields the inner loop design model

$$\mathbf{A}_{di} = \begin{bmatrix} -\frac{\sigma_2}{m} & 0 \\ 1 & 0 \end{bmatrix}, \mathbf{b}_{di} = \begin{bmatrix} \frac{K_a K_t}{m} \\ 0 \end{bmatrix}. \quad (4.20)$$

The rest of the design process is completely identical to the one described in detail in Section 4.1.1 above. We first place the eigenvalues of the inner loop design model in the locations specified by the zeros of the second order Bessel polynomial, scaled by the requested settling time of the inner loop, $T_{sv} = 0.075$ s:

$$\mathcal{S}_i = \{-54.04 \pm 31.2 j\}. \quad (4.21)$$

We therefore arrive at a closed inner loop with the state space representation

$$\begin{aligned} \begin{bmatrix} \dot{x}_2 \\ \dot{x}_{ai} \end{bmatrix} &= \begin{bmatrix} -\frac{\sigma_2}{m} - \frac{K_a K_t}{m} k_{v,1} & -\frac{K_a K_t}{m} k_{v,2} \\ -1 & 0 \end{bmatrix} \begin{bmatrix} x_2 \\ x_{ai} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_v \\ x_2 &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_2 \\ x_{ai} \end{bmatrix}, \end{aligned} \quad (4.22)$$

closely resembling the structure of the inner loop of the previous model. The inner loop is subsequently augmented by an integrator to receive the position:

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\frac{\sigma_2}{m} - \frac{K_a K_t}{m} k_{v,1} & -\frac{K_a K_t}{m} k_{v,2} \\ 0 & -1 & 0 \end{bmatrix} \mathbf{x}_1 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u_v \\ x_1 &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \mathbf{x}_1, \end{aligned} \quad (4.23)$$

which again gives us the design model for the outer loop,

$$\mathbf{A}_{do} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{\sigma_2}{m} - \frac{K_a K_t}{m} k_{v,1} & -\frac{K_a K_t}{m} k_{v,2} & 0 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \mathbf{b}_{do} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}. \quad (4.24)$$

Based on this model, we place the eigenvalues in the locations given by the scaled Bessel poles

$$\mathcal{S}_o = \{-54.04 \pm 31.2 j, -13.51 \pm 7.8 j\}, \quad (4.25)$$

resulting in the reference model of identical form as the one derived in Section 4.1.1. Lastly, we construct an observer for the inner loop. With its poles specified as

$$\mathcal{S}_{obs} = \{-162.12 \pm 93.6 j\}, \quad (4.26)$$

the matrices of the observer subsystem again take the same exact shape as the ones encountered before:

$$\mathbf{A}_o = \begin{bmatrix} -\frac{\sigma_2}{m} - \frac{K_a K_t}{m} k_{v,1} & -\frac{K_a K_t}{m} k_{v,2} \\ -1 & 0 \end{bmatrix}, \mathbf{b}_o = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \text{ and } \mathbf{c}_o = \begin{bmatrix} 1 & 0 \end{bmatrix}. \quad (4.27)$$

How the design procedure is carried out in a programming environment to calculate the controller and observer gains for operation, is shown in Appendix B. Note that the code also covers the case where modal model reduction has to be carried out.

The simulations performed in the following illustrate the first advancement over the initial inspiration in [1]. Not only do they show that the presented control concept is applicable beyond the exemplary system model from the paper, but they also prove that it is successful in compensating friction effects when a dynamic friction model is underlying the simulation. In contrast to the paper studied in Section 4.1.1, the model description and experimental results in the one at hand are really well documented. This spares us the guesswork about parameter values, which are given as

$$F_c = 3.5 \text{ N}, F_s = 9 \text{ N and } v_s = 0.075 \frac{\text{m}}{\text{s}}, \quad (4.28)$$

in addition to the aforementioned friction model parameters. The constant factors in the state space model are given by

$$K_a = 5.3 \times 10^{-2} \frac{\text{Nm}}{\text{V}} \text{ and } K_t = 8.545 \times 10^2 \frac{1}{\text{m}}. \quad (4.29)$$

Figure 4.5: Output Progression (orange) and Reference Output (blue) for the System with LuGre Model in [3]

We once again gear the simulation towards the experimental setup presented in the paper by choosing a rectangular wave of amplitude 0.01 m as the reference signal.

Similarly to the previous simulation, we first neglect the observer and examine the tracking quality of the controller in and of itself. However, instead of plugging only the nonlinearity given by the characteristic curve into the model we now include the LuGre friction model with its additional state z , while the linear controller is based on the reduced linear design model. The simulation results are plotted in Figure 4.5.

It is obvious that the tracking quality in comparison with a linear reference model is, again, very good. This becomes even clearer when the deviation of the actual output variable from the reference progression is viewed isolatedly, as in the plot of Figure 4.6.

Apart from the peaks, which occur when the system approaches zero velocity, the controlled system matches the behavior of the linear reference almost perfectly. The maximum error remains smaller than ten percent of the signal amplitude and

Figure 4.6: Tracking Error without Observer (Model in [3])

the controller achieves a figure of merit value of $F = 99.4\%$ on its own. In the calculation of this indicator, the integral was taken over the whole simulation time. Our required control energy, although not documented in the paper, is reasonable and does not exhibit unrealistically fast changes.

Even though the controller accomplishes decent control quality on its own, it does not match the performance documented in the experimental results of the paper. This changes, once the observer is employed. With the estimate of the additional dynamic state of the inner loop from the linear observer, the control circuit achieves a competitive tracking quality as shown in the error progression in Figure 4.7. As we can see, the maximum error is reduced by more than a factor of two and significantly narrowed. This shows in the increased merit $F = 99.9\%$. As an added bonus, the control energy is thereby reduced to less than a half by use of the observer.

A final remark shall be made about the reduced-order observer mentioned in [1]. While it did yield a small, yet noticeable, improvement over the performance of a full state observer in estimating the additional controller state, this enhancement

Figure 4.7: Tracking Error Using the Observer (Model in [3])

fails to appear in the context of a LuGre friction model. Tests with other parameter sets and the same plant model as in [3], as well as simulations based on other systems incorporating the LuGre model, exhibit the same exact tendency. Since the reduced observer does not improve our design even further in the presence of this very important friction model, it shall be left out in the discussion of further simulations.

The conclusion we can draw from the simulations in this section is that our goal of constructing a simple and versatile control method for mechanical systems is within a grasp. Even though the comparison of a pure simulation is always lacking compared to actual physical experiments, the results are promising and can be examined in more depth in the next chapters. Before we dive into the analysis of the control quality, however, we want to take a look at more complex models and the adaption of the design procedure to those systems.

4.2 More Complex System Dynamics

A logical next question is whether a more detailed system model would render the control architecture useless for application in a physical plant. We therefore

want to try to incorporate our controller in a rotatory electromechanical system, which also includes the model description of an electric motor. The model, which can be found in [28] with the set of parameters collected from [4], has the following nonlinear state space representation:

$$\begin{aligned}\dot{\theta} &= \omega \\ \dot{\omega} &= -\frac{K_{\Phi}}{J}i - \frac{1}{J}F_f \\ \frac{di}{dt} &= -\frac{R_a}{L_a}i - \frac{K_{\Phi}}{L_a}\omega + \frac{1}{L_a}u \\ y &= p,\end{aligned}\tag{4.30}$$

wherein F_f is once again determined by the LuGre friction model, including the additional state z . We denote $\omega \triangleq x_2$ and $i \triangleq x_3$ as the states of the inner loop, consisting of the motor dynamics subject to friction. To arrive at an appropriate design model, we first have to examine the pole constellation of the linearized LuGre equations. This part of the procedure is always identical when dealing with the dynamic friction model. Given the parameters $J = 0.048 \text{ kg m}^2$, $\sigma_0 = 100 \text{ Nm}$, $\sigma_1 = 10 \text{ Nms}$ and $\sigma_2 = 0.4 \text{ Nms}$, the eigenvalues of the system

$$\begin{aligned}\begin{bmatrix} \Delta\dot{\omega} \\ \Delta\dot{z} \end{bmatrix} &= \begin{bmatrix} -\frac{\sigma_1+\sigma_2}{J} & -\frac{\sigma_0}{J} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta\omega \\ \Delta z \end{bmatrix} + \begin{bmatrix} \frac{1}{J} \\ 0 \end{bmatrix} u \\ \Delta y = \Delta\omega &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta\omega \\ \Delta z \end{bmatrix},\end{aligned}\tag{4.31}$$

are $\lambda_1 = -206.58$ and $\lambda_2 = -10.08$. Obviously, these eigenvalues are far apart on the negative real axis and the behavior of the linearized model is predominantly determined by the pole λ_2 , which is closer to the imaginary axis. As we remember from Chapter 3 this case of parameter constellation requires the treatment with a traditional model reduction technique, more specifically the modal procedure proposed by Litz. Executing this method, which is explained in depth in said

chapter, gives us a first-order approximation of the linearized LuGre model, namely

$$\begin{aligned}\dot{\tilde{x}}_2 &= -10.0848 \tilde{x}_2 + 0.0516 u_v \triangleq A \tilde{x}_2 + b u_v \\ y &= \tilde{x}_2 \triangleq c \tilde{x}_2.\end{aligned}\tag{4.32}$$

Since we are dealing with an inner loop which contains additional motor dynamics, constructing the linear model is a little more elaborate than in the previous cases. Inserting the approximation back into the system dynamics results in the following linear representation of the inner loop:

$$\begin{aligned}\begin{bmatrix} \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} &= \begin{bmatrix} A & K_\Phi b \\ -\frac{K_\Phi}{L_a} c & -\frac{R_a}{L_a} \end{bmatrix} \begin{bmatrix} x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L_a} \end{bmatrix} u_v \\ y &= \begin{bmatrix} c & 0 \end{bmatrix} \begin{bmatrix} x_2 \\ x_3 \end{bmatrix}.\end{aligned}\tag{4.33}$$

It can now simply be plugged into the design model and the rest of the design procedure can be carried out in perfect analogy to the systems considered before. Since we are now treating a system of higher order, only a few slight changes result for the process. We need to specify an additional real eigenvalue for the pole placement of the inner loop. It is, motivated by the selection of poles in [1], chosen as the zero of the normalized first-order Bessel polynomial besides the already familiar second-order zeros.

The desired eigenvalues are, again, scaled by the reciprocal of the inner loop settling time $T_{sv} = 0.2$ s, which is a quarter of the overall settling time T_s . The remaining two poles of the outer loop are then chosen as the zeros of the second-order Bessel polynomial, divided by T_s , just like in Section 4.1.1. A similar adaptation has to be made in the construction of the observer, which obviously also is of order three. Its poles are shifted to the left in the complex plane by a factor of three compared to the inner loop poles.

Obviously, the first part of the inner loop feedback vector $\mathbf{K}_v$ is now no longer a scalar but rather a two-dimensional vector $\mathbf{k}_{v,1}$, consisting of its first two entries.

Figure 4.8: Tracking Error without Observer (Model in [4])

Apart from a few differences in dimensionality, the construction of the control architecture is completely identical.

In accordance with the simulation results found in the paper, a rectangular wave of low amplitude (0.01 rad) is used as the reference trajectory. Without the observer in the loop, the controller achieves the tracking error plotted in Figure 4.8, which is satisfactory.

Again, introducing the observer yields a noticeable difference in tracking quality. Looking at the plot in Figure 4.9 we can see right away that the maximum error is reduced to less than a half. The overall tracking result compared to the linear reference can be inferred from Figure 4.10.

A remarkable fact is that, for different reference signals, the observer shines even more. As an example, if a sine wave reference is used, the state estimation reduces the peak error by a factor of three. This goes to show how effective and, ultimately, versatile this linear observer-based control concept seems to be. As a last side note before focusing on the implementation and merit of our controller, we want to see if its significance is limited to a specific mechanical system model

Figure 4.9: Tracking Error Using the Observer (Model in [4])

Figure 4.10: Output Progression (orange) and Reference Output (blue) for the LuGre Model in [4]

description. We will therefore be looking at plants of a different structure.

4.3 Plants with Position Feedback

A large impact on the system behavior is caused by adding feedback of the position state back to the calculation of the acceleration. This structure is encountered, for example, in [5] or [15] and the biggest letdown for us is that our control architecture does not seem to be applicable to these cases anymore. The obvious reason for that is that the velocity state, which is supposed to be the output of the inner loop, can no longer be treated in an isolated manner, since it depends on the position itself. Having a nested velocity and position loop is no longer feasible.

However, modifying the controller composition slightly to arrive at a fully cascaded dual-loop structure allows us to apply the general idea of the design and results in a properly functioning controller. Instead of an inner velocity loop and an outer position loop, both loops now have the position as their output. We want to take a closer look at the model presented in [5]. The plant at hand is a servo system that is driven by two separate motors via a gearbox. It is particularly interesting, because not only is the state space model considerably more complex than the ones encountered so far, but it also considers, aside from the friction effect, a model for backlash in the gears. Treating this nonlinear model will hopefully show us if the idea of our controller design is expandable.

The dynamics of the two driving electric motors and the gearing are given by the nonlinear differential equations

$$\begin{aligned}
 \dot{\theta}_1 &= \omega_1 \\
 \dot{\omega}_1 &= \frac{1}{J_1} (-b_1 \omega_1 + u_1 + w - T_1) \\
 \dot{\theta}_2 &= \omega_2 \\
 \dot{\omega}_2 &= \frac{1}{J_2} (-b_2 \omega_2 + u_2 - w - T_2),
 \end{aligned} \tag{4.34}$$

while the description of the load motor is

$$\begin{aligned}\dot{\theta}_m &= \omega_m \\ \dot{\omega}_m &= \frac{1}{J_m} (T_1 + T_2 - T_f).\end{aligned}\tag{4.35}$$

The friction torque T_f is only relevant in the load motor dynamics due to the fact that its impact is negligible at the comparatively high velocities of the driving motors. It is, again, based on the LuGre friction model. As can be seen, this system has two input variables, the two motor voltages u_1 and u_2 . The parameters J_i ($i = 1, 2, m$) stand for the driving motors' and load motor's moments of inertia and the value w is a bias torque used to maintain contact of the gears in the transmission during operation. The state vector

$$\mathbf{x}^T = [\theta_1 \quad \omega_1 \quad \theta_2 \quad \omega_2 \quad \theta_m \quad \omega_m]\tag{4.36}$$

consists of the angles and angular velocities of the respective subsystems. A novel aspect is the inclusion of backlash in the system dynamics. It is encountered in the terms

$$T_1 = m \, d(\theta_1, \theta_m) + n \, d(\omega_1, \omega_m) \text{ and } T_2 = m \, d(\theta_2, \theta_m) + n \, d(\omega_2, \omega_m)\tag{4.37}$$

and modeled by the nonlinear function

$$d(\tau_i, \tau_m) = \begin{cases} (\tau_i - \tau_m) + \alpha & , (\tau_i - \tau_m) < -\alpha \\ 0 & , -\alpha < (\tau_i - \tau_m) < \alpha \\ (\tau_i - \tau_m) - \alpha & , \alpha < (\tau_i - \tau_m). \end{cases}\tag{4.38}$$

To arrive at our linear design model, some simplifications have to be made. The first modification is obviously the reduction of the LuGre friction model. Since its linearization exhibits the weakly dampened pole pair

$$\lambda_{1,2} = -55.1786 \pm 58.1713 \, j,\tag{4.39}$$

resulting from the parameterization $J_m = 0.028 \text{ kg m}^2$, $\sigma_0 = 180 \text{ Nm}$, $\sigma_1 = 3 \text{ Nms}$ and $\sigma_2 = 0.09 \text{ Nms}$, the best possible way to go about this is to reject the dynamic part of the model including the additional state. This results in the following approximation of the friction model in the load motor dynamics:

$$F_f \approx \sigma_2 \omega_m. \quad (4.40)$$

To cope with the backlash nonlinearity, we assume that the dead zone is narrow enough to neglect it. Assuming $\alpha \approx 0$, we can formulate a linear plant model as a groundwork for our controller design. It is given by the following system equations:

$$\begin{aligned} \dot{\mathbf{x}} &= \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -\frac{m}{J_1} & -\frac{b_1+n}{J_1} & 0 & 0 & \frac{m}{J_1} & \frac{n}{J_1} \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{m}{J_2} & -\frac{b_2+n}{J_2} & \frac{m}{J_2} & \frac{n}{J_2} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{m}{J_m} & \frac{n}{J_m} & \frac{m}{J_m} & \frac{n}{J_m} & -\frac{2m}{J_m} & -\frac{2n+\sigma_2}{J_m} \end{bmatrix}}_{\mathbf{A}} \mathbf{x} + \underbrace{\begin{bmatrix} 0 & 0 \\ \frac{1}{J_1} & 0 \\ 0 & 0 \\ 0 & \frac{1}{J_2} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}}_{\mathbf{b}} \mathbf{u} \\ y = \theta_m &= \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}}_{\mathbf{c}} \mathbf{x} \end{aligned} \quad (4.41)$$

with the control vector $\mathbf{u}^T = [u_1 \ u_2]$. Because the bias torque w is small and does not influence the stability of the linear model, we omit it during the design procedure. Note also that the input matrix $\mathbf{b}$ is now actually a 6×2 matrix and not a vector anymore.

The aforementioned problem becomes evident now. While in previous cases the derivative of the velocity did not depend on the position, it does in the present case. This manifests itself in the respective entries of the system matrix above, $[\mathbf{A}]_{2,1}$, $[\mathbf{A}]_{4,3}$ and $[\mathbf{A}]_{6,5}$, which are nonzero. Constructing a design model which includes only the velocity equations or even only the load motor velocity, is therefore impossible. Instead, we plug the complete system of linear differential equations

into the design model in order to develop an inner position control loop. The design matrices have the same exact form as the ones already dealt with in Sections 4.1.1, 4.1.2 and 4.2:

$$\mathbf{A}_{di} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ b_a \mathbf{c} & A_a \end{bmatrix}, \mathbf{b}_{di} = \begin{bmatrix} \mathbf{b} \\ 0 \end{bmatrix}. \quad (4.42)$$

We opt for an integrator as the additional dynamics again and choose our desired pole locations as an appropriately scaled combination of Bessel poles:

$$\mathcal{S}_i = \{-194, -17.62 \pm 10.17 j, -35.24 \pm 20.35 j, -52.87 \pm 30.52 j\}. \quad (4.43)$$

Performing the first step in the design procedure results in the closed inner loop described by the same model structure that was already derived in the previous sections and that was discussed in Chapter 2. It has the form

$$\begin{aligned} \begin{bmatrix} \dot{\mathbf{x}} \\ \dot{x}_{ai} \end{bmatrix} &= \underbrace{\begin{bmatrix} \mathbf{A} - \mathbf{b}\mathbf{k}_{1,i} & \mathbf{b}\mathbf{k}_{2,i} \\ b_a \mathbf{c} & A_a \end{bmatrix}}_{\mathbf{A}_{ci}} \begin{bmatrix} \mathbf{x} \\ x_{ai} \end{bmatrix} + \underbrace{\begin{bmatrix} \mathbf{0} \\ b_a \end{bmatrix}}_{\mathbf{b}_{ci}} u_i \\ y &= \underbrace{\begin{bmatrix} \mathbf{c} & 0 \end{bmatrix}}_{\mathbf{c}_{ci}} \mathbf{x}, \end{aligned} \quad (4.44)$$

essentially the same constellation as the closed position loop of the nested structure we worked on before. The feedback matrix has to be partitioned in the matrices $\mathbf{k}_{1,i}$, consisting of the first six columns and $\mathbf{k}_{2,i}$, the last column, in analogy with the familiar design procedure. However, instead of leaving the position loop as it is, it serves as the foundation for another, cascaded, position control loop on top, which is complemented by another integrator. Our design model for the outer loop in this instance results from the aggregation of the closed inner loop above and the additional dynamics:

$$\mathbf{A}_{do} = \begin{bmatrix} \mathbf{A}_{ci} & \mathbf{0} \\ b_a \mathbf{c}_{ci} & A_a \end{bmatrix}, \mathbf{b}_{do} = \begin{bmatrix} \mathbf{b}_{ci} \\ 0 \end{bmatrix}. \quad (4.45)$$

The only thing left to do now is to perform the same pole placement procedure on the outer loop, where the set $\mathcal{S}_o$ of desired pole locations is completed by an eigenvalue at -388 . This finalizes the controller design.

Constructing the observer is as straight-forward as it was in Sections 4.1.1, 4.1.2 and 4.2. Based on the state space model of the closed inner loop, $(\mathbf{A}_{ci}, \mathbf{b}_{ci}, \mathbf{c}_{ci})$, we perform pole placement to shift the eigenvalues of the linear observer. Their desired locations are equal to the poles of the inner loop in $\mathcal{S}_i$, scaled by a factor of three.

We are now prepared for a simulation test of the control quality. The experimental setup is identical to the ones before: the nonlinear system model with backlash and LuGre friction dynamics is controlled with the linear dual-loop controller designed above. Based on the work in [33] we choose the dead zone width as $2\alpha = 1$ rad. The remaining parameters used in the simulations are

$$\begin{aligned} J_1 &= J_2 = 0.185 \text{ kg m}^2, b_1 = b_2 = 1.2 \text{ Nms}, \\ m &= 560 \text{ Nm}, n = 0.15 \text{ Nms}, \\ F_c &= 0.28 \text{ Nm}, F_s = 0.4 \text{ Nm and } \omega_s = 0.01 \frac{\text{rad}}{\text{s}}. \end{aligned} \quad (4.46)$$

In line with the previous tests, we first exclude the observer to examine the capabilities of the controller first.

Recreating the reference signal used in the paper, which is a sine wave of amplitude 0.5 rad and frequency 0.2 Hz, the tracking performance is investigated. As we can see from the plot in Figure 4.11, the control quality is already decent. A distinct difference to the previous tests lies in the articulate peak of the output progression at the beginning of the simulation, while the output of the linear reference model is smooth. This deviation is caused by the initial gap between the gears in the transmission that has to be closed before the controller can function properly. To that end, a small constant bias torque $w = 50$ Nm is applied to the

Figure 4.11: Output Progression (blue) and Reference Output (orange) for the Model in [5]

driving motors, which ensures contact between the gears. Without the bias torque the tracking quality deteriorates considerably, as can be told from the curves in Figure 4.12.

As we would expect, the addition of the observer improves the merit of the control. Not only does the estimation of the additional dynamic state of the inner loop decrease the peak error caused by the friction nonlinearity from around seven percent of the reference signal amplitude to less than three, it also mitigates the initial error caused by the gearing backlash. Furthermore, with the reduced error, the control quality becomes competitive with the documented results from the original paper. The conclusion we can draw from this is that the slight modification we made to apply our control strategy to systems of the structure at hand, where the position is fed back into the calculation of the acceleration, is promising with respect to the tracking quality. There are, however, some important remarks to be made at this point.

First of all, the design procedure took a lot of tinkering with the settling time that is demanded from the closed loop. Basing the design procedure on the

Figure 4.12: Output Progression (blue) and Reference Output (orange) without Bias Torque (Model in [5])

slightly larger settling time $T_s = 0.27$ s instead of the ideal $T_s = 0.23$ s results in considerably impaired tracking as seen from the plot in Figure 4.13.

Even though the simulation runs smoothly and the closed loop is robust across a sufficient bandwidth, given the correct T_s was determined, this sensitivity to the settling time is a new issue this system structure brings along. Since a similar effect was encountered during tests based on the model in [15], we have to assume that this is a problem linked to the structure of the plant, which poses a significant drawback compared to the models without position feedback.

Another obstacle is the newly added dead zone nonlinearity. While the bias torque successfully reduces the influence of backlash on the control loop, the settling time has to be tuned appropriately to ensure a sensible trade-off between the compensation of friction and overcoming backlash in the gearbox. This makes the design process more elaborate than that in previous cases.

Yet, all in all, the adaption of our control architecture to accommodate a different plant structure and even another form of nonlinear effect was successful. Under some more attention the control concept could be refined to apply to this

Figure 4.13: Output Progression (blue) and Reference Output (orange) with Sub-par Settling Time (Model in [5])

kind of system more smoothly.

4.4 Stability of the Control Loop

Concluding this chapter, a final remark about an important issue has to be made. Throughout the controller design, we never examined the stability of the closed loop including the friction nonlinearity. Even though the design based on the linearized plant is of course geared towards producing a stable and dynamically beneficial behavior, this does not formally prove stability for the controller in the loop with a nonlinear plant. This holds true, in particular, for this case, where we are constructing a controller based on a significantly reduced system model.

To thoroughly prove stability for our controller design in a conventional sense, one would have few alternatives to applying Lyapunov theory to the system. It is outlined in [16]. Similar approaches were already chosen, for example, in [6], [19] and [29], either to ensure or to demonstrate stability of a control concept. There are, however, some fundamental issues with a formal proof of stability, which make it difficult to perform and rather pointless for our purpose.

First and foremost, the stability theory of nonlinear systems, which revolves mostly around Lyapunov's theorems, requires relatively well-behaved system dynamics to be easily applicable. Since in our case the model equations are non-differentiable, it is difficult or even impossible to find a Lyapunov function, which has to be continuously differentiable. Generally a proof of Lyapunov stability is not trivial to conduct and relies heavily on the patience and experience of the control engineer.

On top of that, Lyapunov's stability theorem is sufficient, not necessary. Just because there is no suitable candidate for a Lyapunov function, Lyapunov stability is by no means precluded. These issues make the proof complicated, but there are some other aspects, which plainly make it superfluous for our case.

In contrast to the stability theory of linear systems, the stability of a nonlinear systems is less expressive. Since there is no such concept as pole positions in the strict sense, aspects like eigenfrequency, overshoot or settling time have no analytical equivalent in the nonlinear realm. More importantly though, Lyapunov theory does not reveal a stability region in a straight-forward manner. This is the biggest issue, because even if a proof of stability existed, it would be difficult to determine what state trajectories would yields stable system behavior.

The bottom line of this section is that, from a practical standpoint, formal stability analysis for our controller design is relatively unsuitable. Instead, we want to carefully inspect the system behavior and its deterioration in the face of uncertainties and disturbances. This is an essential facet of Chapter 5.

CHAPTER 5

Comparison with Nonlinear Control Concepts

After ascertaining that our linear controller can cope with static and dynamic friction effects in system models taken from the literature, an important next step is to compare its performance with the merit of nonlinear control strategies. As discussed thoroughly in Chapter 1, there is a number of different ways to address friction compensation. In applications that require high accuracy, these almost always involve nonlinear system theory in the form of a state observer or the actual controller. To show that our idea is a competitive alternative to these approaches, we want to take a closer look at the construction of the controllers and contrast their performance with that of ours.

Similarly to the methodology in Chapter 4, we are trying to get a grasp of models from literature, for which a nonlinear controller has been designed that successfully compensates friction. Finding sources that sufficiently document the design procedure and present controllers that we can recreate with reasonable effort is a challenge in and of itself. There are, however, two methods presented in the studied material that provide a good foundation for a comparison, because they meet those requirements. On top of that, the systems under consideration choose fairly different strategies to deal with friction compensation. They shall be complemented by a third nonlinear controller we want to design from ground up, to get a better understanding of problems arising from nonlinear controllers.

5.1 Output Feedback Controller with Nonlinear Friction Compensator

5.1.1 Controller Design

The first controller we want to examine is an output feedback controller with superimposed friction compensator. It employs a nonlinear friction observer to es-

timate the unmeasurable friction force as well as an additional high-gain observer to determine the state variables. Its design can be found in [6].

Starting from the plant model of a simple rotatory single-mass system,

$$J\ddot{\varphi} = k_c u - M_f, \quad (5.1)$$

where J is the moment of inertia, M_f is the moment generated by friction and k_c is the gain of the input voltage u , the first objective is to construct a controller based on output feedback, which achieves the dynamic requirements. For the design of that controller, friction is neglected. Given the reference trajectory r , the error dynamics are defined as

$$\begin{aligned} \dot{e}_1 &= e_2 \\ e_1 &= \varphi - r \end{aligned} \quad (5.2)$$

and the goal is to achieve

$$\dot{e}_2 + 2\eta\omega_0 e_2 + \omega_0^2 e_1 \approx 0, \quad (5.3)$$

where η and ω_0 are adjusted appropriately. To that end, the controller is chosen as

$$u = \frac{\text{sat} \left(\hat{J} (\ddot{r} - k_p (\hat{x}_1 - r) - k_d (\hat{x}_2 - \dot{r})) \right)}{\hat{k}_c}. \quad (5.4)$$

The parameters $\hat{J}$ and $\hat{k}_c$ result from the identification of the plant, while the estimates of the state variables $\hat{x}_1$ and $\hat{x}_2$ originate from the high-gain observer

$$\begin{aligned} \dot{\hat{x}}_1 &= \hat{x}_2 + 2\frac{\varphi - \hat{x}_1}{\epsilon} \\ \dot{\hat{x}}_2 &= \frac{\varphi - \hat{x}_1}{\epsilon^2}, \end{aligned} \quad (5.5)$$

wherein $\epsilon > 0$ is another design parameter to be tuned properly. The design goal in Eq. 5.3 is attained employing the controller in Eq. 5.4. If friction is considered

in the process, however, the error is no longer small enough to comply with the requirements. The nonlinear friction effect is described by a modified form of the LuGre model, namely

$$\begin{aligned} M_f &= z + \epsilon_0 \sigma_1 \dot{z} + \sigma_2 \dot{\varphi} \\ \epsilon_0 \dot{z} &= \dot{\varphi} - \frac{|\dot{\varphi}|}{g(\dot{\varphi})} z, \end{aligned} \quad (5.6)$$

ϵ_0 being the reciprocal of the parameter σ_0 . The main difference to the conventional form of the LuGre model lies in the nonlinear function $g(v)$, which now depends on the direction of movement and is therefore defined as:

$$g(v) = \begin{cases} F_{c+} + (F_{s+} - F_{c+}) e^{-\left(\frac{v}{v_s}\right)^2} & , v > 0 \\ F_{c-} + (F_{s-} - F_{c-}) e^{-\left(\frac{v}{v_s}\right)^2} & , v < 0 \\ \frac{g(0+) + g(0-)}{2} & , v = 0. \end{cases} \quad (5.7)$$

To deal with these nonlinear friction phenomena, [6] proposes a friction compensator that relies on the estimate of a nonlinear observer. This observer is described by the equations

$$\begin{aligned} \hat{M}_f &= \hat{z} + \epsilon_0 \sigma_1 \dot{\hat{z}} + \sigma_2 \hat{v} \\ \epsilon_0 \dot{\hat{z}} &= \hat{v} - \frac{|\hat{v}|}{g(\hat{v})} \hat{z} + K(\hat{v}, e_1, \hat{e}_2). \end{aligned} \quad (5.8)$$

As can be seen, the observer is complemented by the additional nonlinear component $K(\hat{v}, e_1, \hat{e}_2)$. This function stems from the stability-based design procedure and is chosen as

$$K(\hat{v}, e_1, \hat{e}_2) = - \left(1 + \sigma_1 \frac{|\hat{v}|}{g(\hat{v})} \right) \frac{e_1 + 2 \frac{\hat{e}_2}{\omega_0}}{\rho \hat{J}}, \quad (5.9)$$

to ensure Lyapunov stability of the closed-loop system. ρ is a parameter that has to be tuned appropriately. The new states that appear in the observer equations are defined as

$$\hat{v} = \text{sat}(\hat{x}_2)$$

Figure 5.1: Controller Structure Proposed in [6]

$$\hat{e}_2 = \hat{v} - \dot{r}. \quad (5.10)$$

According to the paper, the saturation functions in Eqs. 5.4 and 5.10 are used to prevent stability issues caused by the “peaking effect”. For our simulations they are practically irrelevant, since the saturation does not set in for the small control variable values needed to achieve the low amplitudes of our reference progression. The nonlinear compensator

$$u = \frac{\text{sat} \left(\hat{J} (\ddot{r} - k_p e_1 - k_d \hat{e}_2) + \hat{M}_f \right)}{\hat{k}_c}, \quad (5.11)$$

combined with the high gain observer in Eq. 5.5, the friction observer in Eq. 5.8 and the nonlinear function in Eq. 5.9 complete the controller design. Its essential structure is outlined in Figure 5.1.

Choosing the parameter values from the paper, we can address the comparison to our novel controller design. The values documented in the paper are

$$\begin{aligned} J &= 0.095, k_c = 2.5, \\ \epsilon_0 &= 0.01, \sigma_1 = 1.5, \sigma_2 = 0.004, \\ F_{c+} &= 0.023k_c, F_{c-} = 0.021k_c, F_{s+} = 0.058k_c, F_{s-} = 0.052k_c, \\ \epsilon &= 0.01, \eta = 0.7, \rho = 10^6 \text{ and } \omega_0 = 1. \end{aligned} \quad (5.12)$$

While the used parameter values are documented well, their units are not. Since there are some ambiguities in the choice of units, particularly for the controller

gains, we omit them in this case. A minor guess also has to be made as to what Stribeck velocity v_s was used in the design procedure. Assuming $v_s = 0.075$ the parameter selection is complete and a model can be set up for simulation. Before the experiments begin, however, our own controller has to be designed.

The construction of our controller is completely identical to the one described in Section 4.1.2. Despite the modification of the LuGre friction model in the system at hand, the ideal reduction we can make as an approximation of the inner loop is

$$\ddot{\varphi} = -\frac{\sigma_2}{J}\dot{\varphi} + \frac{k_c}{J}u. \quad (5.13)$$

Our subsequent controller design follows the same procedure that we have already discussed in detail. Upon choosing the inner loop eigenvalues as

$$\mathcal{S}_i = \{-32.424 \pm 18.72 j\}, \quad (5.14)$$

placing the outer loop poles in

$$\mathcal{S}_o = \{-32.424 \pm 18.72 j, -8.106 \pm 4.68 j\} \quad (5.15)$$

and opting for the observer poles

$$\mathcal{S}_{obs} = \{-97.272 \pm 56.16 j\}, \quad (5.16)$$

the controller has exactly the same form as the one designed in Section 4.1.2.

Before we turn our attention to the actual simulation, some important remarks have to be made about the circumstances of the experiments and the comparison. It is perfectly evident that the control algorithm proposed in [6] relies on the first and second time derivative of the reference trajectory. A first test with actual differentiator blocks in SIMULINK failed due to stability issues, a problem that is plausible, because of the approximating nature of numerical differentiation. Instead, the reference progression is assumed to be known in an advance and its derivatives are simply fed into the system analytically.

Furthermore, comparing the tracking quality of this approach to a linear reference model is impossible, since not only the controlled plant, but also the controller and observer are inherently nonlinear. We have chosen this comparison as a benchmark test so far, because it is our goal to achieve approximately linear behavior in the closed control loop and there is also an unavoidable time delay in any physical system. It makes the direct comparison with a reference trajectory less meaningful than the comparison with a linear reference model.

The main concept the simulations are supposed to capture, however, is that the control concepts compensate friction. The performance of the controller in Eq. 5.11 is therefore compared to a closed control loop without friction ($M_f = 0$). It is controlled with the initial algorithm described by Eq. 5.4, which was designed under the assumption that no friction is present. This reference system also involves the high-gain observer and the analytical derivative of the reference trajectory. By comparing these two systems, we recreate the framework used in our previous simulations to focus on the ability of the controller to suppress friction effects.

5.1.2 Comparison with the Novel Control Architecture

A first question we want to answer, is whether our control algorithm yields a tracking quality that is competitive with that achieved by the controller described in [6]. To this end, we assume that both approaches have perfect knowledge of the plant model, which means that there are no parameter uncertainties and that there is no disturbance in the system.

Orienting on the experiments conducted in the paper, we choose the angle progression

$$r(t) = \sin(\pi t) \tag{5.17}$$

as the reference signal. Subject to this input, the nonlinear controller shows the

Figure 5.2: Output Progression for the Controller in 5.11

behavior plotted in Figure 5.2.

Evidently, the control system needs around a third of the simulation time to settle in and function properly. In this respect, the simulation deviates from the experimental results documented in the paper, which do not exhibit this behavior. The discrepancy could be due to improper tuning of some of the design parameters or initial conditions of the control system. We want to focus on the latter phase of the control operation, since it will yield the most significant information. If we consider the error between the system subject to friction and the unperturbed reference model, we can see that the nonlinearity causes a peak error of around 6.5×10^{-3} rad in this phase. The error progression is shown in Figure 5.3.

We now want to compare the performance of this controller with that of our own, in simulation. In the following investigations the observer is permanently switched on, since its influence has proven to be consistently beneficial. Applying the same reference signal to the control system yields a maximum error of 5×10^{-4} rad. It is about one order of magnitude smaller than the tracking error caused by the friction effect in Figure 5.3.

Figure 5.3: Difference between Actual System Output and Unperturbed Reference System (Model in [6])

When looking at the error plot in Figure 5.4, another advantage over the nonlinear controller becomes apparent. The integrated absolute error

$$e_{int} = \int_0^{T_{sim}} |\varphi - \varphi_{ref}| dt, \quad (5.18)$$

visually speaking the area enclosed by the time axis and the error curve, is smaller by a factor of almost 45.

This aspect is inspired by the figure of merit proposed in [1] and tells us, for how long the controlled variable, the angle φ , is far off from the reference. If the peaks in the error progression are narrow, the influence of the deviation does not manifest for extended amounts of time, which is preferable. Instead of comparing the controllers based on Eq. 2.19, the indicator above is used, because, for the simulations in this chapter, the figure of merit yields values so close to one that they lose most of their informative value.

The control variable for both controllers has a periodic progression which resembles a sinusoid in phases of higher velocity, since in these intervals the nonlinear friction effect only requires minor controller action. In either case, the control sig-

Figure 5.4: Tracking Error in System with Linear Controller (Model in [6])

nal reaches a maximum amplitude of about 0.4, but the nonlinear controller shows a small but extremely narrow peak when the system crosses zero velocity.

As a next step, we want to make an assessment of the control quality subject to parameter uncertainties and disturbances. Even the best controller will fail in practice, if it relies too heavily on a perfectly accurate plant model, so robustness is an important requirement for any application. Disturbance rejection is another concern. Mechanical systems like, for example, electric motors are always exposed to rapidly varying loads and therefore have to react quickly to unpredictable excitation. We want to address both issues separately.

Based on our experience with the LuGre friction model, a sensible prediction about the robustness of our controller design should be that parameter mismatches do not spoil the tracking quality. After we went through the whole process of linearizing the model and then reducing the resulting state space representation, the degree of approximation is high enough to allow for more discrepancy between the reference model and the true system without any dramatic consequences.

Unfortunately, we are dealing with nonlinear systems and the classical theory

of robust control is not applicable in this setting. Instead, we want to systematically expose the controller to a faulty system identification. This is achieved by keeping the parameter values that are used in the controller design fixed and modifying the true plant parameters.

Judging the robustness of our controller design by its performance rather than by conventional measures of robustness is in line with our overall approach. It will become obvious soon that it takes a lot for the control loop to actually become unstable due to uncertainties. A more interesting question for us to answer is, how much the tracking quality is impaired under these circumstances.

Right away, we can see that the nonlinear controller in Eq. 5.11 is highly susceptible to errors due to parameter mismatches. Exposing the controller to the only slightly varied true plant parameters

$$\begin{aligned}
J_{true} &= 0.095, k_{c,true} = 2.5, \\
\epsilon_{0,true} &= 0.0083, \sigma_{1,true} = 1.4, \sigma_{2,true} = 0.003, \\
F_{c+,true} &= 0.021k_c, F_{c-,true} = 0.023k_c, F_{s+,true} = 0.055k_c, F_{s-,true} = 0.049k_c \\
v_{s,true} &= 0.065
\end{aligned} \tag{5.19}$$

already yields a visible decline in tracking quality. The peak tracking error is increased almost tenfold, which can be inferred from the plot in Figure 5.5.

On the other hand, if our linear controller has to deal with the imprecise parameter identification above, the control quality is much better. Since the controller is designed based on the values of J , k_c and σ_2 alone, the slight modification in the true parameters goes unnoticed in the simulation. Only when the viscous friction parameter is changed by more than an order of magnitude, or the measurement of the moment of inertia is erroneous, the discrepancy becomes larger and tracking becomes compromised. This proves our initial guess that knowing the plant model with high accuracy is not necessary to achieve excellent control

Figure 5.5: Tracking Performance of Nonlinear Controller in Eq. 5.11 Subject to Parameter Uncertainties (blue) Compared with Reference (orange)

quality.

The final question we want to clarify in our investigations is, which system copes better with unforeseen disturbances. To achieve such an effect, the parameterization is, again, assumed to be known perfectly. However, the moment of inertia is periodically increased to 500 times its original value, to pretend that the load varies rapidly. Mathematically speaking the moment of inertia is now a time-dependent function that consists of the constant original value and a pulse wave with amplitude $500 J$, period 0.9s and a duty cycle of five percent:

$$J_{true}(t) = \begin{cases} 500 J & , k < t < k + 0.045 \text{ s}, \quad k \in \mathbb{Z} \\ J & , k + 0.045 \text{ s} < t < k + 1 \text{ s}, \quad k \in \mathbb{Z}. \end{cases} \quad (5.20)$$

If this disturbance is applied to the system at hand, with the nonlinear controller given in Eq. 5.11 in the loop, the tracking quality is severely compromised. Instead of the high accuracy in the first test, the error now reaches peak values of up to ten percent of the reference trajectory. This can be inferred from the contrast between the reference progression and the actual output in Figure 5.6.

Figure 5.6: Nonlinear Controller Subject to Disturbance (blue) Compared with Reference (orange) (Model in [6])

In contrast to that, our novel controller design is, again, almost unaffected by the load disturbance, which can be seen in Figure 5.7. The peaks in the error progression reach a maximum of about 1.1×10^{-2} rad and once more the deviation of the controlled variable with respect to the reference output is limited to significantly shorter time intervals. This shows in the magnitude of the integrated absolute error since it is decreased by a factor of 45 compared to the nonlinear control concept.

There is a point to be made for a disturbance of this dimension being unrealistically large. However, even if the system will never have to face unpredictable load fluctuations like this during operation, it illustrates the immense capabilities of our control approach in the face of disturbances. In this regard, the simulations reveal another advantage of the novel design over the nonlinear controller in [6].

As a final statement, we can conclude that our linear control algorithm yields superior tracking quality compared to the observer-based controller in Eq. 5.11. Especially in the face of parameter uncertainties and disturbances, the novel approach performs significantly better and exhibits predictable behavior, in contrast

Figure 5.7: Tracking Error of Linear Controller Subject to Disturbance (Model in [6])

to the severely impaired merit of the nonlinear controller. The most useful insight we gain through our simulations is that, indeed, for the construction and proper functioning of our controller, the system model does not have to be known perfectly well.

On the other hand, the approach in [6] relies heavily on a sufficiently accurate parameterization of the system. A final remark has to be made about the design procedure. While dimensioning the nonlinear controller requires an elaborate process of identifying a set of ten parameters fairly accurately and tuning another six, our linear approach is decently simple. One only needs to measure three plant parameter values in this particular instance, to arrive at a highly capable controller designed by well-known procedures.

Note however, that we are still in the realm of simulations. Problems that may arise in the practical implementation of such a controller can not be considered in this context. In light of the very promising results of the experiments, these issues are not anticipated to be dramatic.

5.2 Friction Observer and State Feedback Control

5.2.1 Controller Design

We want to take the examination of our proposed controller design one step further, by comparing it to another concept from literature. The model in question is presented in [7]. Its concept is based on a linear observer for the state variables that is accompanied by another model-based nonlinear observer, which is used to estimate the friction force. From the linear observer, the forecasted variables are used in a state feedback controller.

As the plant, a simple translatory single-mass system subject to the LuGre friction model is considered. Given the nonlinear equations

$$\begin{aligned}\dot{p} &= v \\ \dot{v} &= -\frac{1}{m}F_f + \frac{1}{m}u \\ \dot{z} &= v - \sigma_0 \frac{|v|}{g(v)}z \triangleq v - \sigma_0 \alpha(v)z \\ F_f &= \sigma_0 z + \sigma_1 \dot{z} + \sigma_2 v,\end{aligned}\tag{5.21}$$

with the state vector $\mathbf{x}^T = [p, v, z]$, the state-dependent model

$$\begin{aligned}\dot{\mathbf{x}} &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\frac{\sigma_1 + \sigma_2}{m} & -\frac{\sigma_0 - \sigma_0 \sigma_1 \alpha(v)}{m} \\ 0 & 1 & -\sigma_0 \alpha(v) \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ \frac{1}{m} \\ 0 \end{bmatrix} u \\ y &= [1 \ 0 \ 0] \mathbf{x}\end{aligned}\tag{5.22}$$

is formulated. Based on this system model, the paper proposes the equations of a nonlinear observer. Extended by the representation of the friction effect, $\hat{F}_f$, and an additional correctional term $d\hat{F}_f$, this yields the form of a descriptor system:

$$\begin{bmatrix} \dot{\hat{p}} \\ \dot{\hat{v}} \\ d\hat{F}_f \\ \dot{\hat{z}} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{\sigma_{2,o}}{m_o} & -\frac{1}{m_o} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -\sigma_{0,o} \alpha_o(\hat{v}) \end{bmatrix} \begin{bmatrix} \hat{p} \\ \hat{v} \\ d\hat{F}_f \\ \hat{z} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m_o} \\ 0 \\ 0 \end{bmatrix} u' + \begin{bmatrix} l_1 \\ l_2 \\ l_3 \\ 0 \end{bmatrix} (p - \hat{p})$$

$$\begin{bmatrix} \hat{p} \\ \hat{v} \\ d\hat{F}_f \\ \hat{F}_f \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & \sigma_{1,o} & 0 & \sigma_{0,o} - \sigma_{0,o}\sigma_{1,o}\alpha_o(\hat{v}) \end{bmatrix} \begin{bmatrix} \hat{p} \\ \hat{v} \\ d\hat{F}_f \\ \hat{z} \end{bmatrix}. \quad (5.23)$$

In these equations, the index o signifies that the parameters and functions are those used in the observer design. As we have discussed in Section 5.1.2, their values or functional form do not necessarily coincide with those encountered in the true model. This will be highlighted later. The variable $d\hat{F}_f$ is used to make dynamic adjustments to the estimated state variables based on the observer error $(p - \hat{p})$. Since the dynamic part of the observer, described by the variables $\hat{p}$, $\hat{v}$ and $d\hat{F}_f$, is linear, its poles can be placed by traditional design methods through modification of the parameters l_1 , l_2 and l_3 .

The only thing left to do, is to use the modeled system states for feedback control in the loop. The estimates of position and velocity are fed back to the input via the controller gains k_1 and k_2 , respectively, while the friction force estimate is used as a feed-forward compensator. Consequently, the controller has the structure

$$\begin{aligned} u &= u' + \hat{F} \\ u' &= k_1 r - k_1 \hat{p} - k_2 \hat{v} + d\hat{F}_f, \end{aligned} \quad (5.24)$$

r being the reference signal which, is also amplified by the constant gain k_1 .

Essentially, the control strategy presented in the paper is fairly simple. An estimate of the friction force based on the LuGre model is used to compensate friction, while the actual controller is fed with the state estimate of an augmented linear observer. The additional state in this observer, $d\hat{F}_f$, serves the purpose of mitigating the influence of the friction effect and model mismatches.

Upon inspection of the controller structure, one would assume that friction compensation should succeed perfectly, given that the plant model is known immaculately. This is because the friction force is compensated by its estimate, which

is fed forward into the control signal. The superimposed observer-based controller ensures that parameter mismatches do not deteriorate the tracking quality fundamentally and influences the dynamic behavior of the closed loop. Under the assumption of perfect state estimation, tracking should be flawless.

For the parameters in the plant at hand, the following values are used:

$$\begin{aligned}\sigma_0 &= 2940 \frac{\text{N}}{\text{m}}, \sigma_1 = 108 \frac{\text{kg}}{\text{s}}, \sigma_2 = 0 \frac{\text{kg}}{\text{s}}, \\ F_c &= 2.94 \text{ N}, F_s = 5.88 \text{ N}, v_s = 0.001 \frac{\text{m}}{\text{s}} \text{ and } m = 1 \text{ kg}.\end{aligned}\tag{5.25}$$

The observer and controller parameters are chosen as

$$\begin{aligned}l_1 &= 9.4248 \times 10^2, l_2 = 1.7765 \times 10^5, l_3 = -6.6974 \times 10^6, \\ k_1 &= 3.9478 \times 10^3 \text{ and } k_2 = 8.7965 \times 10^1.\end{aligned}\tag{5.26}$$

The system is therewith set up for simulation. Following the same rationale as in Section 5.1.1, a reference system is constructed in which friction is omitted. In contrast to the preceding investigations in Section 5.1.2, this reference system is now linear. It has the simple form

$$\begin{aligned}\begin{bmatrix} \dot{p} \\ \dot{v} \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} u \\ u &= k_1 r - k_1 \hat{p} - k_2 \hat{v}\end{aligned}\tag{5.27}$$

for the plant and controller. For the observer, we arrive at the equally plain equations

$$\begin{bmatrix} \dot{\hat{p}} \\ \dot{\hat{v}} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{p} \\ \hat{v} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} u + \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} (p - \hat{p}).\tag{5.28}$$

Since we obviously want to compare the controller proposed in [7] not only to an unperturbed version of itself but also to our new approach, we have to apply our own design procedure to the plant in 5.21. Starting from the linearization of the

LuGre model, which has a pair of strongly dampened eigenvalues, we approximate the plant by dropping the nonlinear dynamics, to arrive at the inner loop equation

$$\dot{\tilde{v}} = -\frac{\sigma_2}{m}\tilde{v} + \frac{1}{m}u. \quad (5.29)$$

For brevity, we do not want to reiterate the controller design in its entirety. It is described in detail in Chapter 4 and the pole locations specified in 5.14, 5.15 and 5.16, scaled by a factor of five, are used in this case as well. This concludes the experimental setup.

5.2.2 Comparison with the Novel Control Architecture

According to the paper, the observer-based controller was tested in simulation not only with a sinusoid of frequency 2 Hz and amplitude 0.1 m, but also with a step reference of height 0.002 m. We want to address both cases. In line with Section 5.1.2, we want to assume perfect knowledge of the plant first. Applying the sinusoid to the system with the controller of Eq. 5.24 in the loop verifies our assumption that the controller perfectly compensates friction if it is perfectly modeled. The error between the true output of the system and the output of the linear reference system is in the range of 10^{-9} to 10^{-11} m and therewith purely caused by numerical rounding errors during simulation. Same holds, obviously, for the step input.

Naturally, and somewhat unsurprisingly, our own linear controller can not keep up with that level of perfection. Since it is based on the reduced version of a linearized plant model, there has to be some degree of deviation from the linear reference in the plant disturbed by nonlinear friction. Indeed, if we apply the sine wave specified above to our control system, we can inspect a small but noticeable error with a maximum of 2.5×10^{-4} m. The error plot in Figure 5.8 illustrates this behavior.

Figure 5.8: Tracking Error of Linear Controller Compared to Reference Output (Model in [7])

Similarly, if the system is subject to the step input, it shows an error as well. In this case, however, it is even more minute. The only interesting realization we can infer from this experiment relates to the consumed control variable. For both the observer-based controller proposed in [7] and our novel linear control concept, the control variable progression required to track the sine reference is almost identical. It is a periodic signal with a very distinct peak around zero velocity. A kindred correlation holds for the step input, although the nonlinear design peaks at a higher value due to the slightly higher demanded settling time. This is not fundamentally surprising. Because both controllers try to compensate friction occurring in the same plant and both yield good or even phenomenal results in doing so, the control signal they are providing ought to be very similar.

The next step now is to investigate how well the controllers cope with parameter uncertainties and disturbance. Again, we examine these aspects in isolation. Simulations with a modified parameter set are documented in [7], where the true parameters F_c , σ_0 and σ_1 are increased by 20 percent compared to the values used in the controller design. We want to start from there.

If in the loop with the observer-based controller in Eq. 5.24 the true plant parameters are changed accordingly, the control system does show a tracking error peaking at about 4.3×10^{-4} m, when the sinusoid is the reference. The controller therewith already falls short of our linear controller, which yields an error of 3.1×10^{-4} m at most.

Of course our design does not rely on any of the parameters that were modified, so the result is by no means astonishing. We want to try throwing the controllers off with even more drastically varied plant parameters, to get a fairer comparison between the two approaches. Rather than assuming larger flaws in the identification of the friction parameters that were already changed, we modify the ones that are actually used in the design of our controller. Using the set of true parameters

$$\begin{aligned} \sigma_{0,true} &= 3528 \frac{\text{N}}{\text{m}}, \sigma_{1,true} = 129.6 \frac{\text{kg}}{\text{s}}, \sigma_{2,true} = 10 \frac{\text{kg}}{\text{s}}, \\ F_{c,true} &= 3.528 \text{ N}, F_{s,true} = 5.88 \text{ N}, v_{s,true} = 0.0003 \frac{\text{m}}{\text{s}} \text{ and } m_{true} = 0.85 \text{ kg}, \end{aligned} \quad (5.30)$$

in which especially the viscous friction parameter σ_2 and the mass m are off by a considerable amount. Especially the assumption of a nonzero viscous friction coefficient, however, makes sense from a physical standpoint. The response of the observer-based controller to the sine input now already visibly differs from the output of the unperturbed reference model, yielding the error in Figure 5.9.

Examining this plot, we can see that the tracking quality is severely impaired. This results in a peak error of around 2×10^{-3} m, more than four times larger than the error in the first test. Meanwhile, our controller is almost unaffected by the modified plant parameters. In fact, the peak error is not noticeably larger than it was with the initial modifications to F_c , σ_0 and σ_1 . Similarly, the error of the controller in Eq. 5.24 when exposed to the step reference reaches a peak of more than 2×10^{-4} m, while our novel controller design yields a maximum error

Figure 5.9: Tracking Error of Observer-Based Controller with Parameter Uncertainties (Model in [7])

of 5.3×10^{-5} m.

We can therefore conclude that, in simulation, our control approach performs better in the face of parameter uncertainties. This advantage in robustness is particularly astonishing when we remind ourselves that the whole controller design is based on only two individual parameters which, on top of that, are decently simple to identify. On the other hand, it is no surprise that the tracking performance of our controller does not deteriorate any further if parameters are modified that are not used in the design anyway.

Of course we are assuming large uncertainties in the parameterization here. One could argue that, even in a worst-case scenario, the plant could be identified better. However, this examination is supposed to show just how well both controllers can deal with parameter mismatches. It has to be emphasized that the observer-based controller does by no means perform badly. Having said that, given more severe uncertainty in the identification of the plant, our linear controller yields superior tracking quality. This leaves us with a test of disturbance rejection.

To test the controller's reaction to unpredictable disturbances, we again simu-

Figure 5.10: True Output Value with Observer-Based Controller (blue) Subject to Disturbance and Reference Output (orange) (Model in [7])

late a rapidly increased load, just like in Section 5.1.2. We use the same superposing form of the load given in Eq. 5.20, where the moment of inertia is replaced by the mass. For this experiment, the parameter identification is again assumed to be flawless. It has to be recapitulated that load steps of this magnitude might not be realistic for any practical application of a friction compensation controller. The following simulation is purposely dealing with this extreme scenario to illustrate how the controllers react to the disturbance.

Right away, one can see that our linear controller no longer outshines the observer-based approach, when the load varies drastically. This can be inferred from the comparison of both controllers in the plots of Figure 5.10 and Figure 5.11, respectively.

Inspecting the error curve reveals that the peak error in both cases is almost identical. The more important insight here lies in the location of the maximum deviation of the output of the controlled plant from the reference model behavior. As can easily be seen in either of the plots, the largest error caused by the load steps is caused when load steps occur close to zero velocity. This is the case for

Figure 5.11: True Output Value with Linear Controller (blue) Subject to Disturbance and Reference Output (orange) (Model in [7])

$t = 0.9$ s or $t = 3.6$ s, for example, and means that the controller is struggling to track the reference trajectory properly. Here, both controllers yield a peak error of about $0.017 - 0.019$ m, almost 20 percent of the amplitude of the input signal.

Summing up, we can say that our novel linear controller again shows a considerable improvement over the approach originally proposed to deal with the friction effect. The observer-based design in [7] yields perfect tracking, as long as the nonlinear plant subject to friction is perfectly identified. However, its performance quickly deteriorates and becomes worse than that of our novel controller, once the plant parameters are not known without flaws anymore. Although exhibiting great robustness, the tracking error of the controller is larger than that achieved by our controller. When it comes to disturbance rejection, both controllers yield similar results. Tracking errors are particularly dramatic around zero velocity.

5.3 Input-Output Linearization Controller

Inspired by the idea of compensation in the previous section, we want to design our very own nonlinear controller, to highlight the issue of plant identification. Not

only do we want to get a grasp of the effect of erroneous parameter estimation, but also take a look at another controller design, to gain a better understanding of the whole procedure by performing it ourselves. As a naive idea for a nonlinear approach, input-output linearization comes to mind. The rationale here is to suppress all nonlinearities occurring in the plant, while imprinting an arbitrary linear behavior on the system. For a detailed description of the controller design and the source for the following outline see [16].

Starting from the control affine nonlinear plant equations of a single-mass system including the LuGre friction model,

$$\underbrace{\begin{bmatrix} \dot{p} \\ \dot{v} \\ \dot{z} \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} v \\ -\frac{\sigma_1 + \sigma_2}{m}v - \frac{\sigma_0 - \sigma_0\sigma_1 \frac{|v|}{g(v)}}{m}z \\ v - \sigma_0 \frac{|v|}{g(v)}z \end{bmatrix}}_{\mathbf{a}(\mathbf{x})} + \underbrace{\begin{bmatrix} 0 \\ \frac{1}{m} \\ 0 \end{bmatrix}}_{\mathbf{b}(\mathbf{x})} u$$

$$y = \underbrace{\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}}_{\mathbf{c}(\mathbf{x})} \mathbf{x}, \quad (5.31)$$

the first task is to determine the relative degree of the system. To this end we calculate the Lie derivative, which is defined as the gradient of a multivariable function multiplied with a vector field, therefore

$$L_{\mathbf{f}}h(\mathbf{x}) \triangleq \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}). \quad (5.32)$$

For Lie derivatives of higher degree, the abbreviatory notation

$$L_{\mathbf{f}}^2 h(\mathbf{x}) \triangleq L_{\mathbf{f}} L_{\mathbf{f}} h(\mathbf{x}) = \frac{\partial L_{\mathbf{f}} h(\mathbf{x})}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}),$$

$$L_{\mathbf{f}}^3 h(\mathbf{x}) \triangleq L_{\mathbf{f}} L_{\mathbf{f}} L_{\mathbf{f}} h(\mathbf{x}), \dots \quad (5.33)$$

is used, which can be understood as the repeated concatenation of a Lie derivative with itself.

The relative degree now is given by the order of the time derivative of the output equation, Eq. 5.31, which first depends on the control variable. To determine

the time derivatives of the output function $c(\mathbf{x})$, we employ the Lie derivatives of higher order:

$$\begin{aligned}
y &= c(\mathbf{x}) = p \\
\dot{y} &= L_{\mathbf{a}}c(\mathbf{x}) + \underbrace{L_{\mathbf{b}}c(\mathbf{x})}_{=0} u = v \\
\ddot{y} &= L_{\mathbf{a}}^2c(\mathbf{x}) + \underbrace{L_{\mathbf{b}}L_{\mathbf{a}}c(\mathbf{x})}_{\neq 0} u = -\frac{\sigma_1 + \sigma_2}{m}v - \frac{\sigma_0 - \sigma_0\sigma_1 \frac{|v|}{g(v)}}{m}z \frac{1}{m}u. \tag{5.34}
\end{aligned}$$

Not only is the relative degree essentially the order of the linear system dynamics we can demand from the closed control loop, but it is also the order of the nonlinear system dynamics that are controllable and observable from the outside.

Because the relative degree $\delta = 2$ is smaller than the system order, there is a first-order part of the system, which is not “visible” from the output or can not be influenced from the input. This is obviously particularly problematic if these internal dynamics, as they are called, are not stable. Since we will not be able to affect the internal dynamics, we have to keep their stability in mind. Even though their effect might not show at the output, they do have physical influence on the system, leading to destruction or failure of the surrounding structure at worst. Furthermore, their behavior can be determined by the complete system dynamics in unpredictable ways.

We want to treat the external system dynamics and the internal dynamics separately. As we have already found, the relative degree allows us to specify desired linear dynamics of order two. In general, these dynamics are given by the differential equation

$$\ddot{y} + \dot{y}a_1 + ya_0 = Vw \tag{5.35}$$

and can be achieved by using the control rule and prefilter

$$u(\mathbf{x}) = -r(\mathbf{x}) + v(\mathbf{x})w, \text{ where}$$

$$\begin{aligned}
r(\mathbf{x}) &= \frac{L_{\mathbf{a}}^2 c + a_1 L_{\mathbf{a}} c + a_0 c}{L_{\mathbf{b}} L_{\mathbf{a}} c} = \frac{-\frac{\sigma_1 + \sigma_2}{m} v - \frac{\sigma_0 - \sigma_0 \sigma_1 \frac{|v|}{g(v)}}{m} z + a_1 v + a_0 p}{\frac{1}{m}} \quad \text{and} \\
v(\mathbf{x}) &= \frac{V}{L_{\mathbf{b}} L_{\mathbf{a}} c} = \frac{V}{\frac{1}{m}}.
\end{aligned} \tag{5.36}$$

This controller lets the closed loop appear as a linear system from the outside, by eliminating the original nonlinear plant dynamics and defining a completely new linear behavior for the system. This has the obvious consequence of suppressing all original dynamics, even those that are beneficial for the performance of the system.

While the design of the controller in this case is decently easy, dealing with the internal dynamics is considerably more complicated. First of all, finding the equation of the internal dynamics in and of itself is not always simple. Even though in this one-dimensional case it is easy, it requires a high level of mathematical knowledge and involves determining an unambiguous transformation of the state coordinates. This transformation is referred to as a diffeomorphism. It has the form

$$\mathbf{t}(\mathbf{x}) = \begin{bmatrix} c(\mathbf{x}) \\ L_{\mathbf{a}} c(\mathbf{x}) \\ \vdots \\ L_{\mathbf{a}}^{\delta-1} c(\mathbf{x}) \\ t_{\delta+1}(\mathbf{x}) \\ \vdots \\ t_n(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} p \\ v \\ t_3(\mathbf{x}) \end{bmatrix}. \tag{5.37}$$

This transformation contains the external dynamic states in the first δ components and the additional $n - \delta$ states of the internal dynamics in the remaining entries. Basically, there are two requirements that are posed for this nonlinear transformation. As already mentioned, the transformation should be unambiguous. On top of that, it has to be continuously differentiable. Both of these conditions

are met if the Jacobian of $\mathbf{t}(\mathbf{x})$ is regular, or

$$\left| \frac{\partial \mathbf{t}(\mathbf{x})}{\partial \mathbf{x}} \right| = \left| \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \frac{\partial t_3}{\partial p} & \frac{\partial t_3}{\partial v} & \frac{\partial t_3}{\partial z} \end{bmatrix} \right| \neq 0. \quad (5.38)$$

Furthermore, the internal dynamics must not depend on the control variable, so

$$L_{\mathbf{b}} t_3(\mathbf{x}) \stackrel{!}{=} 0 \quad (5.39)$$

in this case.

All of this can be achieved by simply choosing $t_3(\mathbf{x}) = z$, which fulfills the requirements and yields the internal dynamics

$$\dot{t}_3 = \dot{z} = v - \sigma_0 \frac{|v|}{g(v)} z. \quad (5.40)$$

We have already found in Section 3.2.2 that the additional dynamic state z of the LuGre friction model poses a problem to our controller design by causing an undesired pole-zero cancellation with an invariant zero. The occurrence of this state z in the internal dynamics of the nonlinear system is therefore by no means surprising. It merely extends our notion of internal dynamics to the nonlinear realm.

Inspecting Eq. 5.40 reveals the most problematic drawback of the input-output linearization approach in our case. The analysis of stability by use of Lyapunov theory is hard or even unfeasible because of the non-differentiability of the LuGre model. We have to rely on a simulation of the internal behavior of the system.

With the linearizing control law in Eq. 5.36 derived and the internal dynamics sorted out, we are ready to turn our attention to experiments. Before doing so, however, some important remarks have to be made. The predominant goal of this section is to display the importance of accurate system identification for the proper functioning of nonlinear controllers. Our controller above, that was designed from

scratch, does not claim to perform well in comparison with any tried and tested control concept. Rather than trying to improve its merit using common approaches to make input-output linearization more robust towards modeling mismatches, we want to leave it as it is and point out the issues pertaining to robustness.

The specific plant model, the experiments are based on, can be found in [7]. It was already the foundation of the comparison in Section 5.2. Having already calculated the general form of the control law, we only have to specify the desired linear dynamics. Since we have already attained great results with the choice of Bessel poles for the closed loop, we select the zeros of the second order polynomial, scaled by the settling time $T_s = 0.1$ s, as the poles of the linear closed loop. This yields the inhomogeneous differential equation

$$\ddot{y} + 81.06 \dot{y} + 2190.2 y = 2190.2 w, \quad (5.41)$$

with the prefilter $V = a_0$ chosen to achieve stationary accuracy. Plugging $a_1 = 81.06$ and $a_0 = 2190.2$ into our controller completes the design procedure. Just like in Sections 5.1 and 5.2, we need an unperturbed reference model to compare the control quality to. In this case it is simply given by the transfer function resulting from the desired linear dynamics in Eq. 5.41:

$$G(s) = \frac{2190.2}{s^2 + 81.06s + 2190.2}. \quad (5.42)$$

We are therewith finally equipped for our simulations. Almost traditionally at this point, we begin with a perfectly identified plant model. Similarly to the non-linear compensator in Section 5.2, the difference between the output of the plant controlled by the linearizing controller and the linear reference is purely down to rounding errors during simulation. It is in the range of 10^{-17} m when the sinusoid from Section 5.2.2 is applied as a reference trajectory. From a mathematical standpoint this is not surprising, because the idea of the controller is to renew the

plant dynamics completely by compensating the original behavior. If the plant model is known perfectly well, it can also be compensated perfectly well and, as the alternative name exact linearization suggests, the linearization ends up being flawless.

Interestingly, the progression of the control variable is almost identical to that provided by both the observer-based controller and our novel approach in the simulations of Section 5.2.2. It exhibits the same pronounced peak at zero velocity that characterizes the compensation of the LuGre model's behavior. Since the control variable progressions in the face of other LuGre models have noticeably different forms, we can conclude that the reaction of the controllers to the nonlinearity depends sensitively on the choice of parameters of the friction model. This is also a reminder that our novel controller design copes well with various parameterizations of the LuGre model.

Another aspect of this simulation is the internal dynamic behavior. We already discussed that formal stability analysis is problematic here. Instead, we have to keep an eye on the internal dynamics during simulation, in this case the bristle deformation z . Luckily, it turns out that the internal dynamics are well-behaved. The state z is bounded above by 10^{-3} m above and by -10^{-3} m below and performs an almost rectangular wave between those two values. This proves that the internal dynamics are not problematic in this scenario.

What happens when we change the true plant model, however, is what we are truly interested in, in this section. For the first time, we want to take a closer look at the nonlinearity of the LuGre model. Up until now, we have assumed that the functional form of the function $g(v)$ is known for the design of a nonlinear controller without flaws. This is perhaps the most unrealistic assumption in the first place. Expecting that the complex nonlinear effects caused by friction in a

mechanical system can be boiled down to a single function, is somewhat unrealistic. While the LuGre model has proven to approximate friction phenomena very well, we want to take a look at what happens if we modify the true friction effect in the simulation.

It turns out that a slight change in the functional form, for example

$$g_{true}(v) = F_c + (F_s - F_c) e^{-|\frac{v}{v_s}|} \text{ instead of } g(v) = F_c + (F_s - F_c) e^{-\left(\frac{v}{v_s}\right)^2}, \quad (5.43)$$

does not increase the error dramatically. The maximum deviation of the actual controlled variable and the reference system output is only 10^{-7} m. Surprisingly though, even seemingly more severe interference with this function, as in

$$g_{true}(v) = F_c + 1.5 (F_s - F_c) e^{-|\frac{v}{v_s}|}, \quad (5.44)$$

does not impair the behavior of the controller much more. Apparently, a differing functional form of $g(v)$ is easily compensated by the linearizing controller. This modification goes completely unnoticed, when our novel linear controller is applied. Since it is designed without further knowledge about the nonlinearity, its tracking error is already orders of magnitude larger, even if perfect parameter identification is assumed.

The great benefit of our design concept over model-based controllers shows again, once we modify the true plant parameters more drastically. Although the requirement of high dynamic demands from the control loop with the linearizing controller in Eq. 5.36 should yield great performance, even relatively modest changes in the plant model cause noticeable tracking errors. The short settling time should make the system react to the large control deviation due to the parameter uncertainty quicker and is still thrown off quickly by a mismatched plant model. Meanwhile, the control quality of our linear approach does not even show

any deterioration compared to the perfectly identified case.

Modifying only the LuGre friction parameters so they are now

$$\sigma_{0,true} = 2700 \frac{\text{N}}{\text{m}}, \sigma_{1,true} = 100 \frac{\text{kg}}{\text{s}}, \sigma_{2,true} = 4 \frac{\text{kg}}{\text{s}}, \quad (5.45)$$

and leaving all other parameters untouched, results in a peak tracking error of $7 \times 10^{-3} \text{ m}$, while the range of error using our novel controller is still smaller by a factor of almost 30, as we recollect from the simulations of the preceding section.

If the true plant parameters are chosen to be even farther off the values that are underlying the controller design, the tracking quality of the linearizing controller deteriorates quickly. Opting for the values

$$\begin{aligned} \sigma_{0,true} &= 2300 \frac{\text{N}}{\text{m}}, \sigma_{1,true} = 125 \frac{\text{kg}}{\text{s}}, \sigma_{2,true} = 6 \frac{\text{kg}}{\text{s}}, \\ F_{c,true} &= 3.2 \text{ N}, F_{s,true} = 6.3 \text{ N}, v_{s,true} = 0.0003 \frac{\text{m}}{\text{s}} \text{ and } m_{true} = 1 \text{ kg}, \end{aligned} \quad (5.46)$$

results in the output progression depicted in Figure 5.12. Note that neither the observer-based control loop nor our novel linear approach were severely affected by parameter mismatches of this category.

The error is now peaking at 0.022 m, more than 20 percent of the amplitude of the input sinusoid. Obviously, this controller is no strong competition for our linear controller. This shows even more that proper plant identification is indispensable for model-based control concepts, since they all rely more or less heavily on exact knowledge of the physical behavior of the system. Input-output linearization, which is discussed in this section, is an extreme example for such an approach. In essence, however, it sums up what problem all of them have in common. Without knowledge of the system at hand that is at least fairly detailed, they can be almost entirely useless.

This chapter has shown that what was one of the initial goals in the development of our linear controller, is achieved. The design requires only the knowledge

Figure 5.12: Output Progression with Linearizing Controller (blue) Subject to Modeling Uncertainties and Reference Output (orange)

of very few plant parameters, which are, usually, also simple to determine. On top of that, the control quality has proven to be preferable compared to existing nonlinear approaches regarding the tracking quality. Apart from the two designs discussed in Sections 5.1 and 5.2, the nested dual-loop approach was also successfully applied to a number of different plants taken from recent literature and compared to simulation results achieved with the respective control concepts.

For the sake of brevity and especially to keep the results presented in this thesis as meaningful as possible, these experiments are not discussed here. Nonetheless, it has to be said that in all cases our controller design successfully compensated friction and yielded a tracking quality that was at least competitive with the respective nonlinear control concepts presented for the various models. Particularly the LuGre friction model was studied exhaustively. The reason for that is not only the relevance of this dynamic model, but also the wide spectrum of different behaviors it can exhibit depending on its parameterization.

A total of thirteen different plant models were studied with unanimously good results. In all cases our controller achieved at least competitive results or even out-

performed the existing controller, while requiring similar control energy. Especially when facing parameter uncertainties or disturbances, our approach yielded excellent results.

Without going into detail, the control loops were also examined using various different reference signals like sine waves, ramps, steps and rectangular waves. Regardless of the input progression, the controller managed to compensate the occurring friction effect well and at most needed some tuning with respect to the settling time. The tracking accuracy is satisfactory across a wide spectrum of reference amplitudes and over a bandwidth of up to two orders of magnitude without a significant deterioration of quality. We now want to take our idea one step closer to a practical implementation of the controller. For the actual application to a physical problem, the algorithm has to be discretized.

CHAPTER 6

Discretization of the Control Algorithm

6.1 Design of the Digital Controller

Up until now, all simulations were conducted in continuous time, whereas in reality, the control algorithm would have to be implemented on a digital computer. Transferring the controller to discrete time introduces a number of new complications that have to be kept in mind to maintain the excellent performance of our controller.

The goal of this section is to construct a discrete-time control algorithm for a continuous-time plant. We do this based on the plant model presented in [3]. Its state space representation was already presented in Section 4.1.2. While the linear controller we are about to design is going to be digital, the plant used in the subsequent simulation is still continuous, to establish conditions that are close to reality. To make the discretization procedure as universal as possible, we try to refrain from using specific values as much as possible.

As the outset for the discretization of our controller architecture, we choose the complete linearized plant model, which contains the position and the reduced equation for the velocity,

$$\begin{aligned} \begin{bmatrix} \dot{p} \\ \dot{v} \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ 0 & -\frac{\sigma_2}{m} \end{bmatrix} \begin{bmatrix} p \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K_u}{m} \end{bmatrix} u \\ y = p &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} p \\ v \end{bmatrix}. \end{aligned} \tag{6.1}$$

The first step in determining the discrete controller is transferring the plant model into a discrete-time expression as well. This is achieved by following the procedure described in [34]. Based on the discrete plant the controller will be designed.

It is well known that the solution to the differential equation 6.1 is of the form

$$\mathbf{x}(t) = e^{\mathbf{A}(t-t_0)} \mathbf{x}(t_0) + \int_{t_0}^t e^{\mathbf{A}(t-\tau)} \mathbf{b} u(\tau) d\tau, \quad (6.2)$$

if the state space matrices are labeled in the conventional way and the system is subject to the initial conditions $\mathbf{x}(t_0)$ at time t_0 . The matrix $e^{\mathbf{A}(t-t_0)} \triangleq \Phi(t-t_0)$ is referred to as the transition matrix. Switching to the discrete-time domain is possible, if we assume a staircase progression of the control variable. This is sensible, because we are dealing with a digital control system, where the controller automatically outputs a piece-wise constant signal. The discrete model then describes the continuous behavior perfectly at the sampling instances, but not necessarily at the times between them.

To obtain a mathematical description of the discrete system, we simply calculate the solution of the continuous differential equations in a sampling interval of length T by substituting $kT = t_0$ and $(k+1)T = t$. As stated above, the control variable in this interval is constant and can be pulled out of the integral, therefore

$$\mathbf{x}((k+1)T) = e^{\mathbf{A}T} \mathbf{x}(kT) + \int_{kT}^{(k+1)T} e^{\mathbf{A}((k+1)T-\tau)} \mathbf{b} d\tau u(kT). \quad (6.3)$$

Another substitution $(k+1)T - \tau = \nu$, and the consequential modification to the integration operator $d\tau = -d\nu$, allow us to formulate the difference equation

$$\begin{aligned} \mathbf{x}(k+1) &= e^{\mathbf{A}T} \mathbf{x}(k) - \int_T^0 e^{\mathbf{A}\nu} \mathbf{b} d\nu u(k) \quad \text{or} \\ \mathbf{x}(k+1) &= \underbrace{e^{\mathbf{A}T}}_{\mathbf{A}_d} \mathbf{x}(k) + \underbrace{\int_0^T e^{\mathbf{A}\nu} \mathbf{b} d\nu}_{\mathbf{b}_d} u(k). \end{aligned} \quad (6.4)$$

In this expression, the discrete state space model $(\mathbf{A}_d, \mathbf{b}_d, \mathbf{c}_d)$ immediately follows from the evaluation of the transition matrix at the sampling time T and the solution of the integral. For the output matrix $\mathbf{c}_d = \mathbf{c}$ holds. In calculating the new matrices the matrix exponential has to be determined, which is defined in analogy with the

conventional exponential function

$$e^{\mathbf{A}} = \sum_{i=0}^{\infty} \frac{\mathbf{A}^i}{i!} = \mathbf{I} + \mathbf{A} + \frac{\mathbf{A}^2}{2!} + \frac{\mathbf{A}^3}{3!} + \dots \quad (6.5)$$

Given our simple second-order plant model, this is fairly straight-forward:

$$\begin{aligned} e^{\mathbf{A}T} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 0 & -\frac{\sigma_2}{m} \end{bmatrix} T + \begin{bmatrix} 0 & -\frac{\sigma_2}{m} \\ 0 & \left(-\frac{\sigma_2}{m}\right)^2 \end{bmatrix} \frac{T^2}{2} + \begin{bmatrix} 0 & \left(-\frac{\sigma_2}{m}\right)^2 \\ 0 & \left(-\frac{\sigma_2}{m}\right)^3 \end{bmatrix} \frac{T^3}{6} + \dots \\ &= \begin{bmatrix} 1 & \frac{m}{\sigma_2} \left(1 - e^{-\frac{\sigma_2}{m}T}\right) \\ 0 & e^{-\frac{\sigma_2}{m}T} \end{bmatrix} = \Phi(T) \end{aligned} \quad (6.6)$$

gives us the discrete system matrix $\mathbf{A}_d$ and upon plugging the result into the integral and solving it, we also receive the input matrix

$$\begin{aligned} \mathbf{b}_d &= \int_0^T \begin{bmatrix} \frac{K_u}{m} \frac{m}{\sigma_2} \left(1 - e^{-\frac{\sigma_2}{m}\nu}\right) \\ \frac{K_u}{m} e^{-\frac{\sigma_2}{m}\nu} \end{bmatrix} d\nu = \begin{bmatrix} \left[\frac{K_u}{\sigma_2} \nu + \frac{K_u}{\sigma_2} \frac{m}{\sigma_2} e^{-\frac{\sigma_2}{m}\nu} \right]_0^T \\ \left[-\frac{K_u}{\sigma_2} e^{-\frac{\sigma_2}{m}\nu} \right]_0^T \end{bmatrix} \\ &= \begin{bmatrix} \frac{K_u}{\sigma_2} T + \frac{K_u m}{\sigma_2^2} \left(e^{-\frac{\sigma_2}{m}T} - 1\right) \\ \frac{K_u}{\sigma_2} \left(1 - e^{-\frac{\sigma_2}{m}T}\right) \end{bmatrix}. \end{aligned} \quad (6.7)$$

Therefore, the difference equations describing the system in Eq. 6.1 at the sampling times are

$$\begin{bmatrix} p(k+1) \\ v(k+1) \end{bmatrix} = \begin{bmatrix} 1 & \frac{m}{\sigma_2} \left(1 - e^{-\frac{\sigma_2}{m}T}\right) \\ 0 & e^{-\frac{\sigma_2}{m}T} \end{bmatrix} \begin{bmatrix} p(k) \\ v(k) \end{bmatrix} + \begin{bmatrix} \frac{K_u}{\sigma_2} T + \frac{K_u m}{\sigma_2^2} \left(e^{-\frac{\sigma_2}{m}T} - 1\right) \\ \frac{K_u}{\sigma_2} \left(1 - e^{-\frac{\sigma_2}{m}T}\right) \end{bmatrix} u(k). \quad (6.8)$$

Note that determining the transition matrix can be significantly more complicated, especially for systems of higher order. MATLAB provides the built-in function `c2d`, which determines the discrete equivalent of a continuous-time state space model for a certain sampling time. It can also be used to obtain the exact same matrices derived above.

The following controller design directly follows the procedure that was already discussed thoroughly throughout this thesis. However, some obvious modifications

have to be made. First of all, the additional dynamics of the controller have to be transformed. Since both the inner and the outer loop have an integrator as the additional dynamics, the discrete equivalent of their continuous form is given by the difference equation

$$x_a(k+1) = \underbrace{1}_{A_a} x_a(k) + \underbrace{T}_{b_a} e(k), \quad (6.9)$$

wherein e denotes the control deviation of the respective loop. Just like the pole location of the controller dynamics, the closed-loop eigenvalue locations have to be specified differently. Between the continuous and the discrete realm, or the s -domain and the z -domain, complex values are subject to the relationship

$$e^{Ts} = z, \quad (6.10)$$

which maps the left s -half plane to the interior of the unit circle around the origin in the complex z -plane. We use Eq. 6.10 to choose pole locations for the design of our discrete controller that are equivalent to those in the continuous case.

Apart from these minor changes, the construction of the controller and observer are completely identical. They follow the procedure explained in Section 4.1.1. We begin by extracting the difference equation for the inner loop, which describes the velocity, from the discrete model. It is given by

$$v(k+1) = e^{-\frac{\sigma_2}{m}T} v(k) + \frac{K_u}{\sigma_2} \left(1 - e^{-\frac{\sigma_2}{m}T}\right) u(k). \quad (6.11)$$

From this equation and the additional dynamics of the controller we can immediately formulate the design model of the inner loop:

$$\mathbf{A}_{di} = \begin{bmatrix} e^{-\frac{\sigma_2}{m}T} & 0 \\ T & 1 \end{bmatrix}, \mathbf{b}_{di} = \begin{bmatrix} \frac{K_u}{\sigma_2} \left(1 - e^{-\frac{\sigma_2}{m}T}\right) \\ 0 \end{bmatrix}. \quad (6.12)$$

The eigenvalues of this matrix are placed in the locations specified by the set

$$\mathcal{Z}_i = \{0.3968 \pm 0.2006 j\} \quad (6.13)$$

using the `place` command. They result from the second order Bessel poles, scaled by the desired settling time $T_s = 0.2$ s, and the sampling time $T = 0.01$ s using the relationship in Eq. 6.10. We arrive at the closed inner loop

$$\begin{aligned} \mathbf{x}_i(k+1) &= \underbrace{\begin{bmatrix} e^{-\frac{\sigma_2}{m}T} - \frac{K_u}{\sigma_2} \left(1 - e^{-\frac{\sigma_2}{m}T}\right) k_{1,v} & \frac{K_u}{\sigma_2} \left(1 - e^{-\frac{\sigma_2}{m}T}\right) k_{2,v} \\ -T & 1 \end{bmatrix}}_{\mathbf{A}_{ci}} \mathbf{x}_i(k) + \underbrace{\begin{bmatrix} 0 \\ T \end{bmatrix}}_{\mathbf{b}_{ci}} u(k) \\ v(k) &= \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_{\mathbf{c}_{ci}} \mathbf{x}_i(k), \end{aligned} \quad (6.14)$$

with the state vector of the inner loop $\mathbf{x}_i^T = [v \ x_{a,i}]$. Complementing the closed inner loop by the discretized position dynamics gives us the discrete dynamics, to be controlled by the outer loop controller. With the position dynamics in Eq. 6.8, we arrive at

$$\begin{aligned} \mathbf{x}_1(k+1) &= \underbrace{\begin{bmatrix} 1 & \frac{m}{\sigma_2} \left(1 - e^{-\frac{\sigma_2}{m}T}\right) & 0 \\ 0 & e^{-\frac{\sigma_2}{m}T} - \frac{K_u}{\sigma_2} \left(1 - e^{-\frac{\sigma_2}{m}T}\right) k_{1,v} & \frac{K_u}{\sigma_2} \left(1 - e^{-\frac{\sigma_2}{m}T}\right) k_{2,v} \\ 0 & -T & 1 \end{bmatrix}}_{\mathbf{A}_1} \mathbf{x}_1(k) + \dots \\ &\quad \dots + \underbrace{\begin{bmatrix} \frac{K_u}{\sigma_2}T + \frac{K_um}{\sigma_2^2} \left(e^{-\frac{\sigma_2}{m}T} - 1\right) \\ 0 \\ T \end{bmatrix}}_{\mathbf{b}_1} u(k) \\ p(k) &= \underbrace{\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}}_{\mathbf{c}_1} \mathbf{x}_1(k), \end{aligned} \quad (6.15)$$

where the state vector is $\mathbf{x}_1^T = [p \ v \ x_{a,i}]$. Based on these equations and the additional dynamics, the design model of the outer loop follows directly:

$$\mathbf{A}_{do} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{0} \\ b_a \mathbf{c}_1 & A_a \end{bmatrix}, \mathbf{b}_{do} = \begin{bmatrix} \mathbf{b}_1 \\ 0 \end{bmatrix}. \quad (6.16)$$

Placing the eigenvalues of this model in the positions specified by the set

$$\mathcal{Z}_o = \{0.811 \pm 0.0953 j, 0.3968 \pm 0.2006 j\} \quad (6.17)$$

yields the feedback gains for the outer loop and finalizes the controller design. The discrete observer is constructed in the same way as its continuous counterpart, with the design based on the model of the closed inner loop $(\mathbf{A}_{ci}, \mathbf{b}_{ci}, \mathbf{c}_{ci})$. The observer poles end up in the locations

$$\mathcal{Z}_{obs} = \{(-0.0098 \pm 0.3016 j) \times 10^{-3}\}, \quad (6.18)$$

which corresponds to a shift by a factor of ten in the left complex s -plane compared to the inner loop poles.

Note that the settling time of the observer subsystem is considerably smaller than it was during the tests with the continuous system. It results from a systematic tuning procedure, which was performed to ensure the highest possible tracking accuracy.

The continuous control concept is successfully transferred to the discrete domain and therefore ready to be implemented on a digital computer. To test the suitability of the system for the control of a physical plant, we want to deploy it in a simulation with a continuous-time model. Before presenting the findings of said simulation, however, some important modifications to the previous setup have to be explained.

The most obvious change happens to the additional dynamics of the respective loop. Continuous integrators with their poles in the origin of the complex plane have to be replaced by discrete integrators with the transfer function

$$G_I(z) = \frac{T}{z - 1} \quad (6.19)$$

in the z domain. Another alteration that has to be kept in mind when controlling a continuous plant with a digital algorithm, is the addition of a sample and hold element. Because the discrete controller deals with sequences of numbers rather than continuous progressions, it has to be fed with both a sampled version of the

plant output, the controlled position, and the reference input. To achieve this, we append a sample and hold block to the output of the continuous plant and add another one of these blocks to the input. They have to be triggered, which is taken care of by a pulse generator with pulse length equal to the sampling time T .

A last modification pertains to determining the velocity signal. Up until this point, we have always assumed that a measurement of the velocity is available for control. In reality this assumption is relatively problematic, because measuring the velocity is generally difficult to realize in practice. Instead, in most implementations, the velocity has to be calculated from the position. There are various ways to go about this. SIMULINK provides a numerical differentiator, which approximates the necessary mathematical operation of forming the time derivative by using a difference quotient:

$$\tilde{v}(k) = \frac{p(k) - p(k-1)}{T} \quad \circ \text{---} \bullet \quad \tilde{v}(z) = \frac{z-1}{Tz} p(z) \triangleq G_d(z)p(z). \quad (6.20)$$

From a theoretical standpoint, using the numerical approximation introduces the evident problem of pretending to have knowledge about future events. Since the value $p(k+1)$ will never be known beforehand, the difference quotient can not determine the true value of the velocity from the position with absolute certainty. The element that would be required to do so, a true differentiator, is not causal. On top of that, the numerical derivative causes a drastic amplification of noise, which is a practical issue that has to be dealt with.

Another option we have, to come up with an approximation for the velocity signal, is a differentiating lowpass filter. It uses a second-order difference quotient to calculate the velocity, but in addition to the zeros this produces in the transfer function, introduces two fast poles:

$$G_f(z) = K_f \frac{z^2 - 1}{z^2 + a_1 z + a_0}. \quad (6.21)$$

While at low frequencies the differentiation is almost unaffected, the filter does roll off at higher frequencies, to mitigate the effect of noise on the control behavior. As good an idea the differentiating filter is, in practice it does not surpass the performance of the straight-forward derivative block. For the approximation of the derivative by $G_f(z)$ to reach the quality of $G_d(z)$, the cutoff frequency of the filter has to be chosen so high that the noise amplification acts almost in its entirety. As an approximate differentiator the lowpass filter is not a good choice in this context.

Since we have already successfully used an observer to estimate the additional dynamic state of the inner loop controller, applying the same concept to this problem seems to suggest itself. Similarly to the differentiating filter, however, the observer poles have to be chosen close to the origin of the complex z -plane for the quality to be competitive to the numerical derivative in Eq. 6.20. In doing so, the observer obtains a distinctly noise-amplifying behavior, which renders an observer useless for our purpose.

We therefore opt for the standard SIMULINK differentiator in our simulations. Equipped with these modifications, we turn our attention to the experimental investigations. The block diagram of the continuous plant, controlled by the digital controller, is shown in Figure 6.1.

6.2 Performance of the Discrete Controller

To see how well the control quality is maintained in the discrete implementation of the controller, we want to examine two very important aspects of practical application: the effect of sampling on the overall behavior and the influence of noise. Obviously, inserting a digital controller in a control loop with a continuous-time plant will alter the behavior compared to the exclusively continuous system. How much the tracking quality deteriorates and how short the sampling time has

Figure 6.1: Discrete Control System with Observer

to be chosen, are the issues that we want to tackle in this investigation.

On the other hand, a problem in any practical application is associated with noise of various sources. It can be caused by electronic components in the signal chain, by encoder jitter or it can be picked up from the environment. Either way, noise is a source of disturbance that is encountered in any system and that has not been considered up until this point. We want to take a closer look at the consequences of noise in the control circuit.

6.2.1 Impact of Sampling on the Control Behavior

A first elementary test is to whether sampling the reference signal and controlled variable drastically disturbs the good results we previously achieved with the continuous controller. To that end, we compare the discrete linear controller and observer designed above, in the loop with the continuous plant (see Figure 6.1), to a continuous reference system without friction. It is designed based on the same settling times. This is of course oriented on the experiments that were conducted with the continuous controller. As the reference position, we choose a sinusoid of amplitude 0.01 m and frequency 0.5 Hz.

From Figure 6.2, we can immediately tell that the actual position visibly deviates from the continuous reference progression. The largest difference is still reached around zero velocity, where the nonlinear LuGre friction effect has the biggest influence on the control behavior. This is in line with our experiences so far.

On top of that, however, the digital controller entails a staircase-like progression of the controlled variable. Instead of smoothly following the reference trajectory, the course of the position exhibits distinct steps. They are likely caused by the piecewise constant control signal, which forces the system to approach zero velocity in each sampling interval and therewith entails a noticeable impact of

Figure 6.2: Output Progression (blue) and Continuous Reference Output (orange) for System with Discrete Controller

nonlinear friction.

This tendency is intensified if the controller design is based on a shorter settling time, even though the LuGre friction effect shows less in that case. The choice of the settling time $T_s = 0.2\text{s}$ constitutes a trade-off between a small influence of nonlinear friction and the impact of sampling on the control behavior.

Unfortunately though, the significant steps in the progression of the position are unacceptable for the overall performance of the control circuit. To improve the tracking quality, the sampling time has to be decreased. By the choice of a sampling rate twice as high than the previous one, namely $T = 0.005\text{s}$, the output progression is smoothed out and we achieve the behavior plotted in Figure 6.3.

Just by doubling the sampling frequency, we essentially arrive at the good control quality of the continuous control circuit again. Note however, that the deviation around zero velocity still remains. Increasing the dynamic demands by decreasing the desired settling time improves the reaction of the controller to the friction effect, but at the same time evokes steps in the output progression again. We therefore settle with a trade-off again, which is not only dictated by the optimal

Figure 6.3: Output Progression (blue) and Continuous Reference Output (orange) for System with Discrete Controller and Shorter Sampling Interval

tracking quality, but also by the required control energy.

The control variable to achieve the progression in Figure 6.3 is displayed in the plot of Figure 6.4, while the required control variable for a settling time of $T_s = 0.1\text{ s}$ instead of 0.2 s is shown in Figure 6.5. We can see that the progression in the former plot is relatively smooth and reaches a maximum of around 0.25 V , while it peaks at more than 0.65 V in the latter. Furthermore, the plot in Figure 6.5 exhibits an extreme chatter, which an actuator might not be able to provide in reality, or which could entail mechanical wear in a real-life system. Concluding, we can say that a larger settling time combined with a moderate sampling rate is preferable in this case.

Another aspect we want to highlight again in this section, is the robustness and disturbance rejection in the face of parameter uncertainties and load changes, respectively. Rather than systematically testing a number of different parameter sets, we want to simulate faulty system identification based on our experiences from Chapter 5 and in addition to that disturb the system with load steps of the form encountered in said chapter. Since the only information about the plant that is

Figure 6.4: Control Variable Progression for Discrete System, $T_s = 0.2$ s

Figure 6.5: Control Variable Progression for Discrete System, $T_s = 0.1$ s

Figure 6.6: Output Progression (blue) and Continuous Reference Output (orange) for Disturbed and Uncertain System

used here is the system mass and the viscous friction coefficient, we limit ourselves to modifying those two parameters and opt for the severely different values

$$m_{true} = 4.5 \text{ kg} \quad \text{and} \quad \sigma_{2,true} = 50 \frac{\text{kg}}{\text{s}}. \quad (6.22)$$

Even though the plant parameters are assumed to be far from the true values and the disturbance of the load is significant in this simulation, the tracking quality is still surprisingly good. Figure 6.6 shows the progression of the controlled variable. The plot shows that the tracking quality is largely unaffected with the exception of the instances where load steps occur around zero velocity (e.g. at $t = 2.7\text{s}$). At these particular times, the peak error is increased by around 50 percent, also causing a rapid increase of the required control variable. All in all, however, the closed-loop behavior is satisfying.

This section has shown that the promising results of the first chapters translate almost seamlessly into the realm of digital control. Our hope that the control concept proposed in [1] is suitable for experiments in the presence of dynamic friction phenomena is therewith confirmed and the idea is fit for practical trials.

By examining the influence of noise, we want to eliminate another potential source of problems in a physical system.

6.2.2 Influence of Noise on the Tracking Quality

It is well known that noise is an obstacle in physical environments, which is impossible to avoid. Even though there are numerous ways of coping with noise from different sources, reaching from proper shielding of electronics to signal filtering, it is a disturbance which can never be fully avoided. In our case, noise is a particularly delicate topic. Since we are using a differentiator block to obtain the velocity, any form of noise will be amplified in an unduly manner. Therefore, it has to be made absolutely certain that noise will not deteriorate the control behavior to an extent where the tracking quality becomes useless.

We, again, assume that the system is free of other sources of uncertainty or disturbance and set up another simulation of our discrete linear controller. The only modification that is made to the experimental assembly is the addition of a noise source, which superimposes a randomly generated number onto the sampled position signal. Instead of white Gaussian noise a uniformly distributed random number is chosen, since it offers a more intuitive interpretation of its range. Tests with a Gaussian noise source resulted in the same tendencies and showed no superficial difference to the case considered here.

If the noise samples are uniformly distributed on the interval $[-10^{-4}, 10^4]$ and the controller is otherwise unaltered, the progression of the position plotted in Figure 6.7 follows. It shows a zoomed-in section of the simulation, to convey a better impression of the system behavior. The noise unquestionably shows in the trajectory of the controlled variable. It no longer smoothly tracks the reference progression, but deviates from the output of the linear reference model. This discrepancy, however, is very minor, if we consider the small amplitude of the

Figure 6.7: Output Progression (blue) and Continuous Reference Output (orange) for the System Subject to Noise

reference signal. It peaks at around five percent of the reference amplitude.

A bigger problem could be caused with respect to the control variable. It exhibits a high-frequency chatter, similar to that already observed in the plot of Figure 6.5. Without any knowledge about the actuator dynamics, it is difficult to make assumptions to how well a physical system would hold up in the face of noisy measurements, but it is a factor that has to be kept in mind. The problem here is not that an actuator might not be able to provide the required energy, but rather that the rapid changes of the control variable might be unfeasible or could cause unreasonable mechanical wear.

We want to neglect the control variable for the time being and take a closer look at the magnitude of the noise, to verify that our control architecture can in fact operate in the presence of it. If we consider noise of lower amplitude than that used in the simulation above, say uniformly distributed on $[-10^{-5}, 10^5]$, it has virtually no impact on the tracking quality. However, if we increase the noise level, the deterioration in accuracy and the consumption of control energy it entails, is so large that the controller would be rendered useless for most practical situations.

Figure 6.8: Output Progression (blue) and Continuous Reference Output (orange) for the System Subject to Noise of Higher Amplitude

This shall be visualized by the plot in Figure 6.8. If we allow noise samples that are uniformly distributed on $[-10^{-3}, 10^3]$, the controlled variable severely deviates from the continuous linear reference and the required control variable reaches values of up to ± 4 V instead of the previous ± 0.23 V.

At first glance, the noise resistance of the controller therefore seems really rather poor. Since noise samples equal to $\pm 10^{-3}$ m already cause a severe degradation in control quality, one could expect that the controller is not practical at all. If we demand high accuracy from our controller, however, we have to assume high accuracy in the hardware as well. A measuring error that reaches values of up to ten percent of the actual value, which would be the case here, is simply not acceptable in the first place.

Conversely, if we can base our considerations on an encoder providing a measurement within a tolerance of only a few percent, the tracking quality is acceptable and the controller design is applicable in practice. On top of that, if we respect the signal to noise ratio, the noise amplitude is no longer very unrealistic anyway. The output trajectory is a sinusoid of amplitude $A = 0.01$ m as a good approximation

and therefore has a mean square of

$$\text{E} [p_{ref}^2] = \frac{A^2}{2} = 5 \times 10^{-5} \text{ m}^2. \quad (6.23)$$

Zero-mean uniformly distributed noise has a mean square of

$$\text{E} [N^2] = \text{Var} [N] = \frac{1}{12} (2a)^2 = 3.33 \times 10^{-7} \text{ m}^2, \quad (6.24)$$

where a is the maximum value the random variable can take on. Plugging these values in the definition for the signal to noise ratio yields

$$\text{SNR} = \frac{\text{E} [p_{ref}^2]}{\text{E} [N^2]} = 150 \text{ or } \text{SNR}_{\text{dB}} = 21.8 \text{ dB}. \quad (6.25)$$

These numbers are by no means far fetched when it comes to hardware requirements for a high-accuracy system. We can therefore conclude that the system employing our novel linear controller design yields satisfying tracking quality, if reasonable demands to the noise amplitude are made. It has shown to work really well in simulation and defied not only parameter uncertainties, but also disturbance in the form of load variations and noise.

On top of that, a practical application is within a grasp, since the controller was successfully transferred to the discrete domain. This concludes our examination of the linear control concept and its aptitude for friction compensation.

CHAPTER 7

Summary and Conclusion

To reflect on the findings of the research, we want to go over the results again and point out the essential benefits of our approach. We set off with the goal of proving the applicability of the controller design in [1] to arbitrary mechanical systems subject to friction. Our basic condition was that the design is meant to be based on a simple linear approximation of the nonlinear plant. By matching the exceptional results reported in [1] in the context of other systems, we wanted to show that the idea is apt for practical implementation on a larger scale.

As a first step on the way of dealing with arbitrary friction models, we examined the reconditioning of the plant equations to receive a controllable design model. This was necessary to construct our control concept based on alternative system descriptions. Particularly for the case of the very important LuGre model, this step took some creative handling of the linearized state space representation, as can be seen from Section 3.3. Proven by the extraordinary results in the following simulations, we found that different model reduction techniques are suitable for the treatment of the plant equations. This insight constitutes the first major progress of the work at hand, since it makes the design applicable to a wide variety of systems that are studied in literature.

Throughout Chapter 4 the controller was successfully used for tracking in a number of models. Besides the most elementary single-mass systems, the design also allowed for friction compensation if motor dynamics are taken into account and if the system equations are considerably more complex. On top of that, the control architecture was modified to accommodate for a different plant structure in which no distinct velocity loop can be constructed. We have reinforced the hope

that the idea is indeed suitable for many system models and can even be adapted to systems of divergent structure. This proves the potential of the approach.

Chapter 5 proved not only that the linear controller design is competitive with existing nonlinear approaches, but also that our design procedure does not rely on an exact model identification. This was another elementary goal of this work. It was shown that the controller is more robust in the presence of parameter uncertainties and disturbances than nonlinear model-based alternatives. Systems and controllers from literature were used as a benchmark.

Finally, and perhaps most importantly, the control algorithm was transferred to the discrete domain, to prepare it for implementation on a digital computer. In this process, the promising results from the preceding chapters were preserved, which paves the way to an application with high demands on accuracy. Additionally, the influence of noise on the control quality was studied to determine whether this practical aspect could become an obstacle. Despite the differentiator in the control loop, noise of reasonable amplitude does not impair the tracking quality drastically and the controller operates with satisfying accuracy.

Thereby, the design was brought as close to a practical application as possible, leaving only an implementation to a physical system as a next step. We can conclude that using the linear controller with the observer for the inner loop has good prospects for friction compensation in mechanical systems. Especially if high accuracy is paramount, the linear approach seems to be promising.

CHAPTER 8

Future Work on the Topic

Despite the highly promising results of this work, there are still some unanswered questions and topics to cover in order to finalize an approbated control concept. The most obvious item on this list is the implementation to a physical system. Through the tests performed and documented in this thesis and the discretization of the control law, the way to such a practical application was paved. However, no simulation will ever show all problems that could occur during operation with a physical plant. Although the control architecture was successfully used in the context of one particular system, as shown in [1], tests with other models would be indispensable to verify the fitness of this controller.

The next potential issue with the cascaded controller is related to the control variable consumption. As mentioned in Section 6.2.2, the rapid chatter that has to be provided by the actuator might not be feasible or could be harmful for the hardware, if a mechanical actuator is used. While the amplitude of the control signal itself should not be a showstopper, rapid changes in its progression might prevent the control circuit from functioning properly. Taking the actuator dynamics into account would be a sensible step to prevent a deterioration of performance in a real-life scenario.

Note that, in our case, this problem has foreshadowed only in the presence of noise, so there might be a different way of avoiding it using proper filtering or alternative noise reduction techniques.

Another important supplement to the existing structure of the controller would be a prefilter, which allows the controller to reach the desired reference signal amplitudes. It was briefly discussed at the end of Chapter 2 that the controller

only compensates friction and can not achieve accurate tracking of an arbitrarily fast reference trajectory. This could be mended, however, by employing a linear dynamic prefilter.

On top of these facets of the controller, there are some levers in the design process which could have the potential of enhancing the quality even further. One is the location of the required closed-loop poles of the system. The choice of Bessel poles in [1] and throughout this thesis was made because they ensure optimal reproduction of the reference input. These eigenvalues have proven to preserve the shape of the signal. Other pole positions could provide an advantageous dynamic behavior and might be preferable in some applications. This is particularly interesting for the case of plants with position feedback (see Section 4.3), where the sensitivity towards the settling time was an unresolved problem. A different choice of eigenvalues could be the remedy.

Apart from that, one elementary building block of the control architecture was always left untouched. The additional dynamic component of the inner and outer control loop was just assumed to be an integrator while, in theory, any pole configuration of the additional dynamics is possible. Although the integrator by itself has the pleasant advantage of ensuring stationary accuracy in the face of disturbances and reference variable steps, other or further eigenvalues might in some cases improve the performance.

Lastly, the control architecture bears opportunities as well. After successfully handling friction in a nonlinear setting, there is hope that the controller might be able to compensate other forms of nonlinearities as well. We saw in Section 4.3 that backlash, modeled by a deadzone in the system dynamics, was mitigated upon proper tuning of the desired settling time, even on top of the already existing LuGre friction effect. This gives rise to hope that our controller might not even be

limited to the compensation of friction, but could also deal with other nonlinear phenomena in dynamic systems.

LIST OF REFERENCES

- [1] R. J. Vaccaro, "Friction Compensation without a Friction Model," in *2012 American Control Conference (ACC)*, 2012.
- [2] J. Na, Q. Chen, X. Ren, and Y. Guo, "Adaptive Prescribed Performance Motion Control of Servo Mechanisms with Friction Compensation," *IEEE Transactions on Industrial Electronics*, 2013.
- [3] D. Hoshino, N. Kamamichi, and J. Ishikawa, "Friction Compensation Using Time Variant Disturbance Observer Based on the LuGre Model," in *2012 12th IEEE International Workshop on Advanced Motion Control (AMC)*, 2012.
- [4] T. Dumitriu, "A Non-Linear Control Algorithm for a DC Servo Tracking System with Friction," in *EUROCON 2007 - The International Conference on "Computer as a Tool"*, 2007.
- [5] Y. Zhang and X. Ren, "Adaptive Backstepping Control of Dual-Motor Driving Servo Systems with Friction," in *2014 Sixth International Conference on Intelligent Human-Machine Systems and Cybernetics*, 2014.
- [6] L. Freidovich, A. Robertsson, A. Shiriaev, and R. Johansson, "LuGre-Model-Based Friction Compensation," *IEEE Transactions on Control Systems Technology*, 2009.
- [7] J. Ishikawa, S. Tei, D. Hoshino, M. Izutsu, and N. Kamamichi, "Friction Compensation Based on the LuGre Friction Model," in *Proceedings of SICE Annual Conference 2010*, 2010.
- [8] H. Olsson, K. J. Åström, C. C. De Wit, M. Gäfvert, and P. Lischinsky, "Friction Models and Friction Compensation," *European Journal of Control*, 1998.
- [9] R. Stribeck, *Wesentliche Eigenschaften der Gleit-und Rollenlager*. Springer, 1903.
- [10] E. Rabinowicz, "The Nature of the Static and Kinetic Coefficients of Friction," *Journal of Applied Physics*, 1951.
- [11] P. R. Dahl, "A Solid Friction Model," Aerospace Corp El Segundo CA, Tech. Rep., 1968.
- [12] P.-A. Bliman and M. Sorine, "A System-Theoretic Approach of Systems with Hysteresis. Application to Friction Modelling and Compensation," in *Proceedings of the 2nd European Control Conference*, 1993.

- [13] C. C. De Wit, H. Olsson, K. J. Astrom, and P. Lischinsky, "A New Model for Control of Systems with Friction," *IEEE Transactions on Automatic Control*, 1995.
- [14] J. Cheng and X. Wang, "Analysis and Compensation for Friction in Electro-Hydraulic Servo System," in *2011 International Conference on Consumer Electronics, Communications and Networks (CECNet)*, 2011.
- [15] M. Sun, Z. Wang, Y. Wang, and Z. Chen, "On Low-Velocity Compensation of Brushless DC Servo in the Absence of Friction Model," *IEEE Transactions on Industrial Electronics*, 2012.
- [16] H. K. Khalil and J. W. Grizzle, *Nonlinear Systems*. Prentice Hall Upper Saddle River, NJ, 2002.
- [17] Y. Tan, J. Chang, and H. Tan, "Adaptive Backstepping Control and Friction Compensation for AC Servo with Inertia and Load Uncertainties," *IEEE Transactions on Industrial Electronics*, 2003.
- [18] Y. Tan and I. Kanellakopoulos, "Adaptive Nonlinear Friction Compensation with Parametric Uncertainties," in *Proceedings of the 1999 American Control Conference (Cat. No. 99CH36251)*, 1999.
- [19] Q. Chen, L. Tao, Y. Nan, and X. Ren, "Adaptive Nonlinear Sliding Mode Control of Mechanical Servo System With LuGre Friction Compensation," *Journal of Dynamic Systems, Measurement, and Control*, 2016.
- [20] L. Lu, B. Yao, Q. Wang, and Z. Chen, "Adaptive Robust Control of Linear Motors with Dynamic Friction Compensation Using Modified LuGre Model," *Automatica*, 2009.
- [21] Y. H. Kim and F. L. Lewis, "Reinforcement Adaptive Learning Neural-Net-Based Friction Compensation Control for High Speed and Precision," *IEEE Transactions on Control Systems Technology*, 2000.
- [22] S. Huang and K. K. Tan, "Intelligent Friction Modeling and Compensation Using Neural Network Approximations," *IEEE Transactions on Industrial Electronics*, 2011.
- [23] Z. Wenjing, "Parameter Identification of LuGre Friction Model in Servo System Based on Improved Particle Swarm Optimization Algorithm," in *2007 Chinese Control Conference*, 2007.
- [24] R. H. Hensen, M. J. van de Molengraft, and M. Steinbuch, "Frequency Domain Identification of Dynamic Friction Model Parameters," *IEEE Transactions on Control Systems Technology*, 2002.

- [25] D.-P. Liu, “Parameter Identification for LuGre Friction Model Using Genetic Algorithms,” in *2006 International Conference on Machine Learning and Cybernetics*, 2006.
- [26] E. Hendricks, O. Jannerup, and P. H. Sørensen, *Linear Systems Control: Deterministic and Stochastic Methods*. Springer, 2008.
- [27] D. Owens, “Invariant Zeros of Multivariable Systems: A Geometric Analysis,” *International Journal of Control*, 1977.
- [28] T. Dumitriu, M. Culea, T. Munteanu, and E. Ceanga, “Friction Compensation for Accurate Positioning in DC Tracking System,” in *2006 3rd International Conference on Electrical and Electronics Engineering*, 2006.
- [29] H. Jian, Z. Xinhua, W. Guan, S. Zhiyi, and Y. Ting, “Adaptive Friction Compensation of Electromechanical Servo System Based on LuGre Model,” in *2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 2018.
- [30] J. Fukui, T. Yamamoto, G. Chen, and I. Takami, “Real-Time Identification and Compensation of Asymmetric Friction Using Unscented Kalman Filter,” in *2017 IEEE Conference on Control Technology and Applications (CCTA)*, 2017.
- [31] D. Bonvin and D. Mellichamp, “A Unified Derivation and Critical Review of Modal Approaches to Model Reduction,” *International Journal of Control*, 1982.
- [32] K. Khayati, P. Bigras, and L.-A. Dessaint, “LuGre Model-Based Friction Compensation and Positioning Control for a Pneumatic Actuator Using Multi-Objective Output-Feedback Control via LMI Optimization,” *Mechatronics*, 2009.
- [33] H. Zhao and C. Wang, “All-Coefficient Adaptive Control of Dual-Motor Driving Servo System,” in *MATEC Web of Conferences*, 2017.
- [34] R. Isermann, *Digital control Systems*. Springer Science & Business Media, 2013.
- [35] F. Svaricek, “Graphentheoretische Ermittlung der Anzahl von Strukturellen und Streng Strukturellen Invarianten Nullstellen/Graph-Theoretical Determination of the Number of Structural and Strictly Structural Invariant Zeros,” *at-Automatisierungstechnik*, 1986.

APPENDIX A

Appendix A: Structural Analysis of Invariant Zeros

Apart from computing the controllability matrix of a state space model, there is a plainer and more visual way of examining the structural controllability of a dynamic system. It is presented in [35] and comprises a graph-theoretical approach to calculating the number of invariant zeros of a system.

To perform the analysis of invariant zeros of a system and to find out where they are caused, the following graph is constructed to represent the Rosenbrock matrix. Each state variable of the system corresponds to a node in the graph. An oriented edge of this graph from a state node n_i to another one denoted by n_j exists, if in the state space representation the corresponding entry of the matrix $\mathbf{A}$, namely $[\mathbf{A}]_{ji}$, is different from zero.

For the sake of simplicity and sufficiently for our purpose, these edges are not weighted with specific factors. Note that, because the graph depicts the Rosenbrock matrix, each state node receives an edge leading back to itself, which is weighted by the variable λ . Both the input and output receive an extra node marked with I and O, respectively. Connecting edges from the input node to state nodes are put in place, if the respective entries in the input matrix $\mathbf{b}$ are nonzero and, analogously, from state nodes to the output node if the entries in the output matrix $\mathbf{c}$ are different from zero.

Finally, for the analysis, the input and output node are connected, completing what is referred to as the structural graph of a system. It is worth noting that in systems with multiple inputs and outputs every output is supposed to be connected to every input. For the system at hand and for systems of equal structure, for that matter, this procedure results in the graph in Figure A.1.

Figure A.1: Structural Graph of Linearized Single-Mass System with LuGre Friction Model

The actual analysis of the system involves determining the minimal 1-factor of the graph constructed in Figure A.1. This, particularly for MIMO systems and systems with large and densely populated state space matrices, can be extremely complicated. We call a spanning subgraph of a graph its 1-factor, if every single one of its vertices is the starting node and the ending node of an edge exactly once. A 1-factor is minimal if, and only if, its cycles with the input and output nodes have the minimum possible number of edges. A cycle refers to a closed sequence of nodes and edges in a graph.

Visually speaking, the search for 1-factors means finding the shortest path from an input node to an output node through a sequence of state nodes. Note that the 1-factor of a graph does by no means have to be unique.

For the present system structure, the determination of the single minimal 1-factor is very straight-forward. The edges of the particular subgraph of interest are marked blue in Figure A.1. To obtain the number of invariant zeros, all that is left to do is to determine the number of loops in the 1-factor, which means finding the number of state nodes in the subgraph that have an edge connecting them to themselves. Since this is clearly the case for the nodes representing z and x_a , the system has two invariant zeros. Due to the fact that they arise in loops weighted solely by the factor λ , we can conclude that they are located in the origin of the

complex plane.

They are given the name structural invariant zeros because they are a structural property of a dynamic system. This becomes evident when we recall the fact that their identification was carried out without taking the values of model parameters into account, solely considering the mathematical form of the system.

The great advantage of this procedure shows here: we can easily include additional dynamic properties of the system description, like for example motor dynamics, into the analysis. Not only is the algorithm easy to execute and adapt for our purpose, it also shows the same tendency of the linearization of the LuGre friction model to cause an unwanted invariant zero for more complicated system matrices alike.

Analysis of more detailed system models including motor dynamics reveal the same issue due to an invariant zero caused by the dynamic LuGre model.

APPENDIX B

Appendix B: MATLAB Code for the Controller Design

```
%System Model Based On:
%"Friction Compensation Using Time Variant Disturbance Observer...
% Based on the LuGre Model"
%(Hoshino, D., Kamamichi, N., Ishikawa, J. in IEEE International
5 % Workshop on Advanced Motion Control, 2012)
%
% Parameter Values

%Signal Parameters for Sinusoid Reference
10 Amplitude = 0.01;
frequency = 2*pi; %Angular Velocity

%True Parameters
Fcr = 3.5;
15 Fsr = 9;
sig0r = 50000;
sig1r = 600;
sig2r = 93.09;
mr = 3.27;
20 vsr = 0.075;
Ka = 5.3e-2;
Kt = 8.545e2;
Ku = Ka*Kt;

25 %Parameters for Design (e.g. from System Identification)
Fc = 3.5;
```



```

Fs = 9;
sig0 = 50000;
sig1 = 600;
30 sig2 = 93.09;
m = 3.27;
vs = 0.075;

%Poles and Settling Times
35 s1 = [-4.62];
s2 = [-4.053+2.34j;-4.053-2.34j];
s3 = [-5.0093;-3.9668+3.7845j;-3.9668-3.7845j];
Ts = 0.3;
Tsv = Ts/4;
40 Tso = Tsv/3;

%
%% Plant Model

%Linearized Inner Loop (velocity and bristle deformation)
45 Alin = [-(sig1+sig2)/m -sig0/m;1 0];
Blin = [1/m;0];
Clin = [1 0];
sysLin = ss(Alin,Blin,Clin,0);

%
50 %% Modal Model Reduction for Real or Strongly Dampened EV Pair

%Diagonalize System sysLin
[V,Eig] = eig(Alin);
Astar = Eig;
55 bstar = inv(V)*Blin;
cstar = Clin*V;

%Reconstruct Omitted States

```

```

    Qu = 2^2;
60 B11 = -(bstar(2)*Qu*bstar(2)')/(Eig(2) + Eig(2)');
    B21 = -(bstar(1)*Qu*bstar(2)')/(Eig(1) + Eig(2)');
    E = Astar(1)^(-1)*(B21 + (bstar(1) - B21*B11^(-1)*bstar(2))*...
        (bstar(2)'*B11^(-1)*bstar(2))*bstar(2)')*B11^(-1)*Astar(2);
    F = V(1,1) + V(1,2)*E;
65
    %Reduce Model
    A = F*Astar(2,2)*inv(F);
    b = F*bstar(2);
    c = 1;
70 sysRed = ss(A,b,c,0);
    %
    %% Model Reduction for Weakly Dampened EV Pair

    %Inner Plant (reduced)
75 A = -sig2/m;
    b = Ka*Kt/m;
    c = 1;
    %
    %% Controller Design
80
    %Additional Dynamics for Inner and Outer Loop
    Aa = 0;
    ba = 1;

85 %Design Model for Inner Loop
    Adi = [A 0;ba*c Aa];
    bdi = [b;0];

    %Pole Placement for Inner Loop
90 p = s2/Tsv;

```

```

K = place(Adi,bdi,p);
k1v = K(1);
k2v = K(2);

95 %Inner Loop
A1 = [0 1 0;0 A-b*k1v b*k2v;0 -ba Aa];
b1 = [0;0;ba];
c1 = [1 0 0; eye(3)];

100 %Design Model for Outer Loop
Ado = [A1 zeros(3,1);ba*c1(1,:) Aa];
bdo = [b1;0];

%Pole Placement for Outer Loop
105 p1 = s2/Tsv;
p2 = s2/Ts;
Ko = place(Ado,bdo,[p1,p2]);
k1 = Ko(1:3);
k2 = Ko(4);

110
%Reference Model
Ar = [(A1-b1*k1) b1*k2;-ba*c1(1,:) Aa];
br = [zeros(3,1);ba];
cr = [1 0 0 0];

115 %
%% Observer Design

%Observer Matrices
Ao = [A-b*k1v b*k2v;-ba Aa];
120 bo = [0;ba];
co = [1 0];

```

```
%Pole Placement for Observer  
pobs = s2/Tso;  
125 L = place(Ao',co',pobs);
```

BIBLIOGRAPHY

- Bliman, P.-A. and Sorine, M., “A System-Theoretic Approach of Systems with Hysteresis. Application to Friction Modelling and Compensation,” in *Proceedings of the 2nd European Control Conference*, 1993.
- Bonvin, D. and Mellichamp, D., “A Unified Derivation and Critical Review of Modal Approaches to Model Reduction,” *International Journal of Control*, 1982.
- Chen, Q., Tao, L., Nan, Y., and Ren, X., “Adaptive Nonlinear Sliding Mode Control of Mechanical Servo System With LuGre Friction Compensation,” *Journal of Dynamic Systems, Measurement, and Control*, 2016.
- Cheng, J. and Wang, X., “Analysis and Compensation for Friction in Electro-Hydraulic Servo System,” in *2011 International Conference on Consumer Electronics, Communications and Networks (CECNet)*, 2011.
- Dahl, P. R., “A Solid Friction Model,” Aerospace Corp El Segundo CA, Tech. Rep., 1968.
- De Wit, C. C., Olsson, H., Astrom, K. J., and Lischinsky, P., “A New Model for Control of Systems with Friction,” *IEEE Transactions on Automatic Control*, 1995.
- Dumitriu, T., Culea, M., Munteanu, T., and Ceanga, E., “Friction Compensation for Accurate Positioning in DC Tracking System,” in *2006 3rd International Conference on Electrical and Electronics Engineering*, 2006.
- Dumitriu, T., “A Non-Linear Control Algorithm for a DC Servo Tracking System with Friction,” in *EUROCON 2007 - The International Conference on "Computer as a Tool"*, 2007.
- Freidovich, L., Robertsson, A., Shiriaev, A., and Johansson, R., “LuGre-Model-Based Friction Compensation,” *IEEE Transactions on Control Systems Technology*, 2009.
- Fukui, J., Yamamoto, T., Chen, G., and Takami, I., “Real-Time Identification and Compensation of Asymmetric Friction Using Unscented Kalman Filter,” in *2017 IEEE Conference on Control Technology and Applications (CCTA)*, 2017.
- Hendricks, E., Jannerup, O., and Sørensen, P. H., *Linear Systems Control: Deterministic and Stochastic Methods*. Springer, 2008.

- Hensen, R. H., van de Molengraft, M. J., and Steinbuch, M., “Frequency Domain Identification of Dynamic Friction Model Parameters,” *IEEE Transactions on Control Systems Technology*, 2002.
- Hoshino, D., Kamamichi, N., and Ishikawa, J., “Friction Compensation Using Time Variant Disturbance Observer Based on the LuGre Model,” in *2012 12th IEEE International Workshop on Advanced Motion Control (AMC)*, 2012.
- Huang, S. and Tan, K. K., “Intelligent Friction Modeling and Compensation Using Neural Network Approximations,” *IEEE Transactions on Industrial Electronics*, 2011.
- Isermann, R., *Digital control Systems*. Springer Science & Business Media, 2013.
- Ishikawa, J., Tei, S., Hoshino, D., Izutsu, M., and Kamamichi, N., “Friction Compensation Based on the LuGre Friction Model,” in *Proceedings of SICE Annual Conference 2010*, 2010.
- Jian, H., Xinhua, Z., Guan, W., Zhiyi, S., and Ting, Y., “Adaptive Friction Compensation of Electromechanical Servo System Based on LuGre Model,” in *2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 2018.
- Khalil, H. K. and Grizzle, J. W., *Nonlinear Systems*. Prentice Hall Upper Saddle River, NJ, 2002.
- Khayati, K., Bigras, P., and Dessaint, L.-A., “LuGre Model-Based Friction Compensation and Positioning Control for a Pneumatic Actuator Using Multi-Objective Output-Feedback Control via LMI Optimization,” *Mechatronics*, 2009.
- Kim, Y. H. and Lewis, F. L., “Reinforcement Adaptive Learning Neural-Net-Based Friction Compensation Control for High Speed and Precision,” *IEEE Transactions on Control Systems Technology*, 2000.
- Liu, D.-P., “Parameter Identification for LuGre Friction Model Using Genetic Algorithms,” in *2006 International Conference on Machine Learning and Cybernetics*, 2006.
- Lu, L., Yao, B., Wang, Q., and Chen, Z., “Adaptive Robust Control of Linear Motors with Dynamic Friction Compensation Using Modified LuGre Model,” *Automatica*, 2009.
- Na, J., Chen, Q., Ren, X., and Guo, Y., “Adaptive Prescribed Performance Motion Control of Servo Mechanisms with Friction Compensation,” *IEEE Transactions on Industrial Electronics*, 2013.

- Olsson, H., Åström, K. J., De Wit, C. C., Gäfvert, M., and Lischinsky, P., “Friction Models and Friction Compensation,” *European Journal of Control*, 1998.
- Owens, D., “Invariant Zeros of Multivariable Systems: A Geometric Analysis,” *International Journal of Control*, 1977.
- Rabinowicz, E., “The Nature of the Static and Kinetic Coefficients of Friction,” *Journal of Applied Physics*, 1951.
- Stribeck, R., *Wesentliche Eigenschaften der Gleit-und Rollenlager*. Springer, 1903.
- Sun, M., Wang, Z., Wang, Y., and Chen, Z., “On Low-Velocity Compensation of Brushless DC Servo in the Absence of Friction Model,” *IEEE Transactions on Industrial Electronics*, 2012.
- Svaricek, F., “Graphentheoretische Ermittlung der Anzahl von Strukturellen und Streng Strukturellen Invarianten Nullstellen/Graph-Theoretical Determination of the Number of Structural and Strictly Structural Invariant Zeros,” *at-Automatisierungstechnik*, 1986.
- Tan, Y., Chang, J., and Tan, H., “Adaptive Backstepping Control and Friction Compensation for AC Servo with Inertia and Load Uncertainties,” *IEEE Transactions on Industrial Electronics*, 2003.
- Tan, Y. and Kanellakopoulos, I., “Adaptive Nonlinear Friction Compensation with Parametric Uncertainties,” in *Proceedings of the 1999 American Control Conference (Cat. No. 99CH36251)*, 1999.
- Vaccaro, R. J., “Friction Compensation without a Friction Model,” in *2012 American Control Conference (ACC)*, 2012.
- Wenjing, Z., “Parameter Identification of LuGre Friction Model in Servo System Based on Improved Particle Swarm Optimization Algorithm,” in *2007 Chinese Control Conference*, 2007.
- Zhang, Y. and Ren, X., “Adaptive Backstepping Control of Dual-Motor Driving Servo Systems with Friction,” in *2014 Sixth International Conference on Intelligent Human-Machine Systems and Cybernetics*, 2014.
- Zhao, H. and Wang, C., “All-Coefficient Adaptive Control of Dual-Motor Driving Servo System,” in *MATEC Web of Conferences*, 2017.