On Synthesis of Active Disturbance Rejection Controllers for Practical Output Tracking

Leunissen, Tom R.P.

Award date:
2021

EINDHOVEN UNIVERSITY OF TECHNOLOGY

On Synthesis of Active Disturbance Rejection Controllers for Practical Output Tracking

Master:	Mechanical Engineering
Research group:	Dynamics and Control

Student:	Tom Leunissen
Identity number:	0982190
DC number:	DC 2021.088

Supervisor:	Prof. Dr. Ir. N. van de Wouw
Coach:	Dr. Ir. C.G. Murguia Rendon
Additional committee member:	Michelle Chong, PhD (CST)
External committee member:	A. Luviano-Juárez, PhD (Instituto Politécnico Nacional, Mexico City, Mexico)

Eindhoven, October 14, 2021

Acknowledgments

Before you lies the thesis “On Synthesis of Active Disturbance Rejection Controllers for Practical Output Tracking ” which I wrote in partial fulfillment of the requirements for the degree of Master of Science in Mechanical Engineering at the Dynamics & Control group at the Eindhoven University of Technology.

This work marks the end of my life as a student. After leaving high school early, I started at a vocational school to now 11 years later defend my master’s thesis. After having the fascinating experience of seeing all levels the Dutch educational system has to offer, I feel it is time for me to start working in the industry.

First, I would like to thank my coach Carlos Murguia Rendon who made this project possible and introduced me to the very interesting subject of ADRC. Thank you for the teaching, the detailed feedback, the rapid responses, ..., your patience, and the very interesting and often length discussions we had during the past 9 months. Furthermore, I would like to thank my thesis supervisor Nathan van de Wouw for asking the right questions during our progress meetings and for his sharp feedback.

My time as a master’s student at the TU/e was overshadowed by the COVID-19 pandemic, working from home for more than 18 months was often difficult and lonely especially during the graduation period. I’m grateful for the guys I met during the pre-master, we stayed together as a group and made the best of it. I hope that we can continue our “social activities” after graduation.

Then there is my family. I want to express my gratitude to my girlfriend Desie for enduring and supporting me over the last 8 years. The final gratitude goes to my uncle Jo who over the years encouraged me to study and showed a deep interest in my progress during our frequent phone calls.

Abstract

Disturbances are present in all dynamical control systems. These disturbances can originate from outside the system (exogenous) or inside the system (endogenous). A method to mitigate the effect of these disturbances (in feedback linearizable systems) is Active Disturbance Rejection Control (ADRC). In this method, the effects of the endogenous (unmodeled dynamics) and exogenous disturbances (for example vibrations) on the input-output relation(s) to be controlled is lumped into one term, named the lumped disturbance. Subsequently, this disturbance is estimated in real-time using an extended state observer. The estimated lumped disturbance is subtracted from the input-output equation(s) which *practically* cancels it out. Since the lumped disturbance contains nonlinear and time-dependent terms the system is *practically* turned into a linear time-invariant (LTI) system. Moreover, the observer estimates all the states of the system, enabling the use of state feedback.

The observer and feedback controller need to be tuned to ensure desirable performance. In the literature, this tuning is predominantly performed empirically. This tuning method can be difficult and time-consuming for MIMO systems with many inputs and outputs. The main goal of this work is to develop a framework based on Linear Matrix Inequalities (LMIs) to tune the ADRC scheme in an automated manner (synthesis) that ensures optimal performance.

For three performance specifications (H_∞, H_2, and ISS), LMI-based conditions are developed to tune the observer and controller. A method to synthesize (H_∞ only) (sub)optimal observers and controllers for practical output tracking is developed. In simulations, it is verified that the developed method can be used for practical tuning. Furthermore, it is tested what the effects of adding or omitting model information to the synthesis process and observer is on the tracking performance. Moreover, the effect of adding more dynamic extensions to the observer is investigated. These tests are all done first under ideal conditions (no measurement noise and sampling) and then under more realistic conditions (with noise and sampling) to study the effects of unavoidable practical limitations on the tracking performance. Furthermore, the control signal is analyzed to determine how measurement noise enters the signal. Under the ideal conditions, it is tested whether it is possible to reconstruct an external disturbance signal while tracking, and which model information is needed to do this accurately. Next, the scheme is verified on a real electromechanical system named the Generalized Huygens Setup (GHS). It is tested how the tracking performance of the synthesized system compares to the empirical tuning methods. Furthermore, a comparison between the simulation and practical results is made.

A practical synthesis method for the ADRC scheme using the H_∞ specification is developed and demonstrated in simulation and on a real world system. It is shown that known model information can be incorporated into the synthesis framework, and that adding information is useful. The effect is that the observer convergence time decreases, as well as the rise time and the steady-state error. The effect of adding more dynamic extensions in the observer using the synthesis method is positive when full model information is used. When no model information is used the effect is negative. Additional weighting matrices are proposed to overcome this negative effect. Furthermore, it is demonstrated that the ADRC scheme is sensitive to measurement noise. The need for frequency-dependent weighting matrices in the tuning process to account for these noise and actuator dynamics is identified. The ability to reconstruct external disturbances while tracking a reference signal is demonstrated. To separate the exogenous from the endogenous disturbance all system parameters of the concerned input-output relation should be known. The measured tracking performance in the realistic simulations and the practical results are comparable up to a certain bandwidth, but then start to deviate, probably due to actuator dynamics and delays. Finally, it is shown that the synthesis method is able to outperform the empirical tuning method (on a real system) in the sense of tracking performance and the time it takes to tune the scheme.

Contents

List of Abbreviations

ADRC	Active Disturbance Rejection Control
CO	Chaotic Oscillator
GA	Genetic Algorithm
GHS	Generalized Huygens Setup
GPIO	Generalized Proportional Integral Observer
ISE	Integral Square Error
ISS	Input-to-State Stability
LHP	Left Half-Plane
LMI	Linear Matrix Inequality
LQR	Linear Quadratic Regulator
LTI	Linear Time Invariant
MIMO	Multi Input Multi Output
RMS	Root Mean Square
SISO	Single Input Single Output

Nomenclature

$\alpha e \beta$	$\alpha \times 10^{\beta}$
$:=$	Equal by definition
$\alpha^{(\beta)}$	β^{th} derivative of α
$\bar{\sigma}$	Largest singular value
γ_k	Feedback control gain
γ_l	Estimation error gain
$\|\|a\|\|$	Euclidean norm of vector a

$\|a\|_2$	L_2 signal norm of vector a
$\|G(s)\|_2$	H_2 norm of transfer function matrix $G(s)$
$\|G(s)\|_\infty$	H_∞ norm of transfer function matrix $G(s)$
$\mathbb{N}_+$	Set of positive natural numbers
$\mathbb{R}_+$	Set of positive real numbers
$\mathbb{R}_{\geq\alpha}$	Set of real numbers $\geq \alpha$
$\mathcal{KL}$	Class kappa-ell function
$\mathcal{K}$	Class kappa function
σ_i	i^{th} Singular value
$\sup(a)$	Supremum of a
col(a,...,b)	Column vector containing $a, \ldots, b$
$\text{tr}(A)$	Trace A
$A \prec 0$	A negative definite
$A \preceq 0$	A semi-negative definite
$A \succ 0$	A positive definite
$A \succeq 0$	A semi-positive definite
A^H	Hermitian transpose of A
I_a	Identity matrix with dimension a
$L_g h(z)$	Lie derivative of $h(z)$ with respect to g
m	Number of inputs/outputs
n	System dimension
r	Number of dynamic extensions
s	Laplace variable
$\text{diag}(a, \ldots, b)$	Diagonal matrix with $a, \ldots b$ on the main diagonal

Chapter 1

Introduction

1.1 Motivation

Unknown perturbations, unmodeled dynamics and sensor/actuator noise cause unavoidable problems in the operation of engineering systems. These disturbances affect the system dynamics in an unpredictable and disadvantageous manner if they are not accounted for in the system design. Disturbances originated from outside or inside the system are called exogenous and endogenous disturbances, respectively. Examples of exogenous (mechanical) disturbances are geological disturbances, acoustic disturbances, vibrations of nearby machinery, etc. Examples of endogenous disturbances are unmodeled and unknown dynamics in the system. Problems that might arise in mechanical systems due to disturbances include but are not limited to structural instabilities, unpredictable behavior, loss of comfort, undesired acoustic effects, and loss of performance.

Many methods to mitigate or reject these disturbances are known in the literature. Firstly, the frequency response of the system is often engineered (by control or mechanical design) so that unpredictable disturbances in a given frequency range do not negatively affect the system dynamics. However, frequency response methods are known to be only locally applicable for systems with complex nonlinear behavior. Another well-known method is the use of passive vibration absorbers, e.g., dampers and elastomers. However, when unknown disturbances enter the system and high attenuation is desired, the mentioned methods alone are often not sufficient. Therefore, an active approach that can handle unknown dynamics and disturbances/uncertainties is needed to increase performance. A distinction can be made between active disturbance decoupling and active disturbance control [1]. In active disturbance decoupling, the system performance outputs are decoupled from exogenous disturbances. On the other hand, active disturbance control mitigates the effects of disturbances by counteracting their effect using control and the system actuators.

This project focuses on a particular class of active disturbance control, namely, Active Disturbance Rejection Control (ADRC). The main idea behind ADRC is to treat external disturbances and unknown internal dynamics as a lumped vector of disturbances [2]. The effect of disturbances on the system dynamics is estimated in real-time (using an observer) and then canceled using a feedback control action. This cancellation forces the plant to behave as a simplified linear system (ultra) locally. Subsequently, this simplified plant is regulated using an output feedback controller. The key benefit of this methodology is that perturbed complex nonlinear systems with (partially) unknown dynamics can be controlled using a linear controller. The ADRC method has been extensively studied to address academic and industrial problems in motion control [3], power electronics [4], teleoperation [5], process control [6], and underactuated systems [7]. These results show that the ADRC methodology is not limited to disturbances that are from a mechanical origin, but the method is valid for all additive, bounded, and sufficiently smooth disturbances. A more elaborate description of ADRC is given in Section 1.2.

Even though the method has proven itself to work in many systems, the tuning of the required controller/observer is, in general, done empirically on a case-by-case basis. Furthermore, the observer and controller are designed separately assuming that a separation principle holds. However, given the complex nonlinear behavior of many mechanical systems, even though nonlinearities are approximately rejected by the scheme, a separation principle does not hold in general. Hence, methods and techniques that enforce boundedness of solutions of the closed loop dynamics and tuning tools that provide performance guarantees are needed. Although the empirical tuning leads to workable solutions, these solutions do not guarantee the best possible performance, and they heavily rely on the experience of the person that tunes the scheme. Although ADRC is inherently robust against parameter variations [8], a systematic method to tune the scheme can help to quantify and increase robustness. It is therefore highly desired for system designers to have a general tuning methodology that allows designing the scheme in an optimal and robust manner. Optimal tuning enables the identification of the performance limits of the scheme, and hence can lead to performance increases, and allows for a fair comparison between different ADRC implementations. Having a framework that enables optimal and robust tuning in an automated manner (synthesis) could make a significant difference in the ease of implementation and the performance of the scheme. Developing a general framework that enables optimal synthesis for ADRC schemes is the main goal of this thesis.

1.2 State of the art

1.2.1 Working mechanism of ADRC

The key component of ADRC is the use of an observer to estimate the unmeasured internal states and lumped disturbances. Typically, the observer is a linear Luenberger-type state observer that is extended with a sufficient number of auxiliary states. The specific observer used in this thesis is called Generalized Proportional Integral Observer (GPIO) [9]. The most simple version of the GPIO contains a highly simplified system model equivalent to a chain of pure integrators, using only the system dimension, an approximation of the terms accompanying the control input variables, and a state extension. The GPIO can be augmented with additional information of the plant when available. Furthermore, it can be extended with more than one auxiliary state. The first extension estimates the lumped disturbance, the second its first derivative and so on. This enables us to generate a model of the lumped disturbance terms in terms of a self-updating polynomial in time. In [10], the effect of adding more states to the GPIO is described. Adding more states leads to higher accuracy of the disturbance estimation but makes the estimation more sensitive to the effects of high-frequency measurement noise. In Figure 1.1, a block diagram of a typical ADRC scheme with a tracking controller is depicted. It can be seen that the estimated lumped disturbance term $\hat{\xi}_0$ is subtracted from the plant input, thereby compensating the disturbance, forcing the plant to behave as the dynamics used to model the GPIO. This means that the effects of the external disturbance and the internal disturbance are strongly attenuated. These disturbances are not fully rejected since there will always be a residual error between the real plant and the ultra-local model in the GPIO. Simultaneously, the (tracking) controller uses the observed states to stabilize the plant and track a reference signal.

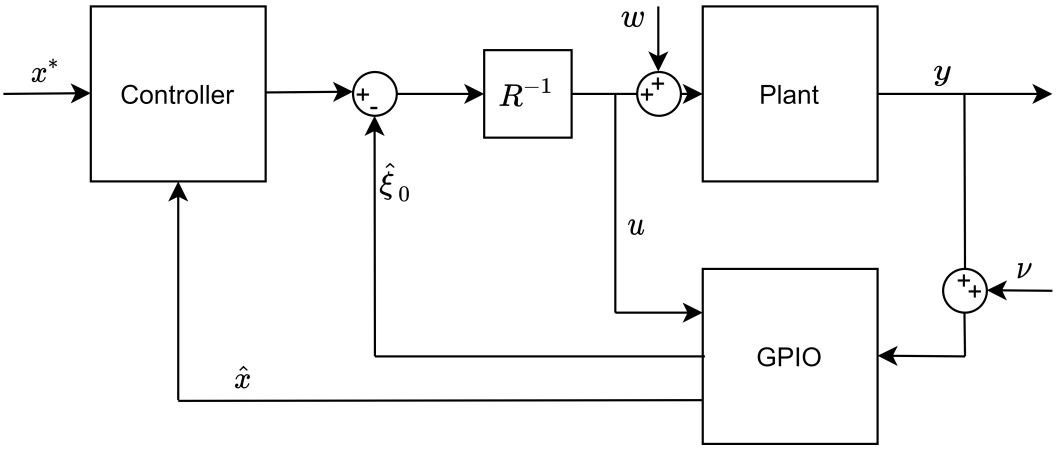


Figure 1.1: ADRC scheme with u, y, x^*, w, ν, $\hat{x}$, and $\hat{\xi}_0$ the control, output, reference, external disturbance, measurement noise, estimated internal states, and the estimated lumped disturbance, respectively. Furthermore, R is a known nonsingular matrix.

1.2.2 Tuning

The tuning of the GPIO and the controller can be done using linear design tools. By the separation principle (assuming it holds), it is known that the observer and controller can be designed separately without compromising stability. For stability of the observer/controller (error) dynamics, the roots of its closed loop characteristic polynomial are placed in the complex open left half-plane (Hurwitz stability) or in the open unit disc for discrete-time systems (Schur stability). However, the tradeoff between performance and robustness must be considered. Namely, selecting the gains too high might lead to amplification of measurement noise in closed loop, which can degrade performance or even destabilize the dynamics. On the other hand, using too low gains will make the observer too slow to follow the system dynamics and can therefore not accurately estimate the disturbance and internal states. Hence, it loses its ability to force the plant to behave as the dynamics in the observer. On the controller side, except for guaranteeing stability, the gains should be selected in such a way that the desired controller (tracking) performance is achieved. Furthermore, in practical situations, digital implementation and measurement noises should be taken into account. The necessary discretization and measurement noises can lead to stability and performance issues. Hence, the effects of discretization and measurement noises should be checked before implementation on the actual system. The work in [11] shows various discrete implementations of ADRC schemes.

In the ADRC literature, the selection of the observer/controller gains is done on a case-by-case basis. The first attempt to select these gains in a more general manner can be found in [12]; in this paper, the observer and controller are parametrized in terms of their bandwidth. This parametrization enables the tuning of the closed loop with just two free parameters, namely the observer bandwidth and the controller bandwidth. Even though only having two design parameters simplifies the tuning process and links it to the well-known bandwidth, the tuning itself still has to be done on an empirical case-by-case manner.

Many more sophisticated tuning methods are reported in the literature, of which a limited selection is mentioned here. The main point is that these results demonstrate the tuning of a specific system and not a general class of systems. A Genetic Algorithm (GA) parameter tuning method for ADRC controllers for aerial remote sensing application is given in [13]. The authors in [14] provide an ADRC build around evolutionary game theoretic optimization applied on a mobile robot. The most general tuning method is probably given in [15] which describes a Linear Quadratic Regulator (LQR) approach in an ADRC setting for the decoupling of MIMO systems. In the paper, two separate LQR problems are solved for the observer and controller. Resulting in two separate loops that can affect each other performance-wise in unpredictable manners. This method could be improved when a co-design of observer and controller is considered.

Albeit many specific tuning methods are demonstrated on specific problems, no references were found on how to select the gains in an optimal/robust manner for a general class of systems. Hence, it is worth seeking methods for selecting these gains in an optimal and robust manner using well-known methods in modern control theory, e.g., LQG, H_∞, or μ-synthesis.

1.2.3 Signal reconstruction

A different application of ADRC is found in [16], where, a signal (internal state) produced by a Chaotic Oscillator (CO) is added to another signal (hidden message). The message is now hidden (encoded) in the chaotic signal. When the original CO shares a certain output with a GPIO that contains the dynamics of the CO, the GPIO can reconstruct the internal states of the CO. Subtracting the estimated state from the encoded signal decodes the message. What this shows is that with a certain knowledge of the system controlled by an ADRC scheme, it is possible to reconstruct an external disturbance (in the example, the hidden message). Because in ADRC, the plant is forced to behave locally as a simplified plant which is known, and the control signal is known. It should be possible to reconstruct the effect of the external disturbance on the plant without knowing the dynamics of the plant. Moreover, when the dynamics of the plant are known, the original disturbance is computable. The ability to reconstruct the external disturbance or at least its effects on the simplified plant might be useful in, for example, for fault detection as described in, e.g. [17].

1.2.4 Open challenges

The following challenges related to the tuning of ADRC schemes are identified:

- A syntheses framework for the optimal tuning of ADRC schemes for a general class of mechanical systems is needed.

- Since there is no framework for optimal tuning, it is not known what the performance limits of the method are.

- The effect of applying more dynamic extensions to the GPIO is described in [10] using empirically tuned ADRC schemes. It is not clear whether these effects can be ascribed to tuning or to adding more auxiliary states.

- The ADRC scheme is known to be robust against parameter variation in the system/input matrix. It is however not known how robust the scheme is under optimal tuning.

The following challenges using ADRC schemes for disturbance reconstruction are identified:

- The authors of [18] describe the reconstruction of a exogenous disturbance added to an open loop system and reconstruct the disturbance using the known system dynamics. What is still open, is the possibility to (partially) reconstruct an exogenous disturbance in closed loop knowing the dynamics only partially.

- It is known that the lumped disturbance can be estimated using a GPIO. It is however not described in the literature if it is possible, and how to distinguish between the exogenous disturbance and the endogenous disturbance (system dynamics).

1.3 Research objectives

1.3.1 Optimal tuning

The optimal tuning of the ADRC scheme as stated in Section 1.2.4 is the main goal of this thesis. To reach this goal, the following steps are identified. We aim to develop and test a synthesis framework that enables the optimal and robust tuning of ADRC control schemes, for general mechanical system with (partially) unknown dynamics. The control goal is to minimize a certain norm/gain of the transfer function (matrix) between a given performance output, and the lumped disturbance while guaranteeing practical tracking and internal stability. The framework must be such that the gain matrices for the observer L and controller K, are determined explicitly. Furthermore, it should be possible to add more than one dynamic extension to the observer.

The following research questions will be addressed:

- How to optimally tune an ADRC scheme by explicitly determining gain matrices K and L by the means of synthesis?

- How to use the developed tools for practical output tracking?

- What are the effects of including model information in the synthesis process and the observer?

- What is the effect of the number of state extensions of the GPIO on the tracking performance of an optimally tuned ADRC scheme?

- How do measurement noises and sampling affect the optimally tuned ADRC scheme?

- How does the tracking performance of the optimally tuned ADRC scheme compare to empirically tuned schemes?

Disturbance reconstruction

To address the challenge regarding disturbance reconstruction posed in Section 1.2.4 the question that needs to be answered is:

- What information of the plant do we need for exogenous disturbance signal reconstruction?

-How to reconstruct an exogenous disturbance signal from the (regulated) output signal using the synthesized ADRC scheme?

1.4 Research approach

1.4.1 Optimal tuning

The tools selected for the optimal tuning of the ADRC scheme are the H_∞ and H_2 system norms and the notion of Input-to-State Stability (ISS). These performance criteria are used in the form of their corresponding linear matrix inequality (LMI) conditions. The idea is to develop an algorithm that takes the known information of the system to be controlled and determines the corresponding ultra-local model. Subsequently, the control/optimization problem is formulated and translated to an LMI and solved* for the controller/observer gains (explicitly) with the objective to minimize a certain parameter γ. The value of γ sets an upper bound to the the H_∞/H_2-norm or ISS-gain of the transfer function (matrix) between the lumped disturbance and the performance output (for ISS the performance output is the state vector). To use these LMIs to tune the ADRC scheme for practical tracking, (frequency-dependent) weighting matrices are needed. How to select these matrices and how to incorporate them into the relevant LMIs is explicitly out of scope of this work. For the H_∞ case, however, a tuning

*Using a semidefinite programming solver.

parameter is used to demonstrate a use case. For conciseness and to prevent a repetitive nature of the work, it is decided to only demonstrate the usage of the H_∞ case. However, similar ideas can be followed to use the other two performance specifications.

1.4.2 Disturbance reconstruction

Here, we investigate if it is possible to reconstruct the exogenous disturbance signal(s) from the measured signal(s) and the outputs of the GPIO while tracking a certain reference. Starting from the assumption the plant and the disturbance are fully known, the effects of omitting all information of the plant in the GPIO on the ability to reconstruct the disturbances are analyzed.

1.4.3 Simulations

Multiple simulations of ADRC schemes tuned using one of the derived synthesis methods (H_∞) are performed. The first goal of these simulations is to verify that the developed syntheses method works in simulation. The system chosen to verify the theoretical results is the Generalized Huygens Setup (GHS), a model of this system is described in Section 4.1. In Chapter 5, this setup will be used for practical experiments. The setup is a relatively simple electromechanical linear MIMO system. The availability of a model, and predictable behavior make it easy to troubleshoot possible issues. Since it is explicitly not the goal to control a specific (complicated) system, but to test the usability of the derived syntheses method, the GHS is a suitable choice.

A specific reference trajectory is designed and a method to quantify tracking performance is selected. Next, a model of the GHS is implemented in MATLAB Simulink. Several controllers are synthesized using no model information (only system order), and another set controllers with full model information. For both cases, multiple numbers of dynamic extensions are used $r = 1, 2, 3$ in the observer.

The first set of simulations is done under ideal circumstances, i.e., no measurement noise and sampling. The ability to track the reference trajectory using no model and full model information with $r = 1, 2, 3$ will be simulated and quantified.

Next, simulations are performed under more realistic conditions (noise and sampling are included now). Note that other practical effects e.g., actuator/sensor dynamics and delays, are omitted. The same experiment as in the former paragraph is repeated under these more difficult conditions. Finally, the control signal will be analyzed (qualitatively) to determine if the generated control signals are realistic/safe to implement on the real setup.

1.4.4 Experiments

The main goal of the experimental phase is to test if the developed H_∞ synthesis method works on a real world system. Furthermore, it is tested how the synthesis method performs (tracking) compared to the well-known empirical tuning methods. Moreover, it is tested how well the simulations agree with the practical experiment.

First, the GHS is prepared for control purposes. Sensor outputs and actuator inputs are translated from Volts into meters and Newtons, respectively. Subsequently, an empirically tuned PID or ADRC controller is used to test the the second-order SISO systems that the system consists of. These tests are to determine whether further input/output filtering is necessary. Furthermore, safety features are (input saturation) implemented to ensure the setup does not get damaged.

The experiment consists of the implementation of empirically tuned ADRC schemes that are often used in the literature. For the empirically tuned ADRC scheme (without model parameters), the tracking performance is quantified for the same trajectory as used in the simulations. The experiment is repeated for different number of dynamic extensions in the observer ($r = 1, 2, 3$). Next, the experiment is repeated, now with the ADRC scheme tuned using the H_∞ synthesized controller, also for $r = 1, 2, 3$. Finally, the results of the (realistic) simulations are compared to the practical findings.

1.5 Report outline

This thesis is organized as follows. In Chapter 2, the theoretical framework is given starting with a mathematical description of the used ADRC framework. This description is followed by an illustrative example to get acquainted with the notation and the ADRC scheme. Next, the used metrics for performance and their LMI representations are introduced.

Chapter 3 contains the theoretical results. Starting with a closed loop representation for ADRC that is suitable for the intended synthesis. In the next section, the need for a two step synthesis approached is explained. In the following part, LMIs are derived that can be used to tune the observer and controller using several performance criteria.

Chapter 4 gives the results of the performed simulations. In the first part, a practical method to use two of the derived LMIs to tune the ADRC scheme is explained. In the next part, a description of the model used for the simulations is given, followed by the design of the experiment. Subsequently, the results of the simulations are described.

In Chapter 5, the experimental results are given. The chapter starts with the practical difficulties encountered. Subsequently, the results for the experiments using empirically and synthesis based tuning methods are presented.

In Chapter 6, the final conclusions, discussion and a summary of the possible future research directions are presented.

Chapter 2

ADRC and Performance

In this chapter, the mathematical machinery and notation used in this thesis are introduced. In Section 2.1.1, ADRC and its working mechanism is explained. Next, an illustrative example is given in Section 2.1.2. Finally, the synthesis performance indicators (H_∞, H_2 and ISS) and their semidefinite programming formulation are treated in Section 2.2.

2.1 ADRC

2.1.1 ADRC formulation

Consider the system

$$\begin{aligned} \dot{z} &= f(z) + g(z)(u+w), \\ y &= h(z), \end{aligned} \tag{2.1}$$

with state $z \in \mathbb{R}^n$, input $u \in \mathbb{R}^m$, output $y \in \mathbb{R}^m$, external actuator disturbance $w \in \mathbb{R}^m$, and sufficiently smooth vector fields $f : \mathbb{R}^n \to \mathbb{R}^n$, $g : \mathbb{R}^m \to \mathbb{R}^n$, and $h : \mathbb{R}^n \to \mathbb{R}^m$, where $u = \text{col}(u_1, \ldots, u_m)$, $w = \text{col}(w_1, \ldots, w_m)$, $y = \text{col}(y_1, \ldots, y_m)$, $g(z) = (g_1(z), \ldots, g_m(z))$, and $h(z) = \text{col}(h_1(z), \ldots, h_m(z))$. Each y_i has a well-defined relative degree k_i, $i = 1, \ldots, m$. Moreover, $\sum_{i=1}^{m} k_i = n$; hence, system (2.1) is completely linearizable by state feedback [19]. It is assumed that each element in the vector of external actuator disturbances, w, is uniformly bounded [20]. Using the notion of *Lie derivatives* [21], we can write $\dot{y} = \frac{dy}{dt} = \frac{\partial h(z)}{\partial z}\dot{z} = \frac{\partial h(z)}{\partial z} + \frac{\partial h(z)}{\partial z}(u+w) := L_f h(z) + L_g h(z)(u+w)$, and $\ddot{y} = \frac{\partial (L_f h(z))}{\partial z}\dot{z} = L_f(L_f h(z)) + L_g(L_f h(z))(u+w) := L_f^2 h(z) + L_g L_f h(z)(u+w)$. Hence, in general, we can write $y^{(k)} = L_f^k h(z) + L_g L_f^{k-1} h(z)(u+w)$. It follows that system (2.1) can be written in terms of inputs and outputs as follows:

$$\underbrace{\begin{bmatrix} y_1^{(k_1)} \\ \vdots \\ y_m^{(k_m)} \end{bmatrix}}_{\bar{y}} = \begin{bmatrix} L_f^{k_1} h_1(z) \\ \vdots \\ L_f^{k_m} h_m(z) \end{bmatrix} + \underbrace{\begin{bmatrix} L_{g_1} L_f^{k_1-1} h_1(z) & \cdots & L_{g_m} L_f^{k_1-1} h_1(z) \\ \vdots & \ddots & \vdots \\ L_{g_1} L_f^{k_m-1} h_m(z) & \cdots & L_{g_m} L_f^{k_m-1} h_m(z) \end{bmatrix}}_{E(z)} \left[\begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix} + \begin{bmatrix} w_1 \\ \vdots \\ w_m \end{bmatrix} \right], \tag{2.2}$$

with $E(z)$ being nonsingular and containing no zero values on its main diagonal. Each input-output relation in (2.2), $y_j^{(k_j)}, j = 1, \ldots, m$, is then given by

$$y_j^{(k_j)} = L_f^{k_j} h_j(z) + L_{g_j} L_f^{k_j-1} h_j(z) u_j + \sum_{i=1, i\neq j}^{m} L_{g_i} L_f^{k_j-1} h_i(z) u_i + \sum_{i=1}^{m} L_{g_i} L_f^{k_i-1} h_i(z) w_i, \tag{2.3}$$

which can be compactly written as

$$y_j^{(k_j)} = L_{g_j} L_f^{k_j-1} h_j(z) u_j + \tilde{\xi}_j, \tag{2.4}$$

with $\tilde{\xi}_j := L_f^{k_1} h_j(z) + \sum_{i=1, i\neq j}^{m} L_{g_i} L_f^{k_j-1} h_i(z) u_i + \sum_{i=1}^{m} L_{g_i} L_f^{k_i-1} h_i(z) w_i$.

Hence, (2.2) can be expressed as

$$\bar{y} := \tilde{R}(z)u + \tilde{\xi} = (R + \bar{R}(z))u + \tilde{\xi} = Ru + \xi, \tag{2.5}$$

with $\xi := \tilde{\xi} + \bar{R}(z)u$, $\tilde{R}(z) := \operatorname{diag}\left(L_{g_1} L_f^{k_1-1} h_1(z), \ldots, L_{g_m} L_f^{k_m-1} h_m(z)\right)$, and R a nonsingular matrix denoting an educated guess of $\tilde{R}(z)$. Vector $\xi = \operatorname{col}(\xi_1, \ldots, \xi_m) \in \mathbb{R}^m$ is referred to as the system lumped disturbance. Note that all coupling terms, dynamics, and external disturbances are now contained in ξ.

We choose new coordinates as follows:

$$x := \operatorname{col}\left(\underbrace{h_1(z), \ldots, h_m(z)}_{x_1}, \underbrace{L_f h_1(z), \ldots, L_f h_m(z)}_{x_2}, \ldots, \underbrace{L_f^{k_1-1} h_1(z), \ldots, L_f^{k_m-1} h_m(z)}_{x_n} \right). \tag{2.6}$$

In the new coordinates, system (2.1) can be written as

$$\begin{aligned} \dot{x} &= Ax + B_u u + B_\xi \xi, \\ y &= Cx, \end{aligned} \tag{2.7}$$

for some known matrices $A \in \mathbb{R}^{n\times n}$, $B_u \in \mathbb{R}^{n\times m}$, $B_\xi \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{m\times n}$, and

$$B_u = \begin{bmatrix} 0 \\ R \end{bmatrix}, \quad B_\xi = \begin{bmatrix} 0 \\ I_m \end{bmatrix}, \quad C = \begin{bmatrix} I_m & 0 & \ldots & 0 \end{bmatrix}. \tag{2.8}$$

Matrix $R \in \mathbb{R}^{m\times m}$ is nonsingular by construction.

Remark 1 *Note that the choice of coordinates* (2.6) *together with* (2.8) *make ξ a matching disturbance that can be canceled by the controller when known.*

Using this new notation, we introduce the following class of (practical) linearizing and tracking controllers:

$$u = -R^{-1}(K(\hat{x} - x^*) + \hat{\xi}), \tag{2.9}$$

where $\hat{x}$ and $\hat{\xi}$ denote the estimates of states and lumped disturbance, respectively, and x^* is the desired reference. Matrix $K \in \mathbb{R}^{m\times n}$ is the controller gain matrix.

Substitution of control (2.9) in system (2.7) yields

$$\dot{x} = Ax + B_\xi(\xi - \hat{\xi} - K(\hat{x} - x^*)). \tag{2.10}$$

Equation (2.10) indicates that the system is "practically" linearized when the estimation error of the lumped disturbance $e_\xi := \xi - \hat{\xi}$ is sufficiently small. Furthermore, the closed loop eigenvalues of the remaining linear system can be placed arbitrarily for small enough state estimations errors $e_x := x - \hat{x}$.

So the question is how to estimate $\hat{\xi}$ and $\hat{x}$? The first step in the ADRC formulation, is to extend the plant state x with r new states (per input-output relation) given by the disturbance and its first $(r-1)$ derivatives, i.e., $\eta := \text{col}(x, \xi, \dot{\xi} \dots \xi^{(r-1)})$. We introduce new notation to write these time derivatives in state-space form: $\eta := \text{col}(x, \xi_0, \xi_1 \dots \xi_{r-1})$. Hence, the extended system is given by

$$\begin{aligned} \dot{\eta} &= A_e\eta + B_{ue}u + B_{\xi e}\xi^{(r)}, \\ y_\eta &= C_e\eta, \end{aligned} \tag{2.11}$$

with $A_e \in \mathbb{R}^{(n+rm)\times(n+rm)}$, $B_{ue} \in \mathbb{R}^{(n+rm)\times m}$, $B_{\xi e} \in \mathbb{R}^{(n+rm)\times m}$, and $C_e \in \mathbb{R}^{m\times(n+rm)}$ given by

$$A_e = \begin{bmatrix} A & [B_\xi\ 0] \\ 0 & \Phi \end{bmatrix}, \quad B_{ue} = \begin{bmatrix} B_u \\ 0 \end{bmatrix}, \quad B_{\xi e} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ I \end{bmatrix}, \quad C_e = \begin{bmatrix} I & 0 & \dots & 0 \end{bmatrix}, \quad \Phi = \begin{bmatrix} 0 & I & 0 & \dots & 0 \\ 0 & 0 & I & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & I \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}, \tag{2.12}$$

and $\Phi \in \mathbb{R}^{rm\times rm}$.

A Luenberger observer is used to estimate η (estimating thus ξ and x simultaneously). In the observer, the effects of $\xi^{(r)}$ (and higher-order derivatives) on the estimation is assumed to be small (≈ 0). This implies that we are effectively modeling the disturbance as a Taylor polynomial in time of order $r-1$. The observer is defined as

$$\begin{aligned} \dot{\hat{\eta}} &= A_e\hat{\eta} + B_{ue}u + L(y - C_e\hat{\eta}), \\ \hat{y}_\eta &= C_e\hat{\eta}, \end{aligned} \tag{2.13}$$

with observer gain matrix $L \in \mathbb{R}^{(n+rm)\times m}$ to be designed to maximize the estimation performance. The estimation error is given by $e_\eta = \eta - \hat{\eta}$ and thus $\dot{e}_\eta = \dot{\eta} - \dot{\hat{\eta}}$. Note that $\dot{e}_\eta$ is perturbed by $\xi^{(r)}$ which is assumed to be bounded. The effect of $\xi^{(r)}$ can be decreased by selecting L and K properly.

2.1.2 Illustrative example

In this section, the control of a nonlinear MIMO system with external disturbances acting on it is demonstrated. The aim of this section is to make the reader familiar with the notation and to demonstrate the possibilities of the ADRC scheme.

Consider the system depicted in Figure 2.1

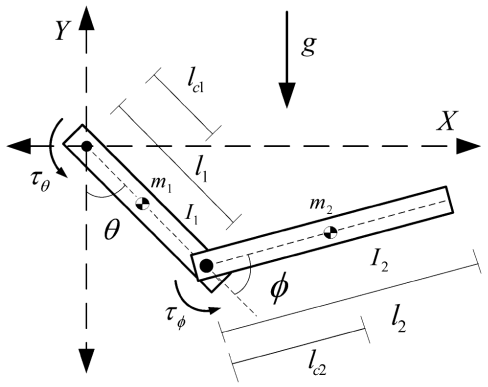


Figure 2.1: Two-DOF robotic manipulator. Adapted from [2, p. 88].

governed by the following equations of motion:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau + w - D\dot{q}, \tag{2.14}$$

with

$$\begin{aligned}
M(q) &:= \begin{bmatrix} m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c2}^2 + 2m_2 l_1 l_{c2}\cos(\phi) + I_1 + I_2 & m_2 l_{c2}^2 + m_2 l_1 l_{c2}\cos(\phi) + I_2 \\ m_2 l_{c2}^2 + m_2 l_1 l_{c2}\cos(\phi) + I_2 & m_2 l_{c2}^2 + I_2 \end{bmatrix}, \\
D &:= \begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix}, \\
C(q,\dot{q}) &:= \begin{bmatrix} -2m_2 l_1 l_{c2}\sin(\phi)\dot{\phi} & -m_2 l_1 l_{c2}\sin(\phi)\dot{\phi} \\ m_2 l_1 l_{c2}\sin(\phi)\dot{\theta} & 0 \end{bmatrix}, \\
G(q) &:= \begin{bmatrix} (m_1 l_{c1} + m_2 l_1)\, g\sin(\theta) + m_2 l_{c2} g\sin(\theta+\phi) \\ m_2 l_{c2} g\sin(\theta+\phi) \end{bmatrix},
\end{aligned} \tag{2.15}$$

$q := [\theta\ \phi]^\top$, $\tau := [\tau_\theta\ \tau_\phi]^\top$, and $w := [w_\theta\ w_\phi]^\top$ are the control torques and external disturbance torques, respectively.

It is well known that system (2.14) is state feedback linearizable with the output q; therefore, ADRC can be applied without additional assumptions on remaining dynamics.

Following the machinery introduced above, the system can be written as two decoupled second-order input-output relations

$$\ddot{q} = \tau + \xi_0 = \begin{bmatrix} \ddot{\theta} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} \tau_\theta \\ \tau_\phi \end{bmatrix} + \begin{bmatrix} \xi_\theta \\ \xi_\phi \end{bmatrix}. \tag{2.16}$$

Equation (2.16) implies that $M(q)$ is unknown and R is set equal to the identity matrix I. Furthermore, $\xi_0 := M^{-1}(q)\left(-C(q,\dot{q}) - G(q) + \tau(I - M(q)) + w - D\dot{q}\right)$.

The system is next written in state-space representation, with $\begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}^\top := \begin{bmatrix} \theta & \phi & \dot{\theta} & \dot{\phi} \end{bmatrix}^\top$,

$$\dot{x} = \underbrace{\begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}}_{A} x + \underbrace{\begin{bmatrix} 0 \\ I \end{bmatrix}}_{B_u} \tau + \underbrace{\begin{bmatrix} 0 \\ I \end{bmatrix}}_{B_\xi} \xi, \tag{2.17}$$

and extended with 2 states per input-output relation (i.e., $r = 2$) with $\begin{bmatrix} \eta_1 & \eta_2 & \eta_3 & \eta_4 & \eta_5 & \eta_6 & \eta_7 & \eta_8 \end{bmatrix}^\top := \begin{bmatrix} \theta & \phi & \dot{\theta} & \dot{\phi} & \xi_\theta & \xi_\phi & \dot{\xi}_\theta & \dot{\xi}_\phi \end{bmatrix}^\top$. The extended dynamics is then given by

$$\begin{aligned}
\dot{\eta} &= \underbrace{\begin{bmatrix} 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{A_e} \eta + \underbrace{\begin{bmatrix} 0 \\ I \\ 0 \\ 0 \end{bmatrix}}_{B_{ue}} \tau + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ I \end{bmatrix}}_{B_{\xi e}} \ddot{\xi}_0, \\
y_\eta &= \underbrace{\begin{bmatrix} I & 0 & 0 & 0 \end{bmatrix}}_{C_e} \eta.
\end{aligned} \tag{2.18}$$

Then, the observer in (2.13) takes the form:

$$\begin{aligned}\dot{\hat{\eta}} &= \underbrace{\begin{bmatrix} -l_1 & I & 0 & 0 \\ -l_2 & 0 & I & 0 \\ -l_3 & 0 & 0 & I \\ -l_4 & 0 & 0 & 0 \end{bmatrix}}_{\mathcal{A}_l} \hat{\eta} + \underbrace{\begin{bmatrix} l_1 \\ l_2 \\ l_3 \\ l_4 \end{bmatrix}}_{L} C_e \eta + \begin{bmatrix} 0 \\ I \\ 0 \\ 0 \end{bmatrix} \tau, \\ \hat{y}_\eta &= C_e \hat{\eta},\end{aligned} \tag{2.19}$$

with matrix L to be selected to minimize the effect of ξ_0 on the closed loop dynamics.

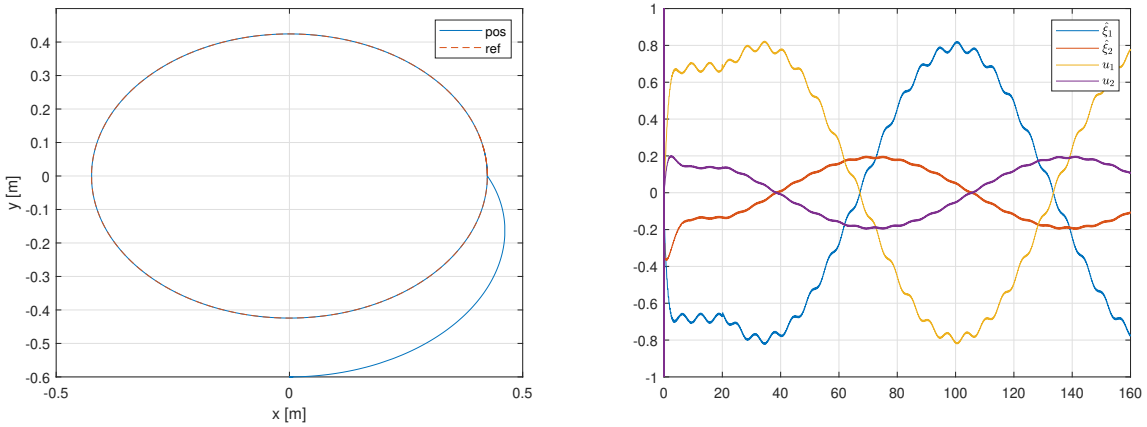
The tracking controller (2.9) is then given by,

$$\tau = -K(\hat{x} - x^*) - \hat{\xi}_0, \tag{2.20}$$

with matrix K to be used to minimize the effect of ξ_0 on the closed loop dynamics.

The reference trajectory is designed as follows. Initial conditions $\theta(0) = \phi(0) = 0$. Then, move to a reference point to start circular motion (upswing): $t \leq 20,\ \theta = \frac{\pi}{4}, \phi = \frac{\pi}{2}$. Next, keep the second link at $\phi = \frac{\pi}{2}$ and sweep out a full rotation with θ. This is denoted as: $t > 20,\ \theta = \frac{\pi}{4} + 0.0075 \cdot 2\pi(t-20), \phi = \frac{\pi}{2}$. We take the disturbance vector $w = \begin{bmatrix} 5\sin t + \frac{\pi}{4} & 3\sin 3t \end{bmatrix}^\top$.

The ADRC scheme is tuned empirically using simple pole placement techniques. The simulation results of the reference trajectory and the actual motion in term of the Cartesian end effector coordinates are given in Figure 2.2a. Observe that perfect tracking is almost reached, meaning that we are indeed controlling a nonlinear system using linear tools using only a chain of pure integrators as model. Furthermore, Figure 2.2b depicts the two estimated lumped disturbances $\hat{\xi}_\theta$ and $\hat{\xi}_\phi$ and the control inputs u_1 and u_2. It can be seen that the control inputs and the estimated lumped disturbances are near mirror images of each other. Indicating that the disturbances are indeed estimated and rejected.



(a) (b)

Figure 2.2: Results simulation. (a) Tracking circular reference. (b) Estimated lumped disturbances and control signals.

2.2 Performance Criteria

In this section, the metrics used for the synthesis and their LMI formulations are explained. Starting with the H_∞ and H_2-norms, in Section 2.2.1 and 2.2.2, respectively, followed by ISS in Section 2.2.3.

The following notation and definitions are used in this section. $P \succ 0$ stands for P being positive definite, when P is negative definite we use $\prec$. Similarly, $\succeq$ and $\preceq$ stand for positive semi-definite and negative semi-definite, respectively. Note that for the H_∞ and H_2-norms only strict inequalities are used in this work. A continuous function $\psi : [0, a) \to [0, \infty)$ belongs to class $\mathcal{K}$ if it is strictly increasing and $\psi(0) = 0$. A continuous function $\upsilon : [0, a) \times [0, \infty) \to [0, \infty)$ belongs to class $\mathcal{KL}$, if for each fixed s, the mapping υ(r,s) belongs to class $\mathcal{K}$ with respect to r and for each fixed r the mapping $\upsilon(r, s)$ is decreasing with respect to s and $\upsilon(r, s) \to 0$ as $s \to 0$, [22]. Furthermore, $\|u\|_2 = \sqrt{\int_{-\infty}^{\infty} u^\top u \; dt}$, and $\|u\| = \sqrt{u^\top u}$.

2.2.1 H_∞ Performance Specification

Consider the strictly proper system [23]

$$\begin{aligned} \dot{x} &= \mathcal{A}x + \mathcal{B}_w w, \\ z &= \mathcal{C}_z x, \end{aligned} \tag{2.21}$$

with state x, disturbance w, and output z. Matrices $\mathcal{A}$, $\mathcal{C}_z$, $\mathcal{B}_w$ are referred to as the system matrix, the performance output matrix, and the disturbance input matrix, respectively. Furthermore, matrix, $\mathcal{A}$ is assumed to be Hurwitz.

The L_2-gain of system (2.21) from disturbance w to output z is defined as:

$$\sup_{\|w\| \neq 0} \frac{\|z\|_2}{\|w\|_2}. \tag{2.22}$$

In [24, p. 54], it is proved that if the exists $\gamma \in \mathbb{R}_+$ and positive definite matrix P satisfying the following LMI:

$$\begin{bmatrix} \mathcal{A}^\top P + P\mathcal{A} & P\mathcal{B}_w & \mathcal{C}_z^\top \\ * & -\gamma^2 I & 0 \\ * & * & -I \end{bmatrix} \prec 0, \tag{2.23}$$

then the L_2-norm of system (2.21) is upper bounded by γ (the so-called bounded real lemma).

For system (2.21), the L_2-gain equals the H_∞-norm of the transfer function matrix of (2.21) [23, p. 91]:

$$G(s) = \mathcal{C}_z(sI - \mathcal{A})^{-1}\mathcal{B}w. \tag{2.24}$$

Hence, $\|G(s)\|_\infty \leq \gamma$.

The H_∞-norm is often denoted as

$$\|G(s)\|_\infty = \sup_\omega \bar{\sigma}(G(j\omega)), \tag{2.25}$$

with $\bar{\sigma}$ the maximum singular value of $G(s)$. Hence, an interpretation for the H_∞-norm is: the worst-case frequency in the worst-case direction. LMI (2.23) is referred to as the "H_∞ LMI" for the remainder of this thesis. For the ADRC scheme, the H_∞-norm is a good candidate when we want to focus on disturbances in the worst-case frequency in the worst-case direction.

2.2.2 H_2 Performance Specification

Consider system (2.21) with transfer function matrix (2.24). For this system, the H_2-norm is defined as:

$$\|G(s)\|_2 = \frac{1}{2\pi}\mathrm{tr}\int_{-\infty}^{\infty} G^H(j\omega)G(j\omega)d\omega. \tag{2.26}$$

In [24, p. 57], it is proved that inequality $\|G(s)\|_2 < \gamma$ is equivalent to the following set of LMIs:

$$\begin{aligned} \begin{bmatrix} \mathcal{A}^\top P + P\mathcal{A} & P\mathcal{B}_w \\ * & -\gamma I \end{bmatrix} &\prec 0, \\ \begin{bmatrix} P & \mathcal{C}_z^\top \\ * & Z \end{bmatrix} &\succ 0, \\ tr(Z) &< \gamma, \end{aligned} \tag{2.27}$$

for some $P = P^\top \succ 0$, $Z = Z^\top \succ 0$, and $\gamma \in \mathbb{R}_+$. Thus, if (2.27) holds; then, the H_2-norm (2.26) is upper bounded by γ. The set of LMIs (2.27) is referred to as the "H_2 LMI" for the remainder of this thesis.

The H_2-norm can be written as

$$\|G(s)\|_2 = \sqrt{\frac{1}{2\pi}\int_{-\infty}^{\infty}\sum_i \sigma_i^2 G(j\omega)d\omega}, \tag{2.28}$$

with σ_i the i^{th} singular value of $G(j\omega)$. Hence, we can interpret minimizing the H_2-norm as minimizing the gain of the average frequency in the average direction. So, to tune the ADRC scheme the H_2-norm is interesting when we want to consider the average performance over all frequencies.

The H_2-norm can also be interpreted in the time-domain in a stochastic and deterministic manner. The H_2-norm can be found by applying a unit impulse to the i^{th} input and waiting for the output to settle to zero before applying a unit impulse to the next input. The H_2-norm is the maximum 2-norm of the output using this method. This deterministic interpretation can be written as:

$$\|G(s)\|_2 = \max_{w=\text{unit impulses}} \|z\|_2. \tag{2.29}$$

For the stochastic interpretation, we can say that minimizing the H_2-norm minimizes the output power (steady-state covariance) of the system while driven by a unit intensity white noise input. In other words, we are minimizing the RMS value of z. This interpretation indicates that the H_2-norm can be used to tune the ADRC scheme such that the (average) effects of measurement noises on a certain performance output are minimized.

2.2.3 ISS Performance Specification

For the ISS criterion, the definitions in [25] are adapted. Consider system (2.21), if there exist functions $\beta \in \mathcal{KL}$ and $\gamma \in \mathcal{K}$ satisfying

$$\|x(t)\| \leq \beta(\|x(0)\|, t) + \gamma(\sup_{0\leq\tau\leq t} \|w(\tau)\|), \tag{2.30}$$

for all u, $x(0)$, and $t \geq 0$, along the trajectories of (2.21); then, system (2.21) is said to be Input-to-State Stable (ISS) with ISS-gain $\gamma(\cdot)$.

A quadratic ISS Lyapunov-function [26] for system (2.21) is defined as $V = x^\top Px$ satisfying:

$$\begin{aligned} &\|x\|^2 \leq x^\top Px \leq \gamma\|x\|^2, \\ &\dot{x}^\top Px + x^\top P\dot{x} \leq -\|x\|^2 + \gamma\|w\|^2, \end{aligned} \tag{2.31}$$

with P a positive definite matrix and $\gamma \in \mathbb{R}_{\geq 1}$. If there exists P and γ satisfying (2.31), along the trajectories of (2.21), system (2.21) is ISS.

Condition (2.31) is equivalent to the LMI:

$$\begin{bmatrix} \mathcal{A}^\top P + P\mathcal{A} + I & \mathcal{B}_w^\top P \\ * & -\gamma I \end{bmatrix} \preceq 0, \tag{2.32}$$

with $0 \preceq P \preceq (\gamma - 1)I$. For a derivation of LMI (2.32), the reader is referred to Section 2.2.3. The scalar γ in (2.32) equals the ISS-gain γ of (2.21), see [27] for details. The choice to select the bounds as in (2.31) is made to ease the analysis (at cost of conservatism). When we do not use twice the same scalar γ in the upper bounds in (2.31), the definition for the ISS-gain in (2.30) changes and an extra decision variable is introduced. Furthermore, a (nonlinear) cost function has to be evaluated in order to minimize the resulting ISS-gain.

The ISS performance specification can be used for the tuning of ADRC schemes when we want to upper bound the worst-case disturbance amplification in the time-domain.

Chapter 3

LMIs for ADRC

In order to tune the ADRC scheme in the observer/controller form using an LMI-based synthesis approach, multiple steps are necessary. These steps are described in this chapter. The first step is to write a general closed loop form for ADRC. Since matrices K and L are the variables to be determined, they should be explicitly expressed in the LMI formulation. The derivation of this closed loop form is described in Section 3.1. Next, a two-step synthesis approach is introduced in Section 3.2. Subsequently, LMIs for various performance specifications are derived. In particular, we consider the H_∞ and H_2-norms, and ISS (see Section 2.2). The characterizations of these specifications in terms of LMIs is given in in Sections 3.3, 3.4, and 3.5, respectively. Furthermore, LMIs are derived in Section 3.6 to limit the sensitivity to measurement noises on the observer for both the H_∞ and H_2-norms.

3.1 Closed loop ADRC

In this section, a closed loop representation for ADRC suitable for the envisioned synthesis is derived. Note that in this derivation the measurement noise ν is included.

Consider the extended dynamics in (2.11)-(2.12) (with $y = Cx + \nu$), and the Luenberger observer (2.13). Let the estimation error be given by $e_\eta = \text{col}(e_x, e_{\xi_0}, \ldots, e_{\xi_{r-1}}) = \eta - \hat{\eta}$. We can write the estimation error dynamics

$$\dot{e}_\eta = \dot{\eta} - \dot{\hat{\eta}} = A_e\eta + B_{ue}u + B_{\xi e}\xi^{(r)} - ((A_e - LC_e)\hat{\eta} + L\nu + LC_e\eta + B_{ue}u). \tag{3.1}$$

Using the fact $\hat{\eta} = \eta - e_\eta$ yields

$$\begin{aligned}\dot{e}_\eta &= A_e\eta + B_{ue}u + B_{\xi e}\xi^{(r)} - ((A_e - LC_e)(\eta - e_\eta) + L\nu + LC_e\eta + B_{ue}u) \\ &= (A_e - LC_e)e_\eta - L\nu + B_{\xi e}\xi^{(r)}.\end{aligned} \tag{3.2}$$

Consider the feedback

$$u = -R^{-1}(K(\hat{x} - x^*) + \hat{\xi}_0). \tag{3.3}$$

Using $\hat{x} = x - e_x$, and $\hat{\xi}_0 = \xi_0 - e_{\xi_0}$, we can write

$$u = -R^{-1}(K(x - e_x - x^*) + \xi_0 - e_{\xi_0}). \tag{3.4}$$

Substitution in system (2.7) yields

$$\dot{x} = Ax + B_u u + B_\xi \xi = (A - B_\xi K)x + B_\xi(K(e_x + x^*) + e_\xi). \tag{3.5}$$

Combining the plant dynamics (3.5) with the estimation error dynamics (3.2), we can write the closed loop dynamics in terms of estimation errors:

$$\begin{bmatrix}\dot{x}\\ \dot{e}_\eta\end{bmatrix} = \underbrace{\begin{bmatrix}A-B_\xi K & \overline{BK}\\ 0 & A_e-LC_e\end{bmatrix}}_{\mathcal{A}_{cl}}\begin{bmatrix}x\\ e_\eta\end{bmatrix} + \begin{bmatrix}0\\ \overline{B_\xi}\end{bmatrix}\xi^{(r)} - \begin{bmatrix}0\\ L\end{bmatrix}\nu + \begin{bmatrix}B_\xi K\\ 0\end{bmatrix}x^*, \tag{3.6}$$

where

$$\overline{BK} := [B_\xi K, B_\xi, 0], \qquad \overline{B_\xi} := \begin{bmatrix}0\\ \vdots\\ 0\\ I\end{bmatrix}. \tag{3.7}$$

A detailed example for a second-order SISO system is given in Appendix A.

3.2 Two-step synthesis

In this section, a two-step synthesis approach is introduced. The H_∞ LMIs as described in Section 2.2.1 are used as an example; however, all results in this section are also valid for the H_2 and ISS specifications described in Sections 3.4 and 3.5, respectively. As introduced in Section 2.2, the H_∞ LMI (2.23) is given by

$$\begin{bmatrix}\mathcal{A}^\top P + P\mathcal{A} & PB_w & C_z^\top\\ * & -\gamma^2 I & 0\\ * & * & -I\end{bmatrix} \prec 0, \tag{3.8}$$

for some positive definite P and positive constant γ.

In the ADRC case, $\mathcal{A}$ equals $\mathcal{A}_{cl}$ in (3.6), and $B_w = \begin{bmatrix}0 & 0 & B_\xi K\\ \overline{B}_\xi & -L & 0\end{bmatrix}$. Substituting $\mathcal{A}_{cl}$ in (3.8) leads to the following nonlinear problem

$$\begin{bmatrix}\begin{bmatrix}A-B_\xi K & \overline{BK}\\ 0 & A_e-LC_e\end{bmatrix}^\top \begin{bmatrix}P_{11} & P_{12}\\ P_{12} & P_{22}\end{bmatrix} + \begin{bmatrix}P_{11} & P_{12}\\ P_{12} & P_{22}\end{bmatrix}\begin{bmatrix}A-B_\xi K & \overline{BK}\\ 0 & A_e-LC_e\end{bmatrix} & \begin{bmatrix}P_{11} & P_{12}\\ P_{12} & P_{22}\end{bmatrix}B_w & C_z^\top\\ * & -\gamma^2 I & 0\\ * & * & -I\end{bmatrix} \prec 0,$$

where B_w can depend on L and/or K depending on which inputs are considered. In order to make the synthesis problem tractable, it is necessary to convexify it, meaning transforming the matrix back into an LMI, as products PK and PL are nonlinear in the decision variables $P_{11}, P_{12}, P_{22}, K$, and L. A similar problem is recognized in the general dynamic output feedback framework, where controllers in state-space form with general matrices A_k, B_k, C_k, D_k are designed [28]. However, when a specific structure in A_k, B_k, C_k, D_k is required (as is the case here), and the matrices K and L need to be determined explicitly, the problem becomes intractable (as pointed out in [29], [30]).

To overcome this obstacle, we perform a two-step synthesis approach, where first the estimation error dynamics are considered to determine L and subsequently, the system dynamics to determine K.

Consider again the closed loop system (3.6)-(3.7) and let $\nu = x^* = 0$, i.e., without considering noise and references:

$$\begin{bmatrix}\dot{x}\\ \dot{e}_\eta\end{bmatrix} = \underbrace{\begin{bmatrix}A-B_\xi K & \overline{BK}\\ 0 & A_e-LC_e\end{bmatrix}}_{\mathcal{A}_{cl}}\begin{bmatrix}x\\ e_\eta\end{bmatrix} + \begin{bmatrix}0\\ \overline{B_\xi}\end{bmatrix}\xi^{(r)}. \tag{3.9}$$

Note that only the disturbance $\xi^{(r)}$ is considered here. The measurement noise ν will be treated in Section 3.6. Moreover, the reference signal x^* enters the dynamics in the same manner as the part of the estimation error e_η that contains the estimation errors in x, referred to as e_x. Since no tuning/weighting

matrices will be used for e_x or x^*, the transfer function matrices that relate disturbances e_x and x^* to the performance output are equal. Hence, minimizing the influence of disturbance e_x on the performance output will also minimize the effect of the reference signal on this output.

The closed loop system (3.6) can be written as a series interconnection of

$$\begin{aligned} \dot{e}_\eta &= (A_e - LC_e)e_\eta + \overline{B_\xi}\xi^{(r)} = \mathcal{A}_l e_\eta + \overline{B_\xi}\xi^{(r)}, \\ y_e &= C_{ze} e_\eta, \end{aligned} \tag{3.10}$$

and

$$\begin{aligned} \dot{x} &= (A - B_\xi K)x + \overline{BK}e_\eta = \mathcal{A}_k x + [B_\xi K \ B_\xi] y_e, \\ y_x &= Cx, \end{aligned} \tag{3.11}$$

with $C_{ze} = \begin{bmatrix} I_{(n+m)} & 0 \end{bmatrix}$.* Which are referred to as the estimation error dynamics (3.10) and plant dynamics (3.11), respectively. For tuning purposes, two performance outputs, z_e and z_x, are defined. These performance outputs are in general not equal to the measured outputs y_e and y_x. In Figure 3.1, a schematic representation of the interconnection is depicted.

What are the implications of separately tuning the series interconnection of (3.10) and (3.11) using system norms H_∞ and H_2? The H_∞-norm is an induced norm and thus a matrix norm. Therefore, it satisfies the multiplicative property [31, p. 159-160]:

$$\|G(s)H(s)\|_\infty \leq \|G(s)\|_\infty \|H(s)\|_\infty \tag{3.12}$$

with $G(s)$ and $H(s)$ transfer function matrices, which in this case represent the estimation error and the plant dynamics, respectively. Hence, when we separately tune the estimation error and the plant dynamics and we find upper bounds γ_l and γ_k on the H_∞ gains for the two subsystems. Then we can conclude that $\gamma \leq \gamma_l \gamma_k$, with γ, γ_l, and γ_k upper bounds on the H_∞-norm of the total interconnection, the estimation error and the plant dynamics, respectively.

The H_2-norm however is not an induced norm and does in general not satisfy the multiplicative property [31, p. 159-160]. Therefore, when the two subsystems are tuned separately, the H_2-norm of the total interconnection must be considered.

*Matrix C_{ze} selects only the estimated states $\hat{x}$ and the estimated lumped disturbance $\hat{\xi}_0$ and not the higher derivatives.

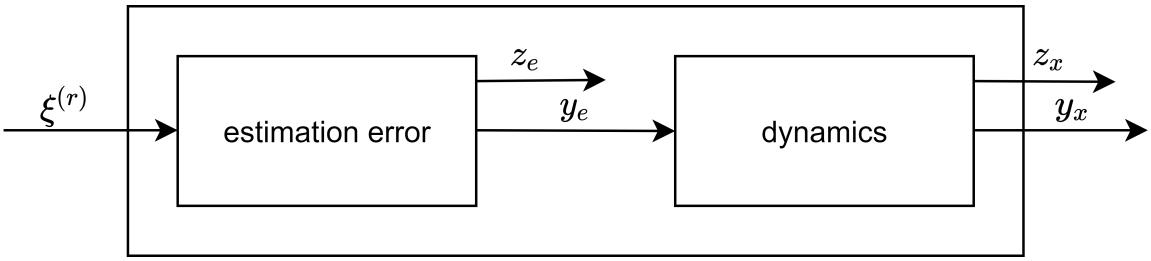


Figure 3.1: Series representation closed loop

3.3 H_∞ LMIs

3.3.1 H_∞ observer

In this section an LMI is derived that can be used to determine matrix L of the observer such that the H_∞-norm between $\xi^{(r)}$ and z_e is minimized.

The estimation error dynamics forced by the r^{th} derivative of the lumped disturbance including disturbance output z_e is given by

$$\begin{aligned} \dot{e}_\eta &= \mathcal{A}_l e_\eta + \overline{B_\xi}\xi^{(r)}, \\ y_e &= C_{ze} e_\eta, \\ z_e &= W_e C_{ze} e_\eta = W_e y_e, \end{aligned} \tag{3.13}$$

and $W_e \in \mathbb{R}^{(n+m)\times(n+m)}$ a diagonal weighting matrix used for tuning purposes. Matrix W_e has the following structure

$$W_e = \begin{bmatrix} I_n & 0 \\ 0 & \frac{1}{w_\xi} I \end{bmatrix}, \tag{3.14}$$

for some tuning parameter $w_\xi \in \mathbb{R}_+$.

Since only the estimation errors in y_e are injected into the plant dynamics only these errors are essential for the tracking problem. The errors in the derivative(s) of the lumped disturbance are less relevant in this case. If for a specific application the estimation error of the derivative(s) of the lumped disturbance is important then the definition of C_{ze} can be changed to account for it.

Substituting (3.13) in the H_∞ LMI (3.8) gives

$$\begin{bmatrix} \mathcal{A}_l^\top P + P\mathcal{A}_l & P\overline{B_\xi} & (W_e C_{ze})^\top \\ * & -{\gamma_l}^2 I & 0 \\ * & * & -I \end{bmatrix} \prec 0, \tag{3.15}$$

with $P = P^\top \succ 0 \in \mathbb{R}^{(n+mr)\times(n+mr)}$, and $\gamma_l \in \mathbb{R}_+$.

Consider, the change of variables

$$L = P^{-1}\hat{L}, \;\; \gamma_l = \sqrt{\hat{\gamma}_l}, \tag{3.16}$$

which makes the system of inequalities convex in $\hat{L}$ and $\hat{\gamma}_l$,

$$\begin{bmatrix} A_e^\top P + PA_e - {C_e}^\top \hat{L}^\top - \hat{L}C_e & P\overline{B_\xi} & (W_e C_{ze})^\top \\ * & -\hat{\gamma}_l I & 0 \\ * & * & -I \end{bmatrix} \prec 0. \tag{3.17}$$

This is an LMI that can be solved for $\hat{L}$ and P, minimizing $\hat{\gamma}_l$, and the optimal L and γ_l can be recovered using (3.16).

3.3.2 H_∞ controller

In this section, an LMI is derived that can be used to determine matrix K such that the H_∞-norm between y_e and z_x is minimized.

For the plant dynamics part, the relevant part of the estimation error is treated as an external disturbance amplified by gain $\overline{BK}$:

$$\begin{aligned} \dot{x} &= \mathcal{A}_k x + [B_\xi K \;\; B_\xi] y_e = \mathcal{A}_k x + [B_\xi K \;\; B_\xi] W_e^{-1} z_e = \mathcal{A}_k x + [B_\xi K \;\; B_\xi w_\xi] z_e, \\ y_x &= Cx, \\ z_x &= C_{zx} x, \end{aligned} \tag{3.18}$$

and constant w_ξ defined in (3.14). For convenience $B_\xi w_\xi := B_{w\xi}$ is defined.

Substitution of (3.18) in LMI (3.8) yields

$$\begin{bmatrix} \mathcal{A}_k^\top Q + Q\mathcal{A}_k & Q[B_\xi K \; B_{w\xi}] & C_{zx}^\top \\ * & -\gamma_k^2 I & 0 \\ * & * & -I \end{bmatrix} \prec 0, \tag{3.19}$$

with $Q = Q^\top \succ 0 \in \mathbb{R}^{n\times n}$, $\gamma_k \in \mathbb{R}_+$.

Expanding $\mathcal{A}_k$ gives

$$\begin{bmatrix} A^\top Q + QA - K^\top {B_\xi}^\top Q - QB_\xi K & Q\begin{bmatrix} B_\xi K & B_{w\xi} \end{bmatrix} & C_{zx}^\top \\ * & -\gamma_k^2 I & 0 \\ * & * & -I \end{bmatrix} \prec 0. \tag{3.20}$$

Next, the matrix is expanded by splitting the term $Q\begin{bmatrix} B_\xi K & B_{w\xi} \end{bmatrix}$:

$$\begin{bmatrix} A^\top Q + QA - K^\top {B_\xi}^\top Q - QB_\xi K & QB_\xi K & QB_{w\xi} & C_{zx}^\top \\ * & -\gamma_k^2 I & 0 & 0 \\ * & * & -\gamma_k^2 I & 0 \\ * & * & * & -I \end{bmatrix} \prec 0. \tag{3.21}$$

Introducing $S = Q^{-1}$ and performing a congruence transformation by pre- and post -multiplying by $diag(S, S, I, I)$ gives

$$\begin{bmatrix} SA^\top + AS - SK^\top {B_\xi}^\top - B_\xi KS & B_\xi KS & B_{w\xi} & SC_{zx}^\top \\ * & -\gamma_k^2 S^2 & 0 & 0 \\ * & * & -\gamma_k^2 I & 0 \\ * & * & * & -I \end{bmatrix} \prec 0. \tag{3.22}$$

Observe that the S^2 term in (3.22) is not easily removed using the Schur complement or a variable substitution. Using a special case of Young's relation $H^\top F^{-1} H \succeq H + H^\top - F$ [24, p. 24] gives

$$SIS \succeq 2S - I \Rightarrow -S^2 \preceq -2S + I. \tag{3.23}$$

Substitution of $-S^2$ in (3.22) by $-2S + I$ gives $\gamma_k^2(-2S + I)$ which is still bi-linear. Let $\gamma_k^2 := \mu$.

Substitution of $\mu(I - 2S)$ for the $-\gamma_k^2 S^2$ term in (3.22) gives

$$\begin{bmatrix} SA^\top + AS - SK^\top {B_\xi}^\top - B_\xi KS & B_\xi KS & B_{w\xi} & SC_{zx}^\top \\ * & \mu(I - 2S) & 0 & 0 \\ * & * & -\mu I & 0 \\ * & * & * & -I \end{bmatrix} \prec 0. \tag{3.24}$$

This new matrix (3.24) is an upper bound on matrix (3.22). Hence we have (3.22) $\prec$ (3.24) $\prec 0$, therefore, by enforcing (3.24) $\prec 0$, we guarantee (3.22) $\prec 0$ (by transitivity).

Setting $K = \hat{K}S^{-1}$ in (3.24) yields the following LMI for fixed values of μ

$$\begin{bmatrix} SA^\top + AS - \hat{K}^\top B_\xi{}^\top - B_\xi \hat{K} & B_\xi \hat{K} & B_\xi w_\xi & SC_{zx}^\top \\ * & \mu(I-2S) & 0 & 0 \\ * & * & -\mu I & 0 \\ * & * & * & -I \end{bmatrix} \prec 0. \tag{3.25}$$

A line search can be performed to determine which value of μ results in the smallest value for γ_k. After S and $\hat{K}$ are determined, using the optimal value of μ, K can be recovered using $K = \hat{K}S^{-1}$.

3.4 H_2 LMIs

In this section, the LMI for the H_2-norm as defined in Section 2.2.2 is used to determine an LMI to tune the observer and controller of the ADRC scheme. Note that the steps taken are similar to those to derive the LMIs for the H_∞-norm. However, the tuning parameter w_ξ is in this case not considered (required).

3.4.1 H_2 observer

Recall the H_2 LMIs in (2.27):

$$\begin{aligned} \begin{bmatrix} \mathcal{A}^\top P + P\mathcal{A} & PB_w \\ * & -\gamma_l I \end{bmatrix} &\prec 0, \\ \begin{bmatrix} P & C_z^\top \\ * & Z_l \end{bmatrix} &\succ 0, \\ tr(Z_l) &< \gamma_l, \end{aligned} \tag{3.26}$$

with $P = P^\top \succ 0$, $Z_l = Z_l^\top \succ 0$, and $\gamma_l > 0$.

The closed loop dynamics for the estimation error is given by

$$\begin{aligned} \dot{e}_\eta &= \mathcal{A}_l e_\eta + \overline{B_\xi}\xi^{(r)}, \\ y_e &= C_{ze} e_\eta, \\ z_e &= C_{ze} e_\eta. \end{aligned} \tag{3.27}$$

So no weighting here. Letting $\mathcal{A}$, B_w and C_z in (3.26) equal $\mathcal{A}_l$, $\overline{B_\xi}$, and C_{ze}, respectively, and applying the change of variable $\hat{L} = PL$ gives

$$\begin{aligned} \begin{bmatrix} A_e^\top P + PA_e - C_{ze}^\top \hat{L}^\top - \hat{L}C_{ze} & P\overline{B_\xi} \\ * & -\gamma_l I \end{bmatrix} &\prec 0, \\ \begin{bmatrix} P & C_{ze}^\top \\ * & Z_l \end{bmatrix} &\succ 0, \\ tr(Z_l) &< \gamma_l, \end{aligned} \tag{3.28}$$

with $P = P^\top \succ 0 \in \mathbb{R}^{(n+mr)\times(n+mr)}$, $Z_l \in \mathbb{R}^{m\times m}$, and $\gamma_l \in \mathbb{R}_+$, which are LMIs that can be solved for P, Z_l, and $\hat{L}$ while minimizing γ_l. Finally, L is recovered using $\hat{L}P^{-1} = L$.

3.4.2 H_2 controller

Starting with substitution of the closed loop dynamics

$$\begin{aligned} \dot{x} &= (A - B_\xi K)x + \overline{BK}e_\eta = \mathcal{A}_k x + [B_\xi K \ B_\xi] z_e, \\ y &= Cx, \\ z &= C_{zx} x, \end{aligned} \tag{3.29}$$

in (3.26) yields

$$\begin{aligned} \begin{bmatrix} \mathcal{A}_k^\top Q + Q\mathcal{A}_k & Q[B_\xi K \; B_\xi] \\ * & -\gamma_k I \end{bmatrix} &\prec 0, \\ \begin{bmatrix} Q & C_{zx}^\top \\ * & Z_k \end{bmatrix} &\succ 0, \\ tr(Z_k) &< \gamma_k, \end{aligned} \tag{3.30}$$

with $Q = Q^\top \succ 0 \in \mathbb{R}^{n\times n}$, $Z_k = Z_k^\top \succ 0 \in \mathbb{R}^{m\times m}$, and $\gamma_k \in \mathbb{R}_+$. Let's start with the first of the three inequalities. Expanding $\mathcal{A}_k$, and splitting $[B_\xi K \; B_\xi]$ gives

$$\begin{bmatrix} (A - B_\xi K)^\top Q + Q(A - B_\xi K) & QB_\xi K & QB_\xi \\ * & -\gamma_k I & 0 \\ * & * & -\gamma_k I \end{bmatrix} \prec 0. \tag{3.31}$$

Introducing $S = Q^{-1}$, and pre- and post -multipling with $\mathrm{diag}(S, S, I)$ yields

$$\begin{bmatrix} SA^\top + AS - SK^\top B_\xi^\top - B_\xi K S & B_\xi K S & B_\xi \\ * & -\gamma_k S^2 & 0 \\ * & * & -\gamma_k I \end{bmatrix} \prec 0. \tag{3.32}$$

Setting $K = S^{-1}\hat{K}$, $S^2 = I - 2S$ and $\gamma_k = \mu$ yields the following LMI for fixed values of $\mu \in \mathbb{R}_+$

$$\begin{bmatrix} SA^\top + AS - \hat{K}^\top B_\xi^\top - B_\xi \hat{K} & B_\xi \hat{K} & B_\xi \\ * & \mu(I - 2S) & 0 \\ * & * & -\mu I \end{bmatrix} \prec 0. \tag{3.33}$$

We next consider,

$$\begin{bmatrix} Q & C_{zx}^\top \\ * & Z_k \end{bmatrix} \succ 0 \tag{3.34}$$

which is still in Q variables. A congruence transformation transforms it into S variables. Defining $S = Q^{-1}$, and pre- and post -multiplying with $\mathrm{diag}(S, I)$ gives

$$\begin{bmatrix} S & SC_{zx}^\top \\ * & Z_k \end{bmatrix} \succ 0. \tag{3.35}$$

So, the total set of LMIs for the the controller part is:

$$\begin{aligned} \begin{bmatrix} SA^\top + AS - \hat{K}^\top B_\xi^\top - B_\xi \hat{K} & B_\xi \hat{K} & B_\xi \\ * & \mu(I - 2S) & 0 \\ * & * & -\mu I \end{bmatrix} &\prec 0, \\ \begin{bmatrix} S & SC_{zx}^\top \\ * & Z_k \end{bmatrix} &\succ 0, \\ tr(Z_k) &< \mu, \end{aligned} \tag{3.36}$$

with $S = S^\top \succ 0$, $Z_k = Z_k^\top \succ 0$, $\mu = \gamma_k > 0$, and $K = S^{-1}\hat{K}$.

3.5 ISS LMIs

3.5.1 ISS observer

The goal of this section is to determine an LMI for the estimation error dynamics using an ISS Lyapunov function.

Recall the estimation error dynamics

$$\dot{e}_\eta = \mathcal{A}_l e_\eta + \overline{B_\xi}\xi^{(r)}. \tag{3.37}$$

For convenience the following notations are adopted $e_\eta = e$, and $\overline{B_\xi}\xi^{(r)} = Bv$, giving

$$\dot{e} = \mathcal{A}_l e + Bv. \tag{3.38}$$

The following quadratic Lyapunov candidate function is selected

$$V_e(e) = e^\top P e. \tag{3.39}$$

Hence,

$$\dot{V}_e(e,v) = \dot{e}^\top P e + e^\top P \dot{e} = (\mathcal{A}_l e + Bv)^\top P e + e^\top P(\mathcal{A}_l e + Bv). \tag{3.40}$$

Next, V_e is lower and upper bounded as follows for a certain $\gamma_l \in \mathbb{R}_{\geq 1}$ and $P = P^\top \succ 0 \in \mathbb{R}^{(n+mr)\times(n+mr)}$

$$\begin{aligned} &\|e\|^2 \leq e^\top P e \leq \gamma_l \|e\|^2, \\ &\dot{V}_e(e,v) \leq -\|e\|^2 + \gamma_l \|v\|^2. \end{aligned} \tag{3.41}$$

These conditions (ISS-Lyapunov) are cast into an LMI. Using $\|e\|^2 = \sqrt{e^\top e}^2 = e^\top e$ we can write

$$\dot{V}_e(e,v) = \begin{bmatrix} e^\top & v^\top \end{bmatrix} \begin{bmatrix} \mathcal{A}_l^\top P + P\mathcal{A}_l & PB \\ * & 0 \end{bmatrix} \begin{bmatrix} e \\ v \end{bmatrix} \leq -e^\top e + \gamma_l v^\top v. \tag{3.42}$$

Placing the right hand terms into the matrix gives

$$\begin{bmatrix} e^\top & v^\top \end{bmatrix} \begin{bmatrix} \mathcal{A}_l^\top P + P\mathcal{A}_l + I & PB \\ * & -\gamma_l I \end{bmatrix} \begin{bmatrix} e \\ v \end{bmatrix} \leq 0. \tag{3.43}$$

Hence,

$$\begin{bmatrix} \mathcal{A}_l^\top P + P\mathcal{A}_l + I & PB \\ * & -\gamma_l I \end{bmatrix} \preceq 0. \tag{3.44}$$

Expanding $\mathcal{A}_l$ and setting $L = P^{-1}\hat{L}$ gives

$$\begin{bmatrix} {A_e}^\top P + P A_e - C_e^\top \hat{L}^\top - \hat{L} C_e + I & PB \\ * & -\gamma_l I \end{bmatrix} \preceq 0. \tag{3.45}$$

Which is an LMI that can be solved for P and $\hat{L}$ while minimizing γ_l.

3.5.2 ISS controller

Recall, the closed loop dynamics

$$\dot{x} = (A - BK)x + \overline{BK}e_\eta. \tag{3.46}$$

Which is for convenience denoted as

$$\dot{x} = \mathcal{A}_k x + \overline{BK}e. \tag{3.47}$$

The following Lyapunov candidate is proposed

$$V_x(x) = x^\top Q x. \tag{3.48}$$

Hence,

$$\dot{V}_x(x,e) = \dot{x}^\top Q x + x^\top Q \dot{x} = (\mathcal{A}_k x + \overline{BK}e)^\top Q x + x^\top Q(\mathcal{A}_k x + \overline{BK}e). \tag{3.49}$$

Which is bounded as follows

$$\begin{aligned} &\|x\|^2 \leq x^\top Q x \leq \gamma_k \|x\|^2, \\ &\dot{V}_x(x,e) \leq -\|x\|^2 + \gamma_k \|e\|^2. \end{aligned} \tag{3.50}$$

Substitution of $\dot{V}_x(x,e)$ and recasting into a matrix gives

$$\dot{V}_x(x,e) = \begin{bmatrix} x^\top & e^\top \end{bmatrix} \begin{bmatrix} \mathcal{A}_k^\top Q + Q\mathcal{A}_k & Q\overline{BK} \\ * & 0 \end{bmatrix} \begin{bmatrix} x \\ e \end{bmatrix} \leq -x^\top x + \gamma_k e^\top e. \tag{3.51}$$

Therefore, we require

$$\begin{bmatrix} \mathcal{A}_k^\top Q + Q\mathcal{A}_k + I & Q\overline{BK} \\ * & -\gamma_k I \end{bmatrix} \preceq 0. \tag{3.52}$$

As seen in Section 3.3.2, $\overline{BK}$ can be split, resulting in

$$\begin{bmatrix} \mathcal{A}_k^\top Q + Q\mathcal{A}_k + I & QB_\xi K & QB_\xi \\ * & -\gamma_k I & 0 \\ * & * & -\gamma_k I \end{bmatrix} \preceq 0. \tag{3.53}$$

Let's convexify this matrix inequality by introducing $S = Q^{-1}$ and pre- and post -multiplying with $diag(S, S, I)$ which gives

$$\begin{bmatrix} S\mathcal{A}_k^\top + \mathcal{A}_k S + S^2 & B_\xi K S & B_\xi \\ * & -\gamma_k S^2 & 0 \\ * & * & -\gamma_k I \end{bmatrix} \preceq 0. \tag{3.54}$$

Note, that two terms with a S^2 term are obtained. One with a positive sign and the other with a negative sign which need to be upper and lower bounded respectively. Let's start with the lower bound.

Using $HF^{-1}H \succeq H + H^\top - F$ yields

$$SIS \succeq 2S - I \Rightarrow -S^2 \preceq -2S + I. \tag{3.55}$$

Substitution gives

$$\begin{bmatrix} S\mathcal{A}_k^\top + \mathcal{A}_k S + S^2 & B_\xi K S & B_\xi \\ * & \gamma_k(I - 2S) & 0 \\ * & * & -\gamma_k I \end{bmatrix} \preceq 0. \tag{3.56}$$

Now the remaining S^2 term has to be considered.

To impose an upper bound the following LMI is added

$$S \succeq \mu I \tag{3.57}$$

hence,

$$S^2 \succeq \mu^2 I. \tag{3.58}$$

Substitution gives

$$\begin{bmatrix} S\mathcal{A}_k^\top + \mathcal{A}_k S + \mu^2 I & B_\xi K S & B_\xi \\ * & \gamma_k(I - 2S) & 0 \\ * & * & -\gamma_k I \end{bmatrix} \preceq 0. \tag{3.59}$$

Now that the two S^2 terms are tackled, the variable substitution $K = \hat{K} S^{-1}$ is performed. Resulting in

$$\begin{bmatrix} SA^\top + AS - \hat{K}^\top B_\xi^\top - B_\xi \hat{K} + \mu^2 I & B_\xi \hat{K} & B_\xi \\ * & \gamma_k(I - 2S) & 0 \\ * & * & -\gamma_k I \end{bmatrix} \preceq 0 \tag{3.60}$$

which is an LMI for fixed $\mu \in \mathbb{R}_+$ and $\gamma_k \in \mathbb{R}_{\geq 1}$ that can be solved for $\hat{K}$ and $S \in \mathbb{R}^{n \times n}$. A grid search can be performed over the variables μ and γ_k to find the smallest feasible value for γ_k.

3.6 High-frequency output disturbances

The used observer is a high-gain observer. In an ideal world, the gain can be set arbitrarily high, so the estimation error can be made arbitrarily small (in a specific frequency range). Minimizing gain γ_l (between $\xi^{(r)}$ and z_e) typically leads to high observer gains L. However, in reality, measurement noise will enter the system and is amplified by L. So, there exists a tradeoff between the effect of the measurement noise and $\xi^{(r)}$ on z_e. Therefore, an additional performance criterion is added to the observer design.

In Section 3.4.1, the goal is minimizing the H_2/H_∞-norm between $\xi^{(r)}$ and z_e. Here the focus is on minimizing the H_2/H_∞-norm between input ν and performance output z_n. LMIs are derived to determine L such that the H_2/H_∞-norm (between ν and z_n) upper bounded by a certain scalar $\gamma_n \in \mathbb{R}_+$.

The choice to upper bound the effect of the measurement noise on the performance output is made to make it possible to mix the H_2 and H_∞ specifications, e.g., H_∞ for the disturbance $\xi^{(r)}$ and H_2 for the measurement noise ν. It is however difficult to select such an upper bound in an intuitive manner.

An alternative approach to consider the measurement noises is using $B_w = \begin{bmatrix} \overline{B}_\xi & -L \end{bmatrix}$ in the H_∞ or H_2 LMIs for the observer design. Using this approach, the tradeoff between disturbance attenuation of the effect of $\xi^{(r)}$ and that of measurement noise on the performance output is considered in the optimization problem without the need to manually select an upper bound on γ_n, however mixing the H_2 and H_∞ LMIs is not possible using this approach.

3.6.1 H_2 LMI with noise input

Consider the following estimation error dynamics omitting disturbance $B_\xi \xi^{(r)}$

$$\dot{e}_\eta = \mathcal{A}_l e_\eta - L\nu, \tag{3.61}$$

and the H_2 LMI

$$\begin{aligned} \begin{bmatrix} \mathcal{A}_l^\top P + P\mathcal{A}_l & PB_w \\ * & -\gamma_n I \end{bmatrix} &\prec 0, \\ \begin{bmatrix} P & C_z^\top \\ * & Z_l \end{bmatrix} &\succ 0, \\ tr(Z_l) &< \gamma_n \end{aligned} \tag{3.62}$$

with $P = P^\top \succ 0 \in \mathbb{R}^{(n+mr)\times(n+mr)}$, $Z_l \in \mathbb{R}^{m\times m}$, and $\gamma_n \in \mathbb{R}_+$. Expanding $\mathcal{A}_l$ and substituting B_w, C_z with $-L$, C_n respectively gives

$$\begin{bmatrix} (A_e - LC_e)^\top P + P(A_e - LC_e) + C_n^\top & -PL \\ * & -\gamma_n I \end{bmatrix} \prec 0. \tag{3.63}$$

Setting $L = P^{-1}\hat{L}$ gives

$$\begin{aligned} \begin{bmatrix} A_e^\top P + PA_e - C_e^\top \hat{L}^\top - \hat{L}C_e & -\hat{L} \\ * & -\gamma_n I \end{bmatrix} &\prec 0, \\ \begin{bmatrix} P & C_n^\top \\ * & Z_n \end{bmatrix} &\succ 0, \\ tr(Z_n) &< \gamma_n, \end{aligned} \tag{3.64}$$

which is an LMI that can be solved for $\hat{L}$ and P. This LMI can be used as extra constraint when solving LMIs (3.17) or (3.28) in the observer synthesis. It is obviously necessary to use the same $\hat{L}$ and P in all the used LMIs.

3.6.2 H_∞ LMI with noise input

In a similar fashion as in the last section the H_∞ criterion is used to derive an LMI that can be used to set a bound on the H_∞-norm between measurement noise and a certain performance output z_n.

Consider,

$$\begin{bmatrix} \mathcal{A}_l{}^\top P + P\mathcal{A}_l & PB_w & C_z^\top \\ * & -\gamma_n^2 I & 0 \\ * & * & -I \end{bmatrix} \prec 0, \tag{3.65}$$

with $P = P^\top \succ 0 \in \mathbb{R}^{(n+mr)\times(n+mr)}$, and $\gamma_n \in \mathbb{R}_+$. Expanding $\mathcal{A}_l$ and substituting B_w, C_z with $-L$, C_n, and performing the following change of variables,

$$L = P^{-1}\hat{L}, \;\; \gamma_n = \sqrt{\hat{\gamma}_n}, \tag{3.66}$$

yields,

$$\begin{bmatrix} A_e^\top P + PA_e - C_e{}^\top \hat{L}^\top - \hat{L}C_e & -\hat{L} & C_n^\top \\ * & -\hat{\gamma}_n I & 0 \\ * & * & -I \end{bmatrix} \prec 0. \tag{3.67}$$

This LMI could be used in the same manner as the one derived in Section 3.6.1.

3.7 Discussion

A closed loop matrix formulation for ADRC with r (for each input-output relation) dynamic extensions in the GPIO is derived. This closed loop is characterized by the observer gain L and control feedback K.

For the co-design of matrices L and K in terms of LMIs no convex algorithms are available in the literature. Hence, the need for a two-step approach is identified and subsequently such an approach is developed.

Using the H_∞, H_2 and ISS specifications, we derived LMIs to tune matrices K and L in an optimal manner. For the estimation error dynamics, LMIs are derived that can tune matrix L such that the specific (H_∞, H_2, ISS) gain from the lumped disturbance to the interconnection output is minimized for all three the specifications. For the plant dynamics LMIs are derived that can tune K such the gain from the interconnection input toward a performance output is minimized. The LMIs to determine K are sub-optimal. To make the optimization problems convex for the H_∞/H_2 and ISS LMIs the problem had to be simplified one and two times, respectively. Furthermore, the two H_∞ LMIs are equipped with parameter w_ξ that is used for tuning purposes in Chapter 4.

For both the H_∞ and the H_2 specifications LMIs are derived that enable to upper bound the H_∞ or the H_2-norms (in the observer) between measurement noise and a performance output z_n by scalar γ_n. These LMIs are never used solely but are used as additional constraints in the observer design.

Chapter 4

Simulation results

In this chapter, simulations are performed to demonstrate and verify the usefulness of the H_∞ LMIs derived in the previous Chapter. First, a benchmark system is introduced in Section 4.1, which is used in this chapter for simulations and in Chapter 5 for practical experiments. Next, a methodology to use the two H_∞ LMIs, (3.17), (3.25), derived in Section 3.3 is described in Section 4.2. Subsequently, a standard reference trajectory and a measure for performance are introduced in Section 4.3.1, followed by the results of the ideal world simulations in Section 4.3.2. In Section 4.4.1, results for more realistic simulations are given, and in Section 4.4.2 the resulting control signals are analyzed. The ability to reconstruct disturbances is described in Section 4.4.3. In Section 4.5, the conclusions for Chapter 4 are given.

4.1 Generalized Huygens Setup Model

The chosen system for the simulations and practical experiments is the GHS of which a schematic representation is given in Figure 4.1. The simplified equations of motion (taken from [32]) are

$$\begin{aligned}
\ddot{x}_1 &= -\omega_1^2 (x_1 - x_3) - 2\zeta_1\omega_1 (\dot{x}_1 - \dot{x}_3) + u_1 \\
\ddot{x}_2 &= -\omega_2^2 (x_2 - x_3) - 2\zeta_2\omega_2 (\dot{x}_2 - \dot{x}_3) + u_2 \\
\ddot{x}_3 &= \sum_{i=1}^{2} \mu_i \left[\omega_i^2 (x_i - x_3) + 2\zeta_i\omega_i (\dot{x}_i - \dot{x}_3) - u_i\right] - \omega_3^2 x_3 - 2\zeta_3\omega_3\dot{x}_3 + u_3
\end{aligned} \tag{4.1}$$

with $w_i = \sqrt{\frac{k_i}{m_i}}$ the undamped natural frequencies [rad/s], $\zeta_i = \frac{b_i}{2\omega_i m_i}[-]$ the dimensionless damping coefficients, and u_i $[m/s^2]$ the actuator input, $i = 1, 2, 3$. Constants $\mu_i = \frac{m_i}{m_3}$ are the coupling coefficients, $i = 1, 2$. For more information about the GHS, the interested reader is referred to [32], [33],[34].

Introducing the states $x := [x_1\ x_2\ x_3\ x_4\ x_5\ x_6]^\top := [x_1\ x_2\ x_3\ \dot{x}_1\ \dot{x}_2\ \dot{x}_3]^\top$.

The following first-order representation can be written

$$\begin{aligned}
\dot{x} &= Ax + B_u u, \\
y &= Cx
\end{aligned} \tag{4.2}$$

with

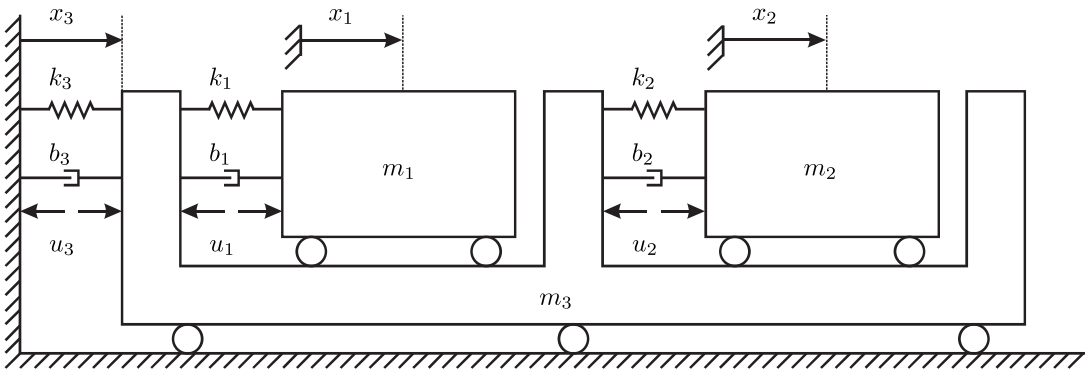


Figure 4.1: Schematic representation generalized Huygens setup. Adapted from [34].

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -\omega_1^2 & 0 & \omega_1^2 & -2\zeta_1\omega_1 & 0 & 2\zeta_1\omega_1 \\ 0 & -\omega_2^2 & \omega_2^2 & 0 & -2\zeta_2\omega_2 & 2\zeta_2\omega_2 \\ \mu_1\omega_1^2 & \mu_2\omega_2^2 & -\mu_1\omega_1^2 - \mu_2\omega_2^2 - \omega_3^2 & 2\mu_1\zeta_1\omega_1 & 2\mu_2\zeta_2\omega_2 & -2\mu_1\zeta_1\omega_1 - 2\mu_2\zeta_2\omega_2 - 2\zeta_3\omega_3 \end{bmatrix},$$
$$B_u = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ -\mu_1 & -\mu_2 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}, \quad u = \begin{bmatrix} u_1 \\ u_3 \\ u_3 \end{bmatrix}. \tag{4.3}$$

In the simulations, the following parameters are used: $\begin{bmatrix}\omega_1 & \omega_2 & \omega_3\end{bmatrix} = \begin{bmatrix}12.5521 & 14.0337 & 9.7369\end{bmatrix}$, $\begin{bmatrix}\zeta_1 & \zeta_2 & \zeta_3\end{bmatrix} = \begin{bmatrix}0.3362 & 0.4296 & 0.0409\end{bmatrix}$, and $\begin{bmatrix}\mu_1 & \mu_2\end{bmatrix} = \begin{bmatrix}0.0411 & 0.0578\end{bmatrix}$. These parameters were identified in [34].

Two different cases are considered. The first case is referred to as the GHS with full model information

$$\begin{aligned} \dot{x} &= Ax + B_u u + B_\xi \xi_0, \\ y &= Cx, \end{aligned} \tag{4.4}$$

with

$$B_\xi = \begin{bmatrix} 0 \\ I \end{bmatrix}, \tag{4.5}$$

and A, B_u, and C as defined in (4.3). Furthermore, lumped disturbance ξ_0 equals zero when no additional disturbances (e.g., external forces and measurement noise) enter the system.

The second case is referred to as the GHS without model information

$$\begin{aligned} \dot{x} &= Ax + B_u u + B_\xi \xi_0, \\ y &= Cx, \end{aligned} \tag{4.6}$$

with

$$A = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}, \quad B_u = B_\xi = \begin{bmatrix} 0 \\ I \end{bmatrix}, \quad C = \begin{bmatrix} I & 0 \end{bmatrix}. \tag{4.7}$$

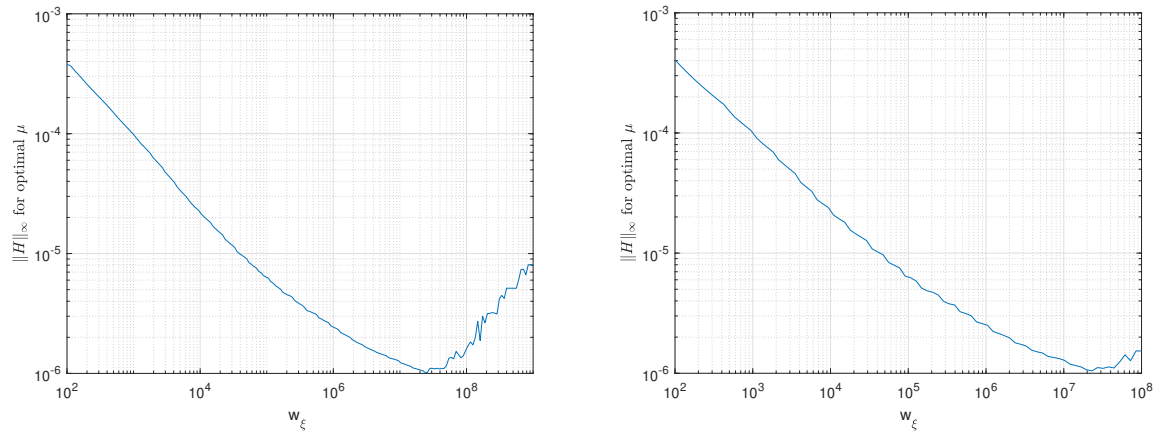
In this case lumped disturbance ξ_0 contains the unmodeled dynamics.

4.2 Practical use of the H_∞ LMI

A practical method to tune the observer and the controller that together form the ADRC scheme is explained in this section. The method makes use of the two H_∞ LMIs (3.17), (3.25) derived in Section 3.3 to perform a synthesis based tuning. It is assumed that a (simplified) linear model of the exact feedback linearizable system we want to control is available. This model is in the normal form with matrices A, B_u, and C as given in (2.7). The model should at least contain the right number of integrators for each input-output relation. Furthermore, the input matrix is assumed to be sufficiently accurate (sign definiteness and order of magnitude right). Note, that any extra linear information about the system dynamics can be added in A and B_u. These A, B_u, and C matrices together with the choice for the number of dynamic extensions r and tuning parameter w_ξ are then loaded into the MATLAB code found in Appendix B. This code generates matrices $\mathcal{A}_k$, $\mathcal{A}_l$ and total closed loop matrix $\mathcal{A}_{cl}$ in (3.9) for any $r \in \mathbb{N}_+$ and $w_\xi \in \mathbb{R}_+$. Subsequently, LMI (3.17) is solved minimizing γ_l, and matrix L is recovered using (3.16). Then, LMI (3.25) is solved iterative performing a line search over μ. Using bisection, it is possible to locate the smallest feasible μ quickly. All obtained matrices K (near the smallest feasible μ) are saved. When it is clear in which region the smallest feasible μ can be found, a more precise line search is performed (smaller steps). This time the obtained L, and all K found near the smallest feasible μ are consecutively substituted in the total closed loop dynamics (3.6) of which the actual H_∞-norm (between $\xi^{(r)}$ and z_x) is determined and stored in a vector. Finally, the K that results in the smallest H_∞-norm is selected. This process can be repeated for different values of w_ξ in a for-loop. So, effectively a grid search is performed over μ and w_ξ. However, a grid search can (depending on the grid density and search area) use much computation time. A faster approach is to manually select a value for w_ξ and perform a line search over μ. In practice, it takes less then a minute to pinpoint the optimal value for μ. This process can be repeated for different values of w_ξ until an acceptable performance is reached.

For the GHS described in Section 4.1, a grid search over μ and w_ξ is performed. For the case $r = 1$, two plots are made using no model and full model information in Figure 4.2a and Figure 4.2b, respectively. With variable w_ξ on the horizontal axis and, on the vertical axis, the value of the H_∞-norm (between $\xi^{(r)}$ and z_x) that is found after selecting (line search) the μ that yields the smallest H_∞-norm for the specific w_ξ. These figures are slices out of the three dimensional space that w_ξ, μ and $\|H_\infty\|$ span. Note that to obtain these figures with sufficient resolution, LMI (3.25) has to be solved ten-thousands of times.* These figures show that there are well defined minima. The K and L found at these minima ensure that the effects of $\xi^{(r)}$ on z_x in the worst-case direction and worst-case frequency are mitigated in an optimal manner using the regarded type of control scheme. Furthermore, the results for $r = 1, 2, 3$ are given in Table 4.1, the minimizers are all found near $w_\xi = 2.62 \times 10^7$. For the case where model information is used, the case $r = 1$ scores best followed by $r = 2$ and $r = 3$. In the case where no model information is used, the case $r = 2$ scores best followed by $r = 3$ and $r = 1$. All numbers are however relatively close to each other (within 1e-6), hence it is not clear if the differences are due to numerical errors, gaps in the search grid, or that there is an actual link with the value of r.

*The time involved was ± 6 hours using MATLAB 9.10.0.1710957 (R2021a) Update 4, MOSEK version 9.3.1, and YALMIP version 20210331. At a desktop with an Intel 8700K CPU @ 6× 5GHz and 16 GB RAM @ 3000MHz.



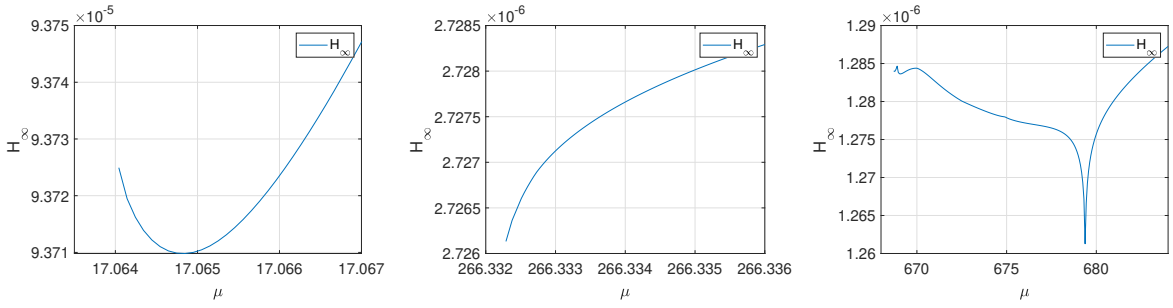
(a) (b)

Figure 4.2: w_ξ plotted against $\|H_\infty\|$ ($\xi^{(r)}$ to z_x) for optimal μ (a) No model. (b) Full model.

$\|H_\infty\|$	Model	No model
$r = 1$	1.0082e-06	1.0110e-06
$r = 2$	1.9450e-06	8.8366e-07
$r = 3$	2.0155e-06	1.2449e-06

Table 4.1: Smallest values for the H_∞-norm found performing a grid search over μ, and w_ξ for, $r = 1, 2, 3$ using full model- and no model -information.

Performing a grid search over a vast amount of values for μ and w_ξ might not always be possible, or necessary. When we look at the graphs of μ plotted against $\|H\|_\infty$ for different values of w_ξ, three cases can be differentiated. These cases are depicted in Figure 4.3. In all cases, the line starts at a certain value for μ, all μ smaller than this specific value yield no feasible controller. For most w_ξ, the plot looks similar to Figure 4.3b, where the smallest feasible μ corresponds to the smallest found $\|H\|_\infty$. For the cases depicted in Figure 4.3a and Figure 4.3c the minimizers are found relatively close to the smallest feasible μ. Furthermore, the smallest $\|H\|_\infty$ for the smallest feasible μ and that for the true minimizer are relatively close to each other. Hence, for practical purposes where it is not necessary to find the optimum but a value near the optimum is sufficient, finding the minimum feasible μ will yield the desired controller. This approach significantly decreases the search space, and thus computation time.



(a) (b) (c)

Figure 4.3: Three cases of μ vs $\|H_\infty\|$. (a) $w_\xi = 1000$. (b) $w_\xi = 1e6$. (c) $w_\xi = 1e7$.

In (4.8) below we show K and L when no model is used, with $w_\xi = 1e7$ and $r = 3$. Since, we modeled the system as three equal chains of integrators the tuning for all three the subsystems is the same. Furthermore, no cross-terms are present.

$$K = \begin{bmatrix} 1983601 & 0 & 0 & 2216 & 0 & 0 \\ 0 & 1983601 & 0 & 0 & 2216 & 0 \\ 0 & 0 & 1983601 & 0 & 0 & 2216 \end{bmatrix}, \quad L = \begin{bmatrix} 148 & 0 & 0 \\ 0 & 148 & 0 \\ 0 & 0 & 148 \\ 3090 & 0 & 0 \\ 0 & 3090 & 0 \\ 0 & 0 & 3090 \\ 36283 & 0 & 0 \\ 0 & 36283 & 0 \\ 0 & 0 & 36283 \\ 264322 & 0 & 0 \\ 0 & 264322 & 0 \\ 0 & 0 & 264322 \\ 952387 & 0 & 0 \\ 0 & 952387 & 0 \\ 0 & 0 & 952387 \end{bmatrix}. \quad (4.8)$$

For the case where full model information is used the following K and L are obtained (4.9). It can be seen that every input-output relation is tuned individually and that there are several cross-terms, even in the higher-order extensions in the GPIO.

$$K = \begin{bmatrix} 1846377 & 0 & 7113 & 2585 & 27 & 0 \\ 0 & 1807753 & 0 & 27 & 2601 & 0 \\ 84268 & 91930 & 1774508 & 95 & 131 & 2579 \end{bmatrix}, \quad L = \begin{bmatrix} 117 & 0 & 0 \\ 31 & 104 & 0 \\ 27 & 39 & 107 \\ 1936 & 117 & 290 \\ 678 & 1664 & 359 \\ 877 & 1036 & 1800 \\ 29937 & 0 & 0 \\ 6286 & 26719 & 0 \\ 12093 & 13614 & 22794 \\ 189205 & 0 & 0 \\ 33699 & 161846 & 0 \\ 94258 & 103478 & 155584 \\ 593617 & 0 & 0 \\ 91137 & 491014 & 0 \\ 338130 & 368805 & 536934 \end{bmatrix}. \quad (4.9)$$

4.3 Ideal simulations

4.3.1 Performance

To make a fair comparison between different controllers, a standard reference trajectory is designed. Since not only the model of the GHS but also an experimental setup of the GHS is studied in Section 5; the reference trajectory should be designed such that it is feasible on the real GHS. The maximum strokes of x_1, x_2, x_3 are approximately 5 mm. To avoid damage on the real setup due to overshoot and unstable closed loops, the references are set to have a maximum stroke of 1 mm. Furthermore, the simulation time of all the simulations/experiments is fixed at 20 seconds. The designed reference trajectory is given by

$$\begin{aligned} x_1^* &= 10^{-3}, & x_2^* &= -10^{-3}, & x_3^* &= 7.5 * 10^{-4} \sin t, \\ x_4^* &= 0, & x_5^* &= 0, & x_6^* &= 7.5 * 10^{-4} \cos t. \end{aligned} \tag{4.10}$$

The reference trajectory (4.10) is depicted in Figure 4.4. The trajectory is designed such that x_3 (the coupling bar) tracks a relative slow (1 rad/s) sinusoidal trajectory. The forces that result from the sinusoidal movement act on masses m_1 and m_2 as a disturbance. Therefore, it is interesting to regulate x_1 and x_2 toward a constant reference to observe the disturbance rejection.

Even though the performance criteria used to tune the ADRC scheme are frequency-domain based techniques, the choice is made to measure performance based on the tracking error in the time-domain. The reason is that it allows to compare the simulations and practical result in a simple but insightful manner.

The selected performance indicator is the integral square error (ISE) defined as

$$ISE = \sum_{i=1}^{3} \int_0^t (x_i(\tau) - x_i^*(\tau))^2 d\tau. \tag{4.11}$$

Note that a lower ISE score means a better performance.

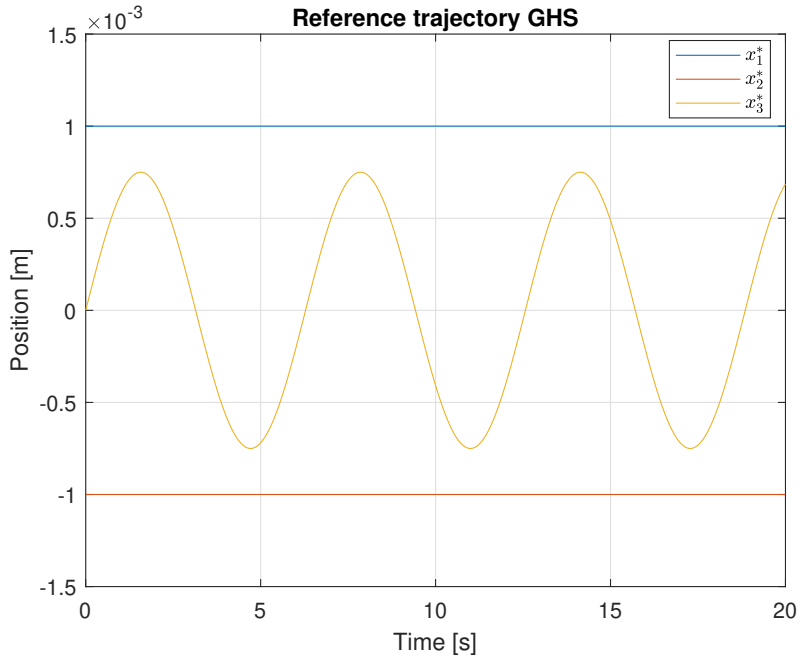


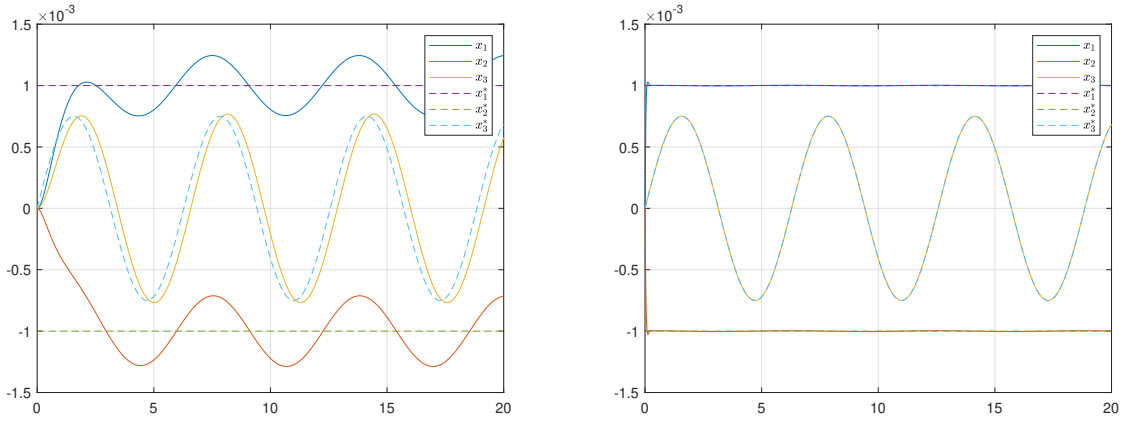
Figure 4.4: Standard reference trajectory GHS.

4.3.2 Simulations

The simulations in this section are made using ADRC schemes tuned using the H_∞ method described in Section 4.2, where in all cases the optimal μ is used.

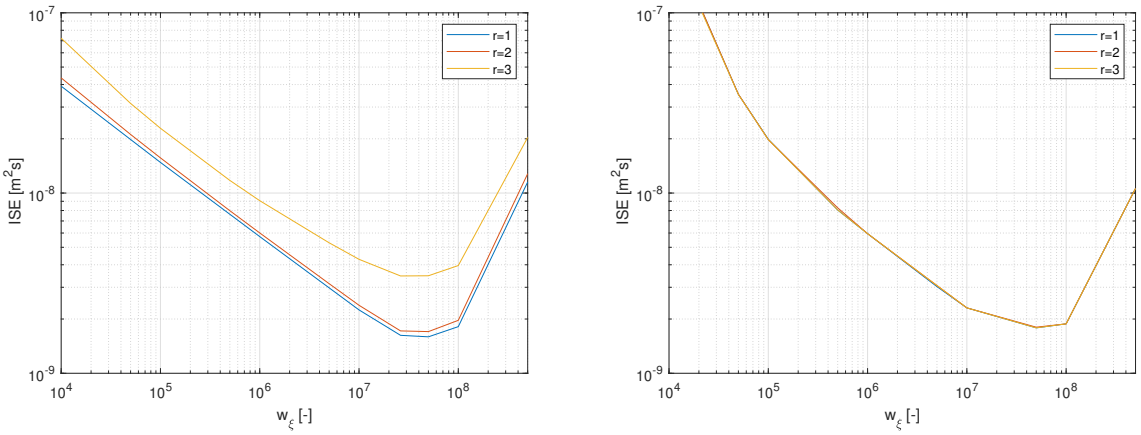
The first simulation is performed using one dynamic extension, $r = 1$, and no tuning parameter, i.e., $w_\xi = 1$. In Figure 4.5a, the reference trajectory and the simulated positions are depicted. Clearly, the tracking performance is not as desired. A brief analysis reveals that the poles of $\mathcal{A}_l$ are placed far into the Left Half-Plane (LHP) and the estimation error is very small. The poles of $\mathcal{A}_k$, however, are marginally in the LHP which explains the relatively slow dynamics with large tracking error. This lack of performance is the reason why tuning parameter w_ξ is necessary. The tuning parameter enables the possibility to place the poles of $\mathcal{A}_k$ further into the LHP.

The result of using $w_\xi = 10,000$ is depicted in Figure 4.5b which is clearly superior with respect to the case $w_\xi = 1$. It does not take much effort to discover that increasing w_ξ results in an improvement of the performance criterion ISE up to a certain level, after which the performance worsens again. Figure 4.6a and Figure 4.6b depict the the effect of increasing w_ξ (sampled from simulations) for $r = 1, 2, 3$, using no model and full model information, respectively. The optimizer can be clearly identified near $w_\xi = 5 \times 10^7$ in both cases. Note that this value is in the same order of magnitude as the $w_\xi = 2.62 \times 10^7$ that yielded the smallest $\|H_\infty\|$. The smallest ISE scores are $1.5932e-09$, and $1.7863e-09$ for the case without $(r = 1)$, and with $(r = 3)$ model, respectively. In the case where full model information is used for all three values of r, the ISE scores obtained for all w_ξ are very close to each other. In the model-free case, an increasing r leads to a higher ISE score. The reason is that the bandwidth of the observer is tuned less high when the value of r is increased. Probably, this problem can be addressed using an additional weighting matrix for the input matrix of the LMI to design the observer.



(a) (b)

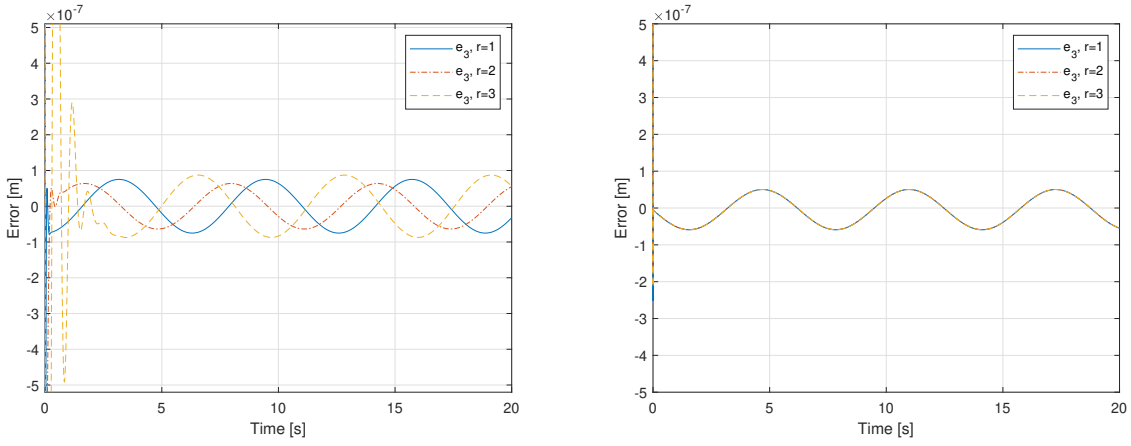
Figure 4.5: Simulation results for different values of w_ξ. (a) $r = 1$, $w_\xi = 1$, $ISE = 3.2231e-06$. (b) $r = 1$, $w_\xi = 10,000$, $ISE = 3.9163e-08$.



(a) (b)

Figure 4.6: Relation between ISE and w_ξ. (a) no model information is used. (b) The full model is used.

The tracking error as a function of time for the third state is depicted in Figure 4.7. The figures show the errors for $r = 1, 2, 3$, in Figure 4.7a no model information is used and in Figure 4.7b, the complete model is used. Observe that in Figure 4.7a the steady-state errors are close to each other, but that it is the transient behavior that impairs the ISE score for $r = 3$. The reason for this is that when r increases the observer dynamics are tuned 'slower' using the synthesis approach. Hence, the observer needs longer to converge. In Figure 4.7b the difference between the three lines is extremely small. Furthermore, the settling time is $t_s < 0.01s$, which is the reason that the transient behavior is not visible on the figure. Using model information has the advantage that the convergence time of the observer can be reduced. It is clear that also the steady-state error is smaller when the model is used. The model-free approach however manages to stay within the same order of magnitude.



(a) (b)

Figure 4.7: Tracking error $e_3 = x_3 - x_3^*$. (a) no model information. (b) full model information.

4.4 Realistic simulations

In order to use the synthesized controllers in a practical setup, the controllers should be tested first in a more realistic simulation. This means adding measurement noise to the outputs, and sampling to input and output. Furthermore, the expected overshoot should be in an acceptable range, and the actuator should not saturate. Finally, the control signal should be (visually inspected) to see if its (frequency)

content is acceptable. For the discretization, a zero-order hold sampling at 1000 Hz is used, which is the same sampling rate as used in the real setup. For the noise, a bandwidth limited white-noise with a power of 10^{-13}W (similar to measured noise in setup) that is uncorrelated for all three channels is used.

4.4.1 Performance

Figure 4.8 shows the relation between tuning parameter w_ξ and the ISE score, for the GHS following the standard trajectory when subject to sensor-noises and sampling. Note that the ISE score is determined using the simulated positions Cx and not the noisy output $Cx + \nu$. The latter case will be discussed in Chapter 5.

Comparing Figure 4.8a with Figure 4.6a the first observation is that the blue line ($r = 1$) starts to inflect more quickly in Figure 4.8a. Furthermore, this line is truncated at $2e7$, the reason is that after this point the system starts to oscillate due to amplified noises resulting in ISE scores that are out of the range of the graph. For the model-free case, the $r = 1$ line has the lowest ISE score for $w_\xi < 4e5$ after which it inflects upwards and is overtaken by $r = 3$. The reason for this inflection is that the observer reconstructs measurement noises, which subsequently are amplified by the controller gains. The case where $r = 2$ can be tuned with higher values of w_ξ before the inflection happens. The reason for this is that the observer in the $r = 2$ case is tuned less aggressively and is hence less capable in reconstructing these high frequency measurement noises. This is positive since the estimated noises are smoothed versions of the original noises.

For the case with model, up to $w_\xi \approx 1e5$, the three lines $r = 1, 2, 3$ stay near each other similar to what is observed in Figure 4.6b. After $w_\xi \approx 1e5$, the line $r = 1$ starts to inflect again due to measurement noises that are reconstructed and then amplified.

The lowest ISE score is measured for $r = 3$ with $w_\xi = 3e7$ yielding, $ISE = 5.0043e - 09$. Figure 4.8b follows a similar trajectory as Figure 4.6b up to $w_\xi \approx 1e5$ after which it start to deviate due to the effects of noise. The minimum ISE score found for the case with model is $3.6935e - 09$, which is found at $w_\xi = 2e7$.
The smallest obtained ISE scores for $r = 1, 2, 3$ are given for the four test-cases in Table 4.2. Note that all scores are in the same order of magnitude. clearly, the ideal case performs better than the non ideal case in all instances. Remarkable is that the model-free case slightly outperforms the case with model in the ideal world simulations, except for $r = 3$. However, in the more realistic simulations the case with model performs better for all r.

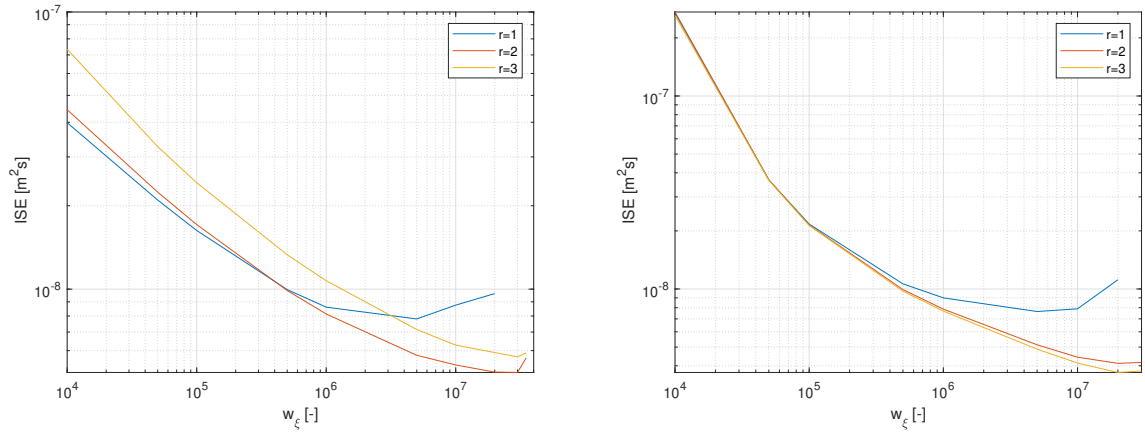


Figure 4.8: Relation w_ξ and ISE when subject to noise and sampling. (a) no model. (b) Full model.

4.4.2 Control signals

In this section, the control signals and their components are analyzed. Figure 4.9a and Figure 4.9b depict the control signals for a synthesized ADRC using $w_\xi = 1e4$ and $w_\xi = 1e5$ respectively and $r = 2$ in both cases. The control signals follow from simulations of the GHS tracking the standard trajectory while subject to measurement noise and sampling. It is clear that measurement noise is present excessively in the control signal. In Figure 4.9b the effect is more severe than in Figure 4.9a. For both cases the same observer is used. Hence, what is observed is the amplification of measurement noise reconstructed by the GPIO that is amplified by the elements in K.

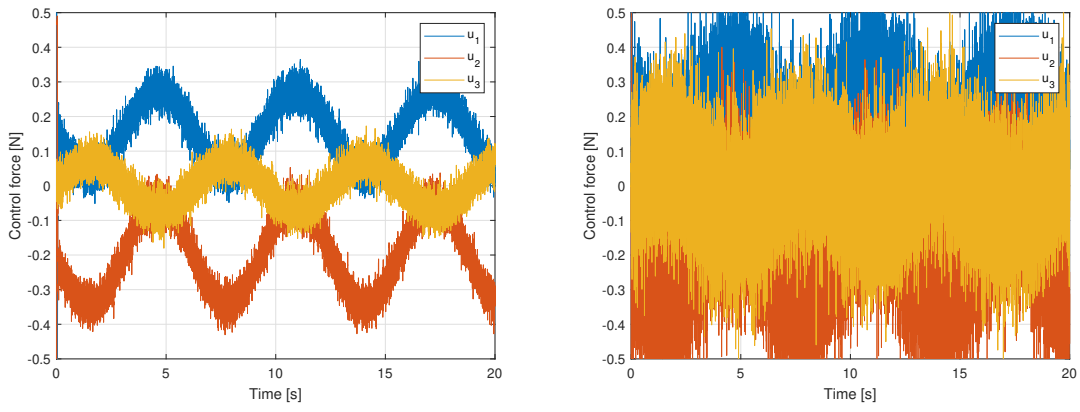
Let's analyze the components of the control signal, $u = -R^{-1}(K(\hat{x} - x^*) + \hat{\xi}_0)$, that bring measurement noise into the signal. The first component is $\hat{x}$, this estimated state can be divided in estimated positions $\hat{x}_{1,2,3}$ and velocities $\hat{x}_{4,5,6}$. Figure 4.10a and Figure 4.10b depict the (estimated) positions and the (estimated) velocities respectively. It can be seen that in the estimates of the position some measurement noise is reconstructed, the amplitude however is relatively low. On the other hand in the velocity plot it can be seen that the (differentiated) measurement noise is present excessively in the estimate. Furthermore the actual velocity also shows rapid changes in velocity that are induced by the noisy control signal. This observation identifies the need for well conditioned measurement signals, and (frequency-dependent) weighting matrices in the tuning process. These weighting matrices should assure that the observer does not reconstruct or at least filters the noise in the first place. Furthermore, the magnitude of the elements in K should be limited to not amplify the noises that are reconstructed any further. Especially the elements multiplied with the estimated velocities.

The final part of the control signal is the estimate (the actual ξ_0 is also depicted) of the lumped disturbance $\hat{\xi}_0$ which is depicted in Figure 4.11a, for completeness also its derivative $\hat{\xi}_1$ is depicted in Figure 4.11b. It is seen that again the noise is reconstructed by the observer. It is however important to realize that the $\hat{\xi}_0$ term is not amplified by a (high-)gain in the control.

A practical method to reduce the negative effects of noises in the output of the observer is placing a low-pass filter between the observer and controller. Placing such a filter, does change the definition used for ADRC in this thesis and is therefore not considered.

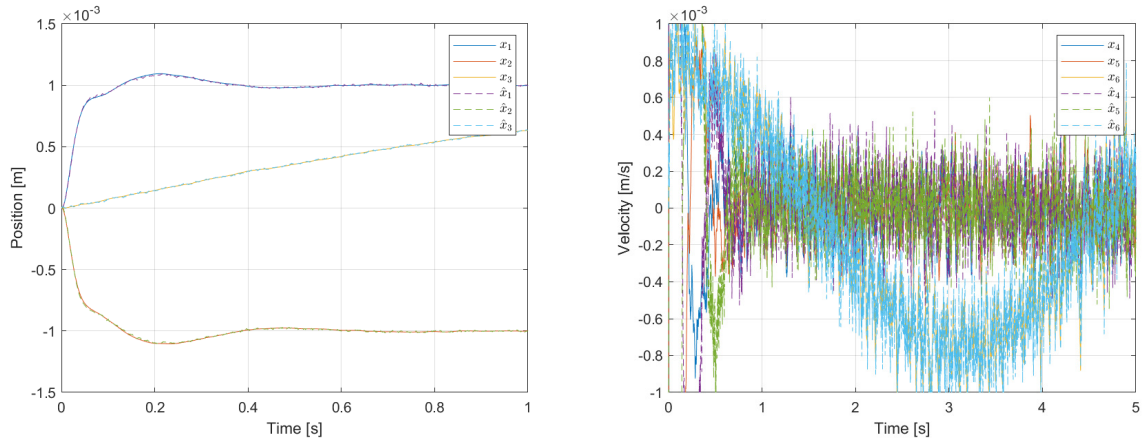
ISE	ideal, no model	ideal, model	not ideal, no model	not ideal, model
r_1	1.5932e-09	1.7865e-09	7.8057e-09	7.6425e-09
r_2	1.7032e-09	1.8021e-09	5.0043e-09	4.1187e-09
r_3	3.4674e-09	1.7863e-09	5.6946e-09	3.6935e-09

Table 4.2: Smallest ISE scores for $r = 1, 2, 3$ in ideal world and more realistic simulations using no model and full model information.



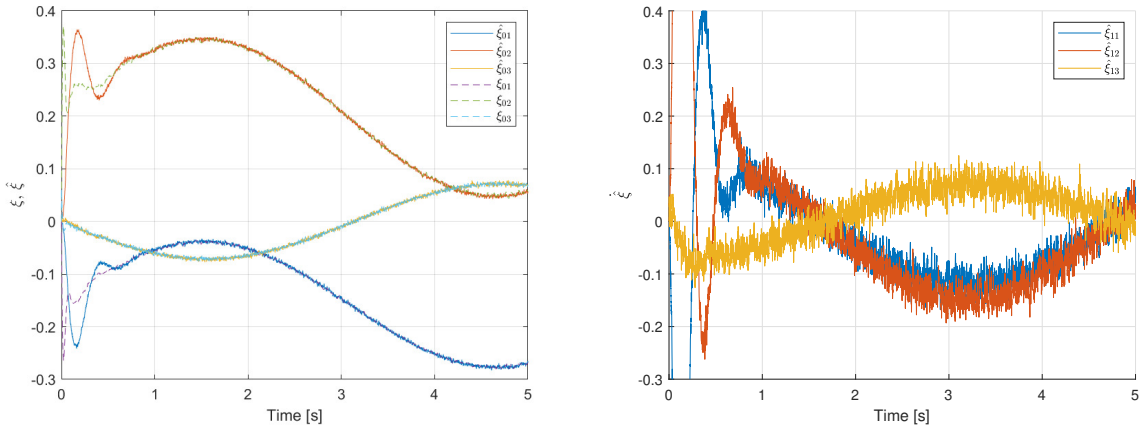
(a) (b)

Figure 4.9: Control signals. (a) $w_\xi = 1e4$. (b) $w_\xi = 1e5$.



(a) (b)

Figure 4.10: (a) Positions and its estimates. (b) Velocities and its estimates.



(a) (b)

Figure 4.11: (a) Estimated and actual lumped disturbances. (b) First derivative estimated lumped disturbances.

4.4.3 Disturbance estimation

In this section, the ability to reconstruct an external disturbance using the synthesized ADRC scheme is verified.

Consider, the forced Van der Pol oscillator

$$f(t) = \ddot{z} - \mu(1 - z^2)\dot{z} + z - a\sin(\omega t), \tag{4.12}$$

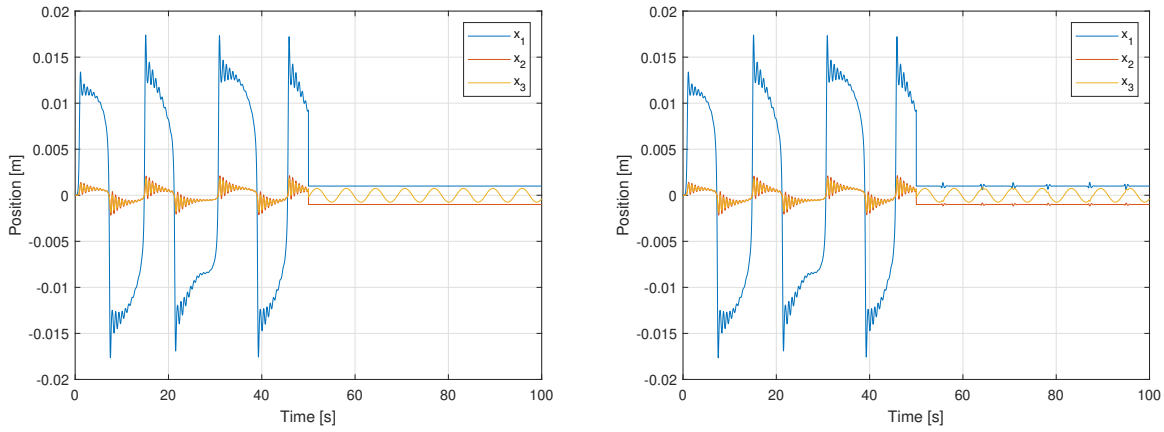
with $\mu = 8.53$, $a = 1.2$, $\omega = 2\pi/10$ and initial conditions $z(0) = \dot{z}(0) = 0$.

This (chaotic) oscillator is added as $w = [f(t)\ 0\ 0]^\top$ to the model of the GHS as matching disturbance. The simulation (with the standard trajectory) is run for 100 seconds, in the first 50 seconds the controller is disabled, in the next 50 seconds the controller is active. The simulation results are shown in Figure 4.12, in both cases a synthesized controller with $w_\xi = 5e7$. However, Figure 4.12a is tuned using $r = 1$ and Figure 4.12b with $r = 3$. It can be seen that in both cases as soon as the controller is started

the desired reference trajectory is tracked. However in the $r = 3$ case the disturbance rejection is clearly less, the reason for this is that the observer is tuned less fast in the $r = 3$ case. Note that in both cases full model information is used. The cases without model information look almost exactly the same, therefore these case are not shown.

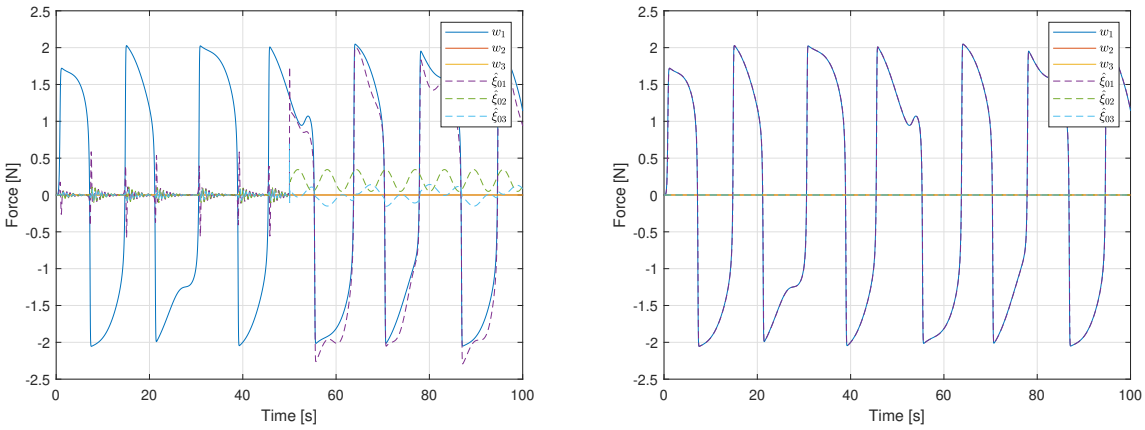
Now, let us investigate what the effects on the disturbance reconstruction are using the full model and no model. In Figure 4.13, the actual external disturbance signal w and the estimated lumped disturbance $\hat{\xi}_0$ are depicted. In Figure 4.13a, no model information is used and in Figure 4.13b the full model is used. In the case where no model is used, we see that before the controller is started the disturbance signal w is not estimated correctly. When the controller is started, we can recognize the forced Van der Pol oscillator w_1, however, the effects of the unmodeled dynamics are added to the signal. Furthermore, $\hat{\xi}_{02,03}$ show the effects of unmodeled internal dynamics and external disturbance w_1 on $\ddot{x}_{2,3}$. For the case where the model is used, the disturbance w_1 is estimated near to perfect, and, as expected, the estimates of $w_{1,2}$ are near zero.

The estimated disturbance for the case $r = 3$ is depicted in Figure 4.14, where Figure 4.14a, and Figure 4.14b show the case without and with model, respectively. In both cases, it can be seen that when the disturbance w_1 undergoes a fast change, the observer needs to converge again. This goes paired with relatively large overshoot and does take time. This overshoot and convergence time are the main reasons for the tracking errors visible in Figure 4.12b.



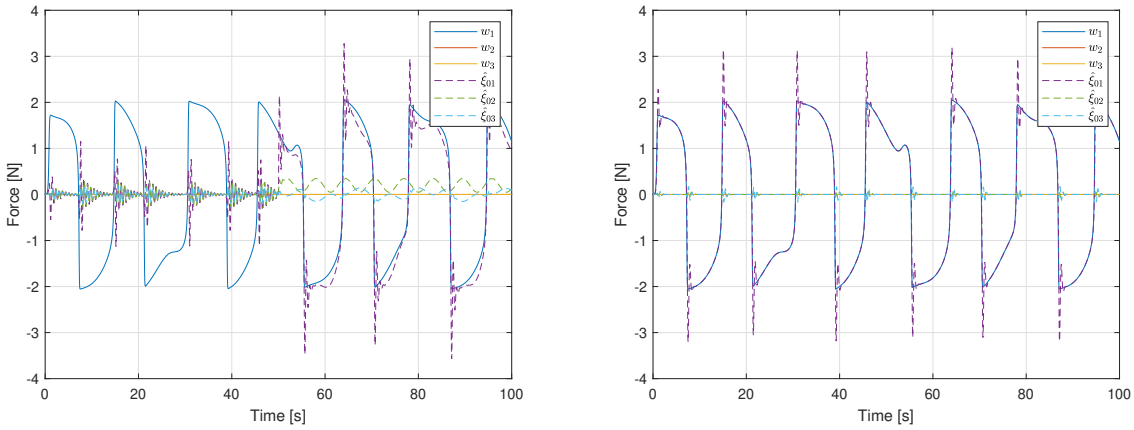
(a) (b)

Figure 4.12: Tracking with forced Van der Pol oscillator as disturbance on $\ddot{x}_1$. (a) Full model $r = 1$, $w_\xi = 5e7$. (b) Full model $r = 3$, $w_\xi = 5e7$.



(a) (b)

Figure 4.13: Estimation forced Van der Pol oscillator as disturbance on $\ddot{x}_1$. (a) no model $r = 1$, $w_\xi = 5e7$. (b) Full model $r = 1$, $w_\xi = 5e7$.



(a) (b)

Figure 4.14: Estimation forced Van der Pol oscillator as disturbance on $\ddot{x}_1$. (a) no model $r = 3$, $w_\xi = 5e7$. (b) Full model $r = 3$, $w_\xi = 5e7$.

4.5 Discussion

A practical method to synthesize a GPIO (L) with r extensions in an optimal manner is described and demonstrated. Next, a method to synthesize a feedback controller (K) is developed and demonstrated. Furthermore, a method to find the (optimal) combination of the obtained matrix L and matrices K resulting in the lowest H_∞ gain between $\xi^{(r)}$ and a performance output z_x is developed and demonstrated.

A MATLAB program is developed that has as input any (MIMO) exact feedback linearizable system in the normal form, and the desired number of dynamic extensions (per input-output relation) r. And as output the (sub)optimal matrices K, L and the observer and controller that together form the ADRC in state-space format.

It has been demonstrated that it is possible to add known model information to the GPIO, the LMI to determine K, and the LMI to determine L. Adding model information is concluded to be beneficial for the ISE score. The rise time is decreased vastly and the steady state error decreased moderately when compared with the model-free method.

The effect of adding more dynamic extensions r to the GPIO, shows to be negative in the synthesis approach using no model information and under ideal simulation conditions (no noise, no sampling). The higher the number of dynamic extensions the slower the observer is tuned. This slower tuning leads to longer convergence time which results in a longer rise time. However in the case where the full model information is used there is only a marginal difference between the different values for r. In the more realistic simulations where measurement noise and sampling are applied the results are different. Compared to the ideal world simulations the overall ISE scores have decreased. In the model-free case the best ISE score is obtained using two extensions and in the full model case three extension give the best score. Adding a weighting matrix to the disturbance input of the observer is proposed to allow more aggressive tuning in the model- free case with higher-order extensions.

It is shown that the (synthesized) ADRC scheme is sensitive to measurement noises. The measurement noise is (partially) reconstructed by the observer and subsequently amplified by the feedback controller. In the tuning of both observer and controller the effects of measurement noises should be taken into account. Therefore the usage of frequency-dependent weighting matrices is proposed.

For the disturbance reconstruction the following conclusions can be drawn. We have demonstrated that in both cases with, and without model similar tracking performance can be achieved when a matching external disturbance is applied to the system. The effects of higher-order extensions are in this case negative for disturbance estimation and therefore for tracking performance when the synthesis approach is used. Furthermore, it is shown that for reconstruction of external disturbances the model parameters of the system (in normal form) need to be known.

Chapter 5

Experimental results

In this chapter, the results of the practical experiments as described in Section 1.4.4 are reported. First, the encountered practical difficulties and the proposed solutions for them are described in Section 5.2. Next, an empirical and semi-empirical method to tune ADRC schemes is explained and tested in Section 5.3 and 5.4 respectively. Subsequently, the synthesis approach as described in Section 4.2 is tested in Section 5.5. Finally, the conclusions are presented in Section 5.6.

5.1 Generalized Huygens Setup

A picture of the GHS is shown in Figure 5.1, the blue (m_1) and the red (m_2) parts in the picture are the two oscillators where position x_1, x_2 are measured. These oscillators are connected by a coupling bar (m_3) in the middle. State x_3 is the position of this bar. The positions are measured using three linear variable differential transformers as sensors. The sensors are calibrated such that 1 V corresponds to a displacement of 5 mm. All three masses are actuated by voice coil actuators. For the actuators, we have motor constants 20.9218, 23.2465 and 1.2589 $\text{ms}^{-2}\text{V}^{-1}$ for actuator 1,2 and 3, respectively. Note that these (actuator) gains are not required to be known in the ADRC scheme, they are used however in the experiments. More information about the mechanical design of the GHS can be found in [33]. Additional information about the sensors, actuators and data acquisition system is given in [34] which is also the source of the sensor and actuator gains.

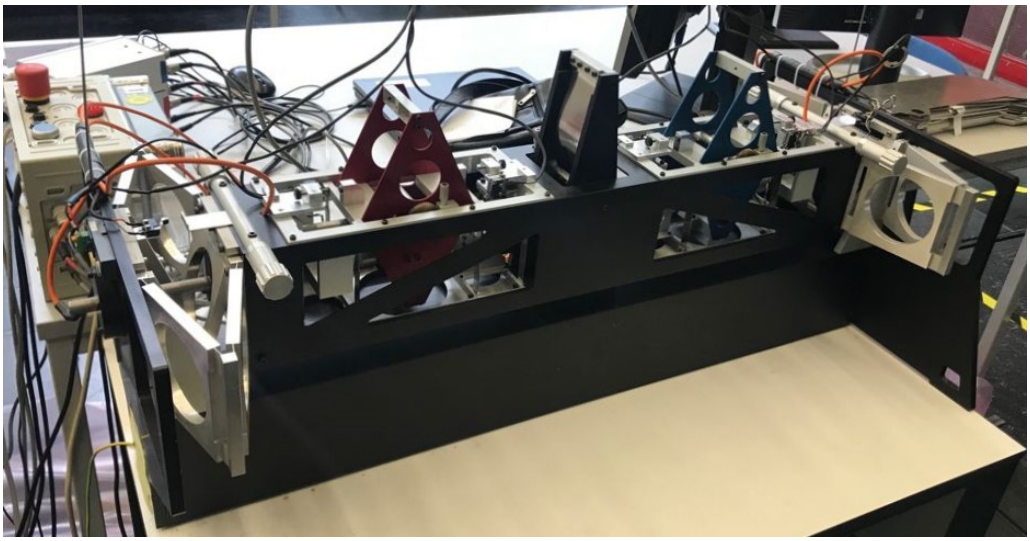


Figure 5.1: Photograph of the Generalized Huygens Setup at the DCT lab at the Eindhoven University of Technology.

5.2 Practical difficulties

The first attempt to control the GHS is done using an empirical tuning method. Without any extra input or output filters. Conservative ADRC and PID controllers are used to test the three second-order SISO systems one by one. Attempting to control towards a constant reference leads to (heavy) vibrations in the leaf springs which are directly connected to the coupling bar (x_3) (it can take seconds to minutes before this happens). Probably, these vibrations are caused by (partially) uncontrollable higher-order dynamics (which are not included in the model).

To analyze the problem, a Frequency Response Function (FRF) is made $u_3 \to y_3$ (since this is the subsystem connected to the leaf springs). The magnitude plot of the FRF is depicted in Figure 5.2. We can clearly recognize the second order dynamics with one fairly damped resonance peak followed by a -4 slope. Then there is a small peak, of unknown origin. Probably it is due to actuator/sensor dynamics or coupling effects. Then there is the area between 36 and 38 Hz. Note the very sharp un-damped peaks. The FRF was made, exiting all 3 actuators with uncorrelated bandwidth limited white noise with relatively low power. Exciting at higher powers will make these peaks grow quickly and possibly damage the setup. Then, there is the peak at 50 Hz which is a benign measurement noise.

To solve the mentioned difficulties, a second-order band-stop filter is placed between 36 and 38 Hz at the input side (for all three actuators), so the controller cannot excite these frequencies. This filter solves the problem with the resonances. Furthermore, a second-order notch filter is placed between $49 - 51$ Hz to attenuate the 50 Hz noise, and a second-order low-pass filter at 25 Hz is placed at all outputs to attenuate high-frequency measurement noises. Experimentation shows that a low-pass filter at the input side reduces audible sound in the actuators, but for some controllers leads to drastic performance degradation. Therefore, such filter is not used during experimentation.

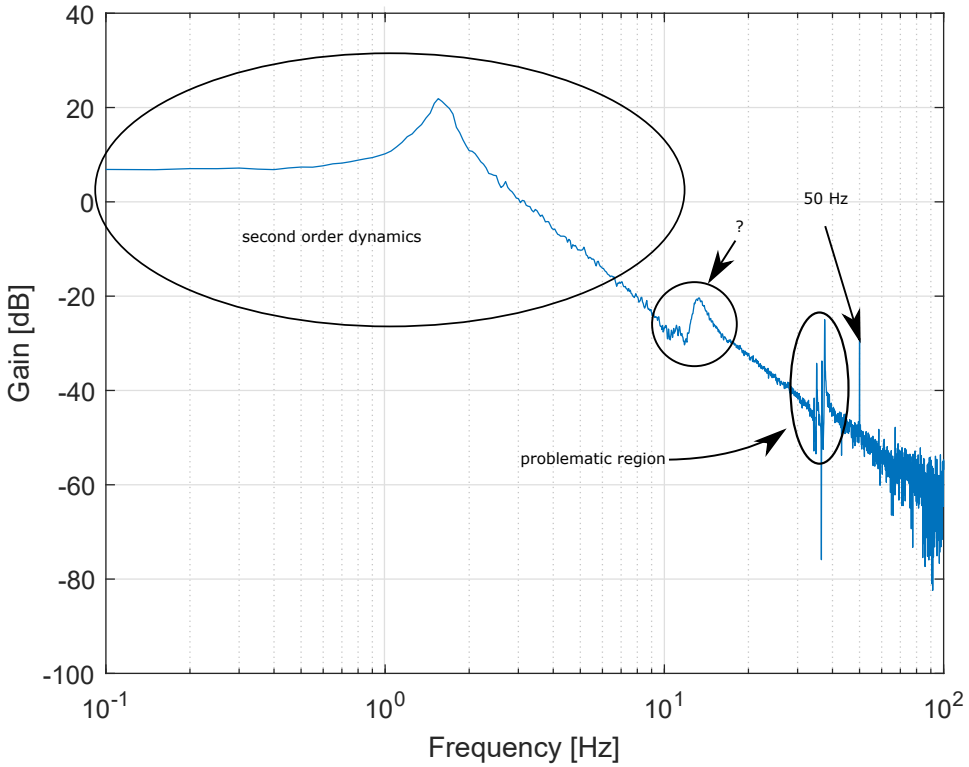


Figure 5.2: FRF magnitude, $\frac{y_3}{u_3}$.

5.3 Empirical tuning

First the ADRC scheme is tuned using an empirical method.
For the observer L, is selected such that the eigenvalues of $\mathcal{A}_l$ equal the roots of characteristic polynomial

$$(s^2 + 2\zeta_o\omega_o s + \omega_o^2)(s + p)^r, \tag{5.1}$$

with ω_o the natural frequency of the observer [rad/s], ζ_o a damping constant $[-]$, p a constant, all in $\in \mathbb{R}_+$, and s is the Laplace operator. Using this method, for each second-order input-output relation three parameters have to be determined.
For the controller side the following characteristic polynomial is matched:

$$(s^2 + 2\zeta_c\omega_c s + \omega_c^2), \tag{5.2}$$

resulting in another 2 parameters to select per input-output relation.

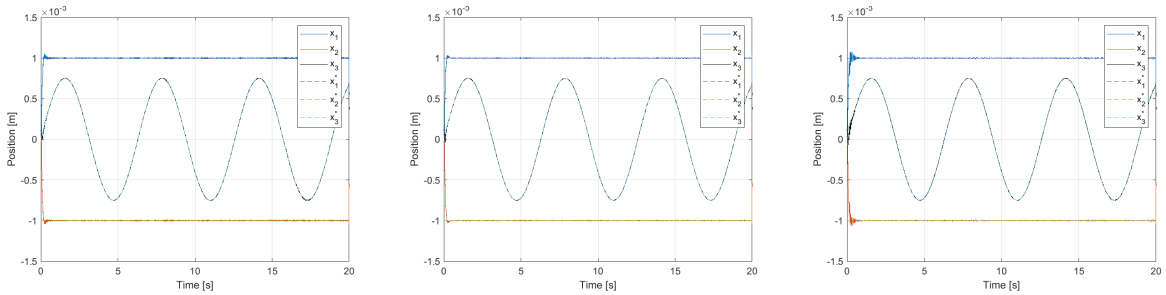
This means that in total 15 design parameters have to be selected to tune the ADRC scheme for the GHS. To reduce the number of parameters to a tractable amount, the decision is made to tune all the input-output relations equally and set $p = \omega_0$ and $\zeta_o = \zeta_c$. Resulting in only 3 free parameters to determine.

For $r = 1, 2, 3$, first ω_o is selected such that the reconstructed states are not dominated by noise. Then in a systematic manner a selection of combinations of $\zeta_o = \zeta_c$ and ω_c are tested. Controllers that result in a relatively short rise time, good tracking, and are not dominated by noise are selected. These controllers are indeed the ones that score the lowest ISE values. The best obtained observer/controller parameters and the resulting ISE scores are given in Table 5.1. Observe that only the ω_o differs. For higher values of r, the observer becomes more sensitive to measurement noise and has to be tuned 'slower' ,i.e., the observer bandwidth ω_o has to be reduced. This slower observer results in the (slightly) higher ISE score for increasing values of r.

The results of the experiments with the three ADRC implementations with parameters as in Table 5.1 are depicted in Figure 5.3 and Figure 5.4 (tracking error). In these figures, it can be seen that all three controllers give a reasonable tracking performance, the difference is that the transient behavior changes in a negative manner for a higher-order extension.

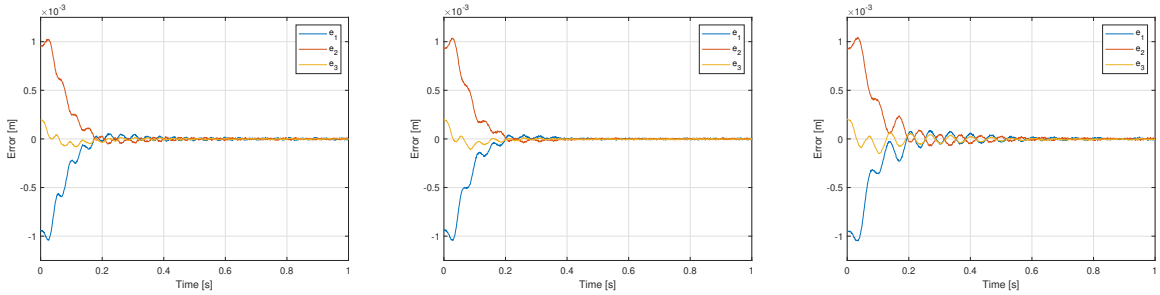
	ω_o	ω_c	ζ	ISE
$r = 1$	200	20	$\frac{1}{\sqrt{2}}$	1.1810e-07
$r = 2$	150	20	$\frac{1}{\sqrt{2}}$	1.2147e-07
$r = 3$	100	20	$\frac{1}{\sqrt{2}}$	1.3446e-07

Table 5.1: Tuning parameters empirical observer, controller, and resulting ISE scores.



(a) (b) (c)

Figure 5.3: Results experiment empirical controller. (a) $r = 1$. (b) $r = 2$. (c) $r = 3$.



(a) (b) (c)

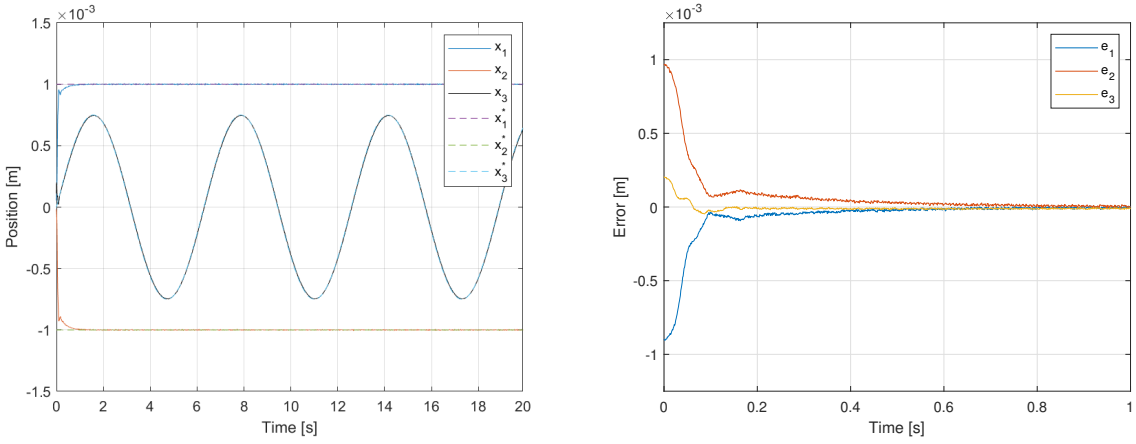
Figure 5.4: Tracking errors experiment empirical controller. (a) $r = 1$. (b) $r = 2$. (c) $r = 3$.

5.4 Semi-empirical tuning

A well-known tuning method for ADRC is the bandwidth parametrization method described in [12], [35] using only one tuning parameter t_s^* which stands for the desired settling time. The following procedure is followed to tune a second-order system with one dynamic extension.

1. choose desired settling time t_s^*
2. let $\omega_c = 10/t_s^*$
3. $k_p = \omega_c^2$, $k_d = 2\omega_c$
4. $\omega_o = 4\omega_c$
5. $l_1 = 3\omega_o$, $l_2 = 3\omega_o^2$, $l_3 = 4l_2$

Where, ω_c, ω_o denote the controller and observer bandwidth respectively. Again all input-output relations are tuned equally. The best ISE score observed was $6.6108e-08$ for $t_s^* = 0.02$ *. The resulting graph is depicted in Figure 5.5. Note that the result for this tuning method is better in the sense of ISE than the empirical method tested in Section 5.3. Since the design space of the semi-empirical method is a subspace of the design space of the empirical method, a similar or even better performance can be found using the empirical method. The fact that no equal or better performing controllers were found confirm the difficulty in tuning controllers in these empirical methods, especially for inexperienced practitioners.



(a) (b)

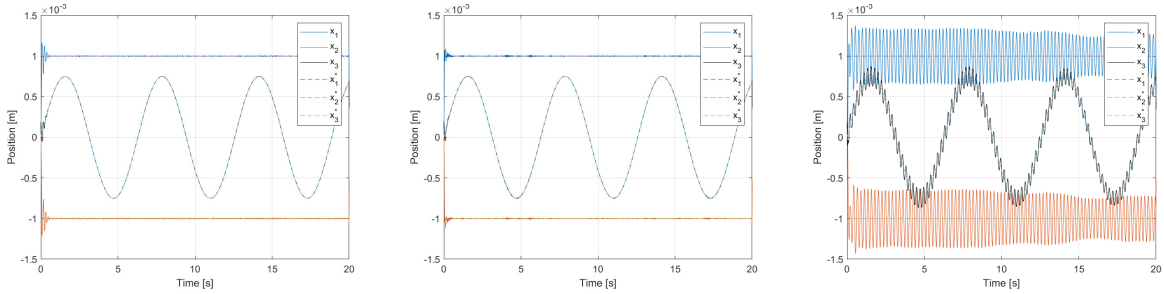
Figure 5.5: Result semi-empirically tuning. (a) positions and references. (b) tracking errors.

*Other ratios than $\frac{\omega_o}{\omega_k} = 4$ and $\frac{l_3}{l_2} = 4$ are evaluated but do in this case not yield better results.

5.5 Synthesis

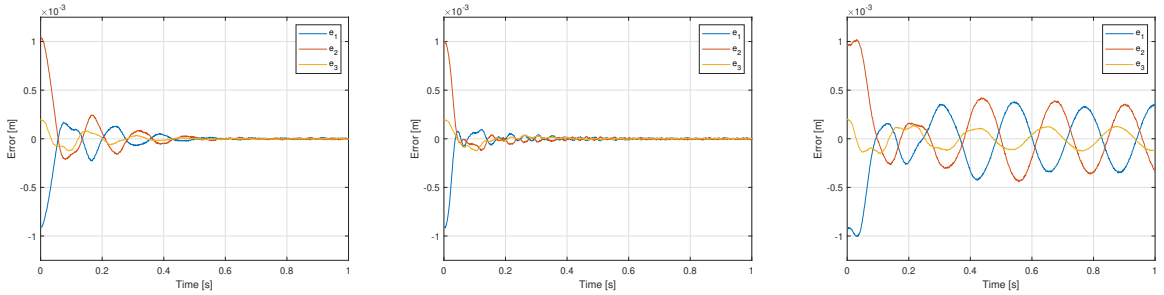
In this section, a selection of controllers tuned using the synthesis approach is tested on the real GHS.

The test results of the controllers that resulted in the lowest ISE score for $r = 1, 2, 3$, respectively are depicted in Figure 5.6 and Figure 5.7 (tracking error). It can be seen that both cases $r = 1, 2$ have a good tracking performance and quick rise time, with a little bit more overshoot for the case $r = 1$. For $r = 3$, however, no synthesized controller is found that tracks the reference in an acceptable manner. The reason for this is unknown; in the simulations, this problem is not present. The conjecture is that the problem arises due to actuator dynamics and input delays.



(a) (b) (c)

Figure 5.6: Results experiment synthesized controller for: (a) $r = 1$. (b) $r = 2$. (c) $r = 3$.



(a) (b) (c)

Figure 5.7: Tracking error experiment synthesized controller for: (a) $r = 1$. (b) $r = 2$. (c) $r = 3$.

In Figure 5.8, the ISE score versus the tuning parameter w_ξ is depicted for the simulation using $y = C_e x + \nu$ (dashed line) and the practical results (solid line). Note that the solid yellow line $r = 1$ stops at $w_\xi = 10^4$. The reason for this is that the system becomes unstable beyond this point. It can be seen that the theoretical and real values match reasonably until $w_x \approx 1000$ after which the real system starts to drop performance. The reason for this performance decrease can most probably be ascribed to actuator dynamics and input delays.

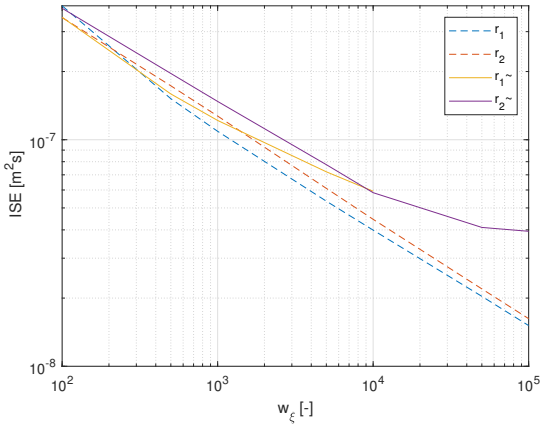


Figure 5.8: Relation between w_ξ and ISE score, theoretical ($r_{1,2}$) and and real ($\tilde{r}_{1,2}$).

In Table 5.2, an overview of the ISE scores for the different controllers tested in practice is given. Clearly it can be seen that the H_∞ synthesis using only tuning parameter w_ξ is able to outperform the (semi-)empirical method.

To give the reader a more physical interpretation of the performance, the error signal $e_3 = x_3 - x_3^*$ for $r = 2$ and $w_\xi = 1e5$ is depicted in Figure 5.9. The RMS value of this error is just below $2\mu m$.

Practical results	ISE
Empirical $r = 1$	1.1810e-07
Empirical $r = 2$	1.2147e-07
Empirical $r = 3$	1.3446e-07
Semi-empirical $r = 1$	6.6108e-08
Synthesis $r = 1$, w_ξ= 100	3.4710e-07
Synthesis $r = 1$, w_ξ= 500	1.5925e-07
Synthesis $r = 1$, w_ξ= 1,000	1.2165e-07
Synthesis $r = 1$, w_ξ= 5,000	7.2177e-08
Synthesis $r = 1$, w_ξ= 10,000	5.9245e-08
Synthesis $r = 2$, w_ξ= 100	3.8042e-07
Synthesis $r = 2$, w_ξ= 500	1.9613e-07
Synthesis $r = 2$, w_ξ= 1,000	1.4758e-07
Synthesis $r = 2$, w_ξ= 5,000	7.7621e-08
Synthesis $r = 2$, w_ξ= 10,000	5.8378e-08
Synthesis $r = 2$, w_ξ= 50,000	4.1015e-08
Synthesis $r = 2$, w_ξ= 100,000	3.9435e-08

Table 5.2: Overview test results.

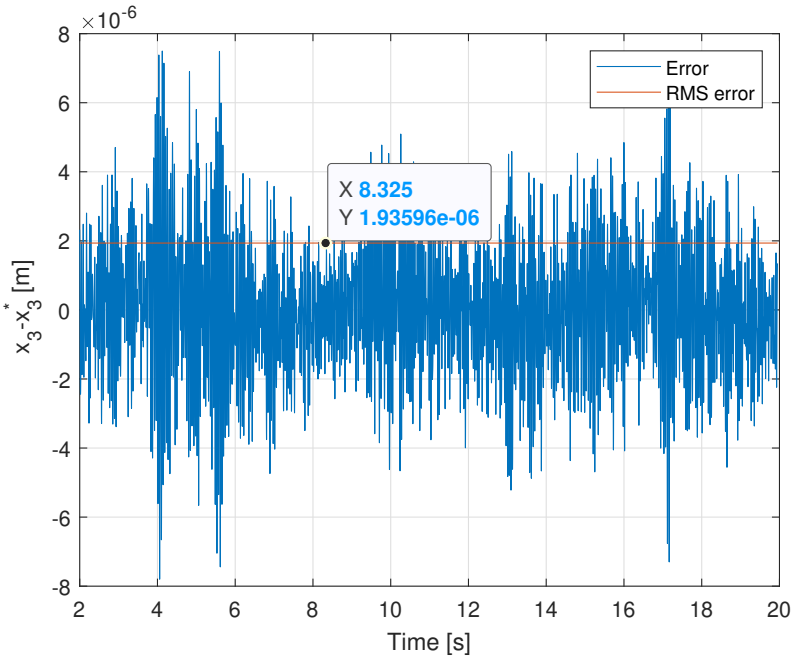


Figure 5.9: Error signal e_3.

5.6 Discussion

The empirical tuning method is tested on the GHS for $r = 1, 2, 3$. For all three cases reasonable performance is achieved. For higher-order extensions, i.e., $r = 2, 3$, the bandwidth of the observer has to be reduced to avoid issues due to measurement noise. This reduction in bandwidth results in increased rise times. Using the empirical tuning method it is relatively easy to find stabilizing controllers that give reasonable performance, however, it is difficult to fine-tune the controllers to reach optimal closed loop behavior.

The semi-empirical approach is relatively easy to use since it requires only one tuning parameter. The reported ISE score is in the tested case better than the scores found using the empirical method. The method gives however not much freedom in controller design.

The synthesis method is tested for $r = 1, 2, 3$, it stands out that the case $r = 3$ does not yield tracking controllers with acceptable tracking performance. Comparing the ISE scores of the synthesized controllers for the performed simulations and real world experiments it becomes clear that the best scoring controllers in the simulations do not work in practice. The reasons for this are most likely actuator/sensor dynamics and input/output delays. However, up to a value of $w_\xi \approx 1000$ the simulations and practical results are comparable.

Tuning the observer and controller that form the ADRC scheme using the derived H_∞ synthesis method is relatively easy compared to the other methods. In practical situations the only parameter that has to be selected is w_ξ (for a given r). In practice, the w_ξ that yields the lowest ISE score is the maximum w_ξ that results in a stabilizing controller.

In practice, the H_∞ syntheses method using no model information outperforms the empirical and semi-empirical tuning methods in the sense of ISE. Furthermore, tuning the synthesis method is more time efficient once it is implemented compared to the other mentioned methods.

Chapter 6

Conclusions, discussion, and future work

6.1 Conclusions

The main goal of this thesis is to develop a framework to tune the ADRC scheme for a general class of feedback linearizable (mechanical) systems. With the ADRC scheme in a observer/controller representation with gain matrices L and K to be determined explicitly. The focus is on tracking performance and disturbance rejection. The method must be "practically" model-free but allow for using known (linear) model information. Multiple tools to reach this goal are developed. The used performance specifications are the H_∞ and H_2-norms and the ISS gain. These specifications are represented in their LMI formulations. For these LMI based specifications, it has become clear that a co-design of gain matrices K and L is intractable. Therefore, a two-step approach is developed. For all the performance specifications, LMIs have been developed to tune a Luenberger type of observer with r dynamic extensions in an optimal manner. For the controller part, LMIs have been developed for the three performance criteria to tune gain matrix K. However, for the controller part the tuning is sub-optimal since the optimization problem had to be relaxed to make it tractable. Furthermore, two LMIs have been derived that can be added as constraints in the optimization problem to minimize the effects of measurement noise on the observer, using the H_2 and H_∞-norms.

The usability for practical output tracking of the LMIs has been investigated for the two H_∞ LMIs (observer and controller). The developed method depends on tuning parameter w_ξ that is present in the performance output of the LMI for the observer and in the disturbance input of the LMI for the controller. It is explained how to optimally determine gain L and find a set of sub-optimal gains K. Subsequently, it is described how to select the gain matrix K that results in the best closed loop performance in combination with the given L. Moreover, a MATLAB script is developed to perform the tuning in a fully automated manner when provided with a (MIMO) system in the normal form.

The effects of using model information in the tuning process and in the observer is compared to the case where no model information is used. This comparison has been made in simulations only. It has been demonstrated that the developed H_∞ syntheses method allows to include model information. It is observed that using model information results in significantly shorter convergence times in the observer, and shorter rise times in the system dynamics. Furthermore, the steady-state tracking error is decreased. However, the steady-state tracking errors for the model-free approach are in the same order of magnitude as those in the case where full model information is used.

The effect of using more state extensions r in the ADRC scheme, when tuned using the H_∞ synthesis method, is investigated for four cases. When full model information is used the effect of adding higher-order extensions is positive (better ISE score) with, and without measurement noise and sampling. When no model information is used the effect of more dynamic extensions is negative when no measurement noise and sampling are present. In the case where noise and sampling are included but no model, the

results depend on the used value of w_ξ. In the model-free case, the usage for a weighting matrix at the input of the observer is proposed to allow the tuning for more/less aggressive observers when needed.

The effect of measurement noises and discretization on the synthesized ADRC scheme are analyzed in simulations. The effect of noises and sampling on the ISE score for different values of r is already described in the last paragraph. Here the observed effects on the control signal are described. It has become clear that the synthesized ADRC scheme is sensitive to measurement noises (as it is nature a high-gain scheme). In the first place because the observer reconstructs these noises in the estimated system states and estimated disturbance. Subsequently, these reconstructed noises are amplified by gain matrix K (for the estimated system states) and together with the estimated disturbance (which also contains noises) combined in the control signal. These (amplified) noises in the control signal are not desired. Therefore, L should be designed such that high-frequency measurement noises are not reconstructed but filtered. Furthermore, K should be tuned such that the reconstructed noises are not amplified too much. The proposed solution are frequency-dependent weighting matrices.

It is investigated how to reconstruct an external disturbance signal while the system is in closed loop with an ADRC scheme. This is done in simulations (no noise, no sampling). For an ADRC scheme without model information used, it is demonstrated that in open loop no acceptable reconstruction of the external disturbance is achieved, and in closed loop the external disturbance with the effects of unknown dynamics added to it are reconstructed. In the case with full model information the external disturbance is reconstructed with high precision in open and closed loop.

The tracking performance of the synthesized ADRC scheme and the empirical tuning methods is compared in real world experiments on the GHS. The performance is quantified using the ISE score. For both the empirical and the synthesis tuning methods additional input and output filters are necessary to control the plant. It is shown that the synthesis method works on a real world setup. Furthermore, it has become clear that the synthesis method can outperform the empirical tuning methods, in the sense of the ISE score and in the sense of the time needed to tune the scheme.

6.2 Discussion

In this section, the conclusions are interpreted and the limitations are pointed out.

The given H_∞ synthesis method only makes use of the tuning parameter w_ξ that scales/weights the elements related to ξ_0 in the performance output and disturbance input of the observer and controller, respectively. The option to scale/weight elements regarding state x is not regarded for these inputs/outputs. Furthermore, no weighting is applied to the disturbance input on the observer side, and to the performance output on the controller side. Hence, the result are only optimal in the sense of the demonstrated tuning method.

For the determined H_2 and ISS LMIs, and the H_2/H_∞ LMIs to protect against measurement noise no practical method to use them is developed. For the H_2-norm, the multiplicative property does not hold; however, it can be used in a similar method as described for the H_∞ method (with proper weighting matrices) as described in Section 4.2. This should be possible since this method does not need the multiplicative property because the full closed loop $\mathcal{A}_{cl}$ is considered. The ISS LMIs are probably limited suitable for practical use in their current form since they only tune towards states and not to performance outputs. The H_2/H_∞ LMIs to protect against measurement noise were not used for the tuning of the GHS. For practical use of these LMIs most likely frequency-dependent matrices are necessary.

The effect of adding more dynamic extensions to the observer is concluded to be negative for tracking performance. This result is however not true in general but only for the tested tuning methods and with a noisy measurement signal.

The conclusion for the practical experiments is that the H_∞ synthesis method outperforms the empirical tuning method when no model info is used. However, it is possible that an experienced practitioner can

empirically determine controllers that perform better then the ones found in this thesis. But, for the case of large MIMO systems and especially when model information is included, the synthesis method will always be more time efficient than the empirical methods.

6.3 Future work

In this section future research directions that follow from this thesis are pointed out.

The co-design of matrices K and L for an ADRC scheme using an LMI based approach is still an open problem. The question that needs to be addressed is how to convexify the matrix that is obtained after substituting the closed loop ADRC representation in the concerned LMI.

The need for frequency-dependent weighting matrices is identified. Using these matrices high-frequency measurement noises and actuator dynamics can be taken into account during the synthesis. Furthermore, these matrices allow to 'shape' the disturbance rejection or tracking properties in certain frequency ranges. The first challenge is how to add these weighting filters to the closed loop and make the LMIs convex again? The second challenge is: how to select these weighting matrices?

The practical use for the derived H_2 and ISS LMIs still have to be validated. Furthermore, the combination of the different metrics, e.g., H_∞ for the observer and H_2 for the controller can be explored.

In Section 4.4.2, it is pointed out that the (synthesized) ADRC scheme is sensitive to the amplification of measurement noises. It is worth investigating the possibility to use low-pass filtered versions of the estimated states $\hat{x}$ and lumped disturbance $\hat{\xi}_0$ in the control.

In this thesis the ability to add known system parameters to the closed loop description used in the synthesis (and to the observer used in the control) is demonstrated. This can be extended to the usage of uncertain parameters that are known to be into a certain range.

In (2.1), control input u and disturbance input w share the same vector field $g(z)$. The ADRC formulation in Section 2.1.1 can be extended to the case where u, and w are multiplied with distinct vector fields.

For fault detection, it is interesting to find out whether it is possible to separate estimated lumped disturbance $\hat{\xi}$ (and its derivatives) in $\hat{\xi}_0 = \hat{\xi}_{0,endo} + \hat{\xi}_{0,exo}$, where subscripts "endo" and "exo" stand for endogenous and exogenous respectively. When (unexpected) changes in $\hat{\xi}_{endo}$ are observed this might be an indication of system failure.

It is possible to optimally tune a GPIO using the LMIs in this thesis. This optimally tuned GPIO can be used to decouple, and linearize (feedback linearizable) MIMO systems. Next, this linear system (with the estimation error as part of the disturbance input) can be tuned using tools like H_∞, H_2, and μ-synthesis in the general dynamic output feedback control framework, for which many (LMI based) tools exist that allow for frequency-dependent tuning matrices.

Bibliography

[1] I. D. Landau, T.-b. Airimitoaie, A. Castellanos-Silva, and A. Constantinescu, *BOOK-Advances in Industrial Control Adaptive and Robust Active Vibration Control*. 2016.

[2] A. F.-b. Approach, "Active Disturbance Rejection Control of Dynamic Systems," *Active Disturbance Rejection Control of Dynamic Systems*, 2017.

[3] Y. X. Su, C. H. Zheng, and B. Y. Duan, "Automatic disturbances rejection controller for precise motion control of permanent-magnet synchronous motors," *IEEE Transactions on Industrial Electronics*, vol. 52, no. 3, pp. 814–823, 2005.

[4] S. Bose, Y. V. Hote, and V. Siddhartha, "Analysis and Application of Linear ADRC for the Control of DC-DC Converters," *2019 5th Indian Control Conference, ICC 2019 - Proceedings*, no. Icc, pp. 436–441, 2019.

[5] L. Zhao, L. Liu, Y. Wang, and H. Yang, "Active Disturbance Rejection Control for Teleoperation Systems with Actuator Saturation," *Asian Journal of Control*, vol. 21, no. 2, pp. 702–713, 2019.

[6] Z. Chen, Q. Zheng, and Z. Gao, "Active disturbance rejection control of chemical processes," *Proceedings of the IEEE International Conference on Control Applications*, no. October, pp. 855–861, 2007.

[7] M. Ramírez-Neria, H. Sira-Ramírez, R. Garrido-Moctezuma, and A. Luviano-Juárez, "Linear active disturbance rejection control of underactuated systems: The case of the Furuta pendulum," *ISA Transactions*, vol. 53, no. 4, pp. 920–928, 2014.

[8] R. Madoński and P. Herman, "An experimental verification of ADRC robustness on a cross-coupled Aerodynamical System," *Proceedings - ISIE 2011: 2011 IEEE International Symposium on Industrial Electronics*, pp. 859–863, 2011.

[9] A. G. Wu and G. R. Duan, "Design of generalized PI observers for descriptor linear systems," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 53, no. 12, pp. 2828–2837, 2006.

[10] R. Madoński and P. Herman, "Survey on methods of increasing the efficiency of extended state disturbance observers," *ISA Transactions*, vol. 56, pp. 18–27, 2015.

[11] R. Miklosovic, A. Radke, and Z. Gao, "Discrete implementation and generalization of the extended state observer," *Proceedings of the American Control Conference*, vol. 2006, pp. 2209–2214, 2006.

[12] Z. Gao, "Scaling and Bandwidth-Parameterization based Controller Tuning," *Proceedings of the American Control Conference*, vol. 6, no. July 2003, pp. 4989–4996, 2003.

[13] X. Zhou, H. Gao, B. Zhao, and L. Zhao, "A GA-based parameters tuning method for an ADRC controller of ISP for aerial remote sensing applications," *ISA Transactions*, vol. 81, no. July, pp. 318–328, 2018.

[14] X. Hai, Z. Wang, Q. Feng, Y. Ren, B. Xu, J. Cui, and H. Duan, "Mobile Robot ADRC with an Automatic Parameter Tuning Mechanism via Modified Pigeon-Inspired Optimization," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 6, pp. 2616–2626, 2019.

[15] P. Teppa-Garran and G. Garcia, "Active disturbance rejection control tuning employing the LQR approach for decoupling uncertain multivariable systems," *Latin American Applied Research*, vol. 44, no. 4, pp. 355–361, 2014.

[16] A. Luviano-Juárez, J. CortÉs-Romero, and H. Sira-RamÍrez, "Synchronization of chaotic oscillators by means of generalized proportional integral observers," *International Journal of Bifurcation and Chaos*, vol. 20, no. 5, pp. 1509–1517, 2010.

[17] P. M. Frank and X. Ding, "Survey of robust residual generation and evaluation methods in observer-based fault detection systems," *Journal of Process Control*, vol. 7, no. 6, pp. 403–424, 1997.

[18] A. Luviano-Juárez, J. CortÉs-Romero, and H. Sira-RamÍrez, "Synchronization of chaotic oscillators by means of generalized proportional integral observers," *International Journal of Bifurcation and Chaos*, vol. 20, no. 5, pp. 1509–1517, 2010.

[19] D. Cheng, A. Isidori, W. Respondek, and T. J. Tarn, "Exact linearization of nonlinear systems with outputs," *Mathematical Systems Theory*, vol. 21, no. 1, pp. 63–83, 1988.

[20] A. D. Sokal, "A really simple elementary proof of the uniform boundedness theorem," *American Mathematical Monthly*, vol. 118, no. 5, pp. 450–452, 2011.

[21] T. J. Willmore, "The Definition of Lie Derivative," *Proceedings of the Edinburgh Mathematical Society*, vol. 12, no. 1, pp. 27–29, 1960.

[22] H. K. Khalil, *Nonlinear Systems*. Pearson Education, Prentice Hall, 2002.

[23] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *2. Some Standard Problems Involving LMIs*. 1994.

[24] R. J. Caverly and J. R. Forbes, "LMI Properties and Applications in Systems, Stability, and Control Theory," *arXiv preprint arXiv:1903.08599*, 2019.

[25] E. D. Sontag, *Input to state stability: Basic concepts and results*, vol. 1932. 2008.

[26] W. P. M. H. Heemels, J. Daafouz, and G. Millerioux, "Systems With Uncertain Parameters G : P i P i," vol. 55, no. 9, pp. 2130–2135, 1991.

[27] Z. P. Jiang, I. Mareels, D. J. Hill, and J. Huang, "A unifying framework for global regulation via nonlinear output feedback: From ISS to iISS," *IEEE Transactions on Automatic Control*, vol. 49, no. 4, pp. 549–562, 2004.

[28] P. Gahinet and P. Apkarian, "A linear matrix inequality approach to H control," *International Journal of Robust and Nonlinear Control*, vol. 4, no. 4, pp. 421–448, 1994.

[29] C. H. Lien, "Robust observer-based control of systems with state perturbations via LMI approach," *IEEE Transactions on Automatic Control*, vol. 49, no. 8, pp. 1365–1370, 2004.

[30] H. Kheloufi, A. Zemouche, F. Bedouhene, and M. Boutayeb, "On LMI conditions to design observer-based controllers for linear systems with parameter uncertainties," *Automatica*, vol. 49, no. 12, pp. 3700–3704, 2013.

[31] S. Skogestad and I. Postlethwaite, *Multivariable feedback control: analysis and design*, vol. 2. Citeseer, 2007.

[32] J. Peña Ramírez, *Huygens' synchronization of dynamical systems : beyond pendulum clocks*. No. 2013, 2013.

[33] D. This, "Eindhoven University of Technology Design of a mechanical synchronizing system for research and demonstration purposes for D van den Tillaart , M . H . L . M . Design of a mechanical synchronizing system for research and demonstration purposes Afstudeerop," 2006.

[34] D. J. Rijlaarsdam, "Huygens Synchronization in Various Dynamical," no. June, 2008.

[35] X. Chen, D. Li, Z. Gao, and C. Wang, "Tuning method for second-order active disturbance rejection control," *Proceedings of the 30th Chinese Control Conference, CCC 2011*, pp. 6322–6327, 2011.

Appendix A

Example second-order SISO system with 2 extensions

Consider the following system in normal form

$$\begin{aligned} \dot{x} &= Ax + B_u u + B_\xi \xi, \\ y &= Cx + Du + \nu, \end{aligned} \tag{A.1}$$

with

$$A = \begin{bmatrix} 0 & 1 \\ -a_0 & -a_1 \end{bmatrix}, B_u = \begin{bmatrix} 0 \\ b \end{bmatrix}, B_\xi = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \end{bmatrix}, D = \begin{bmatrix} 0 \end{bmatrix}. \tag{A.2}$$

When we choose to add 2 states (r=2) we can write the extended system as

$$\begin{aligned} \underbrace{\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{\xi} = \dot{\xi}_0 \\ \ddot{\xi} = \dot{\xi}_1 \end{bmatrix}}_{\dot{\eta}} &= \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ -a_0 & -a_1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{A_e} \underbrace{\begin{bmatrix} x1 \\ x2 \\ \xi_0 \\ \xi_1 \end{bmatrix}}_{\eta} + \underbrace{\begin{bmatrix} 0 \\ b \\ 0 \\ 0 \end{bmatrix}}_{B_{ue}} u + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}}_{B_{\xi e}} \ddot{\xi}, \\ y_\eta &= C_e \eta + \nu, \end{aligned} \tag{A.3}$$

with

$$C_e = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}. \tag{A.4}$$

A Luenberger type of observer is defined as

$$\begin{aligned} \dot{\hat{\eta}} &= A_e \hat{\eta} + B_{ue} u + L(y - \hat{y}) = (A_e - LC_e)\hat{\eta} + L\nu + LC_e \eta + B_{ue} u, \\ \hat{y} &= C_e \hat{\eta}. \end{aligned} \tag{A.5}$$

Introducing the estimation error coordinates $e_\eta = \eta - \hat{\eta}$ we can write the estimation error dynamics

$$\dot{e}_\eta = \dot{\eta} - \dot{\hat{\eta}} = A_e \eta + B_{ue} u + B_{\xi e} \xi^{(r)} - ((A_e - LC_e)\hat{\eta} + L\nu + LC_e \eta + B_{ue} u). \tag{A.6}$$

Using the fact $\hat{\eta} = \eta - e_\eta$ yields

$$\begin{aligned}\dot{e}_\eta &= A_e\eta + B_{ue}u + B_{\xi e}\xi^{(r)} - ((A_e - LC_e)(\eta - e_\eta) + L\nu + LC_e\eta + B_{ue}u),\\ &= (A_e - LC_e)e_\eta - L\nu + B_{\xi e}\xi^{(r)}.\end{aligned} \tag{A.7}$$

The next step is the substitution of the tracking controller in the x dynamics.

$$u = -b^{-1}(k_1(\hat{x}_1 - x_1^*) + k_2(\hat{x}_2 - x_2^*) + \hat{\xi}_0) = -b^{-1}(K(\hat{x} - x^*) + \hat{\xi}_0) \tag{A.8}$$

Using $\hat{x} = x - e_x$, and $\hat{\xi}_0 = \xi_0 - e_\xi$ can write

$$u = -b^{-1}(K(x - e_x - x^*) + \xi_0 - e_{\xi 0}). \tag{A.9}$$

Substitution of u in the x dynamics (A.1) and using $B_u b^{-1} = \begin{bmatrix}0\\1\end{bmatrix} = B_\xi$ gives

$$\dot{x} = Ax + B_u u + B_\xi \xi = (A - B_\xi K)x + B_\xi(K(e_x + x^*) + e_\xi). \tag{A.10}$$

After expansion the x dynamics interconnected with the e dynamics look like

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{e}_{x_1}\\ \dot{e}_{x_2}\\ \dot{e}_{\xi 0}\\ \dot{e}_{\xi 1}\end{bmatrix} = \underbrace{\begin{bmatrix}0 & 1 & 0 & 0 & 0 & 0\\ -(a_0+k_1) & -(a_1+k_2) & k_1 & k_2 & 1 & 0\\ 0 & 0 & -l_1 & 1 & 0 & 0\\ 0 & 0 & -(a_0+l_2) & -a_1 & 1 & 0\\ 0 & 0 & -l_3 & 0 & 0 & 1\\ 0 & 0 & -l_4 & 0 & 0 & 0\end{bmatrix}}_{\mathcal{A}_{cl}} \begin{bmatrix}x_1\\ x_2\\ x_1 - \hat{x}_1\\ x_2 - \hat{x}_2\\ \xi_0 - \hat{\xi}_0\\ 0\end{bmatrix} + \begin{bmatrix}0\\0\\0\\0\\0\\1\end{bmatrix}\ddot{\xi} + \begin{bmatrix}0\\0\\l_1\\l_2\\l_3\\l_4\end{bmatrix}\nu + \begin{bmatrix}0 & 0\\ k_1 & k_2\\ 0 & 0\\ 0 & 0\\ 0 & 0\end{bmatrix} x^*, \tag{A.11}$$

which has the form,

$$\begin{bmatrix}\dot{x}\\ \dot{e}_\eta\end{bmatrix} = \begin{bmatrix}A - B_\xi K & \overline{BK}\\ 0_{(n+m)\times n} & A_e - LC_e\end{bmatrix}\begin{bmatrix}x\\ e_\eta\end{bmatrix} + \begin{bmatrix}0\\ \overline{B_\xi}\end{bmatrix}\xi^{(r)} + \begin{bmatrix}0\\ L\end{bmatrix}\nu + \begin{bmatrix}B_\xi K\\ 0\end{bmatrix}x^*, \tag{A.12}$$

with

$$\overline{BK} := [B_\xi K, B_\xi, 0], \qquad \overline{B_\xi} := \begin{bmatrix}0\\ \vdots\\ 0\\ I\end{bmatrix}. \tag{A.13}$$

Appendix B

Code

```
clc; clear all; close all;

% ADRC Synthesis HINF Tom Leunissen 2021
% To run this code, Yalmip should be installed. The tested SDP solvers are Sedumi and Mosek.

[A,B,C,D] = Huygens_model; %Load model, this model is used for the syntheses.

%% extended system

m = 1; % number of dynamic extensions (per mass/diff eq) !!

n = size(A,1); % dimension original system (total) =  ni*p  (with p 2 for fully actuated system)
ni = size(B,2); % number of inputs
no = size(C,1); % number of outputs
B_ = [zeros(n/2,ni);eye(n/2,ni)];   % Normalized B matrix B tilde (for fully actuated system)

Z = zeros(no,ni);                   % Zero matrix

phi = [zeros((m-1)*ni,ni), eye((m-1)*ni);
       Z,zeros(ni,((m-1)*ni))];

A_e = [A,B_,zeros(n,ni*m-ni);   % Extended state matrix (\eta)
    zeros(m*ni,n),phi];

B_e = [B;zeros(m*ni,ni)];       % Extended input matrix

C_e = [C zeros(no,m*ni)];       % Extended output matrix

D_e = zeros(no,ni);             % Extended feed through matrix

b = B(ni+1:n,1:ni);  % Inverse used in control (R^-1)

%% make struct with sys vars

%These values go to the observer in SIMULINK
ext.A_e = A_e;
ext.B_e = B_e;
ext.C_e = C_e;
ext.D_e = D_e;
```

```
%% LMI Observer  H_inf (estimation error)

W2=logspace(0,9,250);   % search space for w2--> w_\xi in report!

for jj= 1:length(W2)

w2 = 1/W2(jj);

WW=[eye(n),zeros(n,3);  % Weighting matrix W_e
     zeros(3,n),eye(3)*w2];

Cz=WW*[eye(n+no),zeros(n+no,size(A_e,1)-n-no)];  % Performance output estimation error

Bw=[zeros(size(A_e,1)-size(B_,1),size(B_,2));B_];   % Performance input estimation error

%% Setting up LMI for observer

I1 = eye(no);
% define variables
P = sdpvar(size(A_e,1));
lh = sdpvar(size(A_e,1),no); %L^
Z = sdpvar(size(Cz,1));
g = sdpvar(1);   %gamma_l
eta = 0.00;   %
I2=eye(size(Cz,1));

m11 = A_e.'*P+P*A_e-lh*C_e-C_e.'*lh.';
m12 = P*Bw;
m13 = Cz.';
m22 = -g*I1;
m23 = zeros(size(m22,1),size(Cz,1));
m33 = -I2;

LMI = [m11 m12 m13;
       m12.' m22 m23
       m13.' m23.' m33];

       % Define Constraints
F = [];
F = [F,LMI<=eta];
F = [F,P>=eta];
F = [F,g>=eta];

opt = sdpsettings;   % Standard should work

% Solution observer

sol= optimize(F,g,opt)

gamma = (sqrt(value(g))) % remember square root
lh = value(lh);
P = value(P);
L = inv(P)*lh;    %  compute  L
gamobs=gamma;

% Interpret results
```

```
p_o = eig(A_e-L*C_e)  % Eigenvalues unforced observer

%% Check if no constraints were violated

check1 = eig(P)>0;
check2 = eig(value(LMI))<0;

if min(check2)==0
    'error 1'
    pause
end
if     min(check1)==0
'error 2'
end

%% Controller part HINF

MU=logspace(0,9,1000);   %Define search range \mu

GAMC=[];
HINF=[];
REP=[];
KK=[];
count=[0]; %initialize count
for i=1:length(MU)

Bxi = B_;

B=Bxi;  check which B you want to use!! (un)comment

Cz = [eye(no),zeros(size(A,1)-no)];     %Performance output controller part (e.g. tracking error)
S = sdpvar(size(A,1));
kh = sdpvar(ni,n);
Z = sdpvar(size(Cz,1));
I = eye(1);
DD = zeros(1);

mu=MU(i);

m11 = S*A.'+A*S-kh.'*B.'-B*kh; %
m12 = B*kh;
m13 = Bxi*(1/w2);   %  w_\xi
m14 = S*Cz.';
m22 = (-2*S+eye(size(S,1)))*mu;
m23 = zeros(size(m22,1),size(m13,2));
m24 = zeros(size(S,1),size(m14,2));
m33 = -mu*eye(size(m13,2));
m34 = zeros(size(Bxi,2),size(m14,2));
m44 = -eye(size(Cz,1));

LMI = [m11 m12 m13 m14;
        m12.' m22 m23 m24;
        m13.' m23.' m33 m34;
        m14.' m24.' m34.' m44];

eta =0.000;
```

```
%% Define Constraints

F = [];
F = [F, LMI<=eta*eye(size(LMI,1))];
F = [F, S>=0];

%% settings

opt = sdpsettings('verbose',0);     %do not print results

%% sol
sol= optimize(F,[],opt);        % Find feasible sol for each mu

kh = value(kh);
S = value(S);
gamcont=((sqrt(mu)));  %
 K = kh*inv(S);  % compute K

p_k = eig(A-B*K);

%% Check  if constraints are not violated

c1=min(eig(S)>0);
c2= min(eig(value(LMI))<0);

ch = c1.*c2;

if  ch==1;
count= count+1    % only safe valid solutions

fmu(count) = mu;  % feasible \mu

% % total closed loop
A_cl = [A-B*K [B*K Bxi zeros(n,ni*(m-1))];  check whether B=B_xi is (un)commented
        zeros(ni*m+n,n) A_e-L*C_e];
B_cl = [zeros(size(A,1),size(B_,2)) ;zeros(size(A_e,1)-size(B_,1),size(B_,2));B_];
C_cl = [eye(no), zeros(no,size(A_cl,1)-no) ];
D_cl=[];

sys = ss(A_cl,B_cl,C_cl,D_cl);

clhinfgain=norm(sys,inf);  % real closed loop gain
HINFtot(count) = clhinfgain; % safe Hinf gains in vector
gamtot(count) = gamcont;
KK{count} = K;  % Cell array with gains

end
end

[min_hinf,I] = min(HINFtot); % find index of smallest Hinf tot
K = KK{I}  % select K that gives minimum Hinf
min_hinf =  min_hinf % print minimum Hinf
p_k = eig(A-B*K)
p_o

HHH(Cb)=min_hinf   % hinf norm min
```

```
Cb=Cb+1   % count large loop
end

% load ADRC.mat into comp setup

$f = 300; %  input filter [Hz] (experimental)

ADRC.A = A;
ADRC.A_e = A_e;
ADRC.b = b;
ADRC.B_e = B_e;
ADRC.C_e = C_e;
ADRC.A_e = A_e;
ADRC.ext = ext;
ADRC.K = K;
ADRC.L = L;
ADRC.m = m;
ADRC.n =n;
ADRC.ni = ni;
ADRC.no = no;
ADRC.min_hinf=min_hinf;
ADRC.f=f;
```

Declaration concerning the TU/e Code of Scientific Conduct for the Master's thesis

I have read the TU/e Code of Scientific Conduct[i].

I hereby declare that my Master's thesis has been carried out in accordance with the rules of the TU/e Code of Scientific Conduct

Date

October 14, 2021

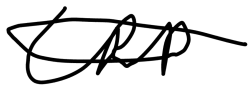
Name

Tom Leunissen

ID-number

0982190

Signature



Submit the signed declaration to the student administration of your department.

[i] See: http://www.tue.nl/en/university/about-the-university/integrity/scientific-integrity/
The Netherlands Code of Conduct for Academic Practice of the VSNU can be found here also.
More information about scientific integrity is published on the websites of TU/e and VSNU