Controller Design for Multistorey Buildings via Convex Optimisation

Emil Vladu

LUND UNIVERSITY

Department of Automatic Control

MSc Thesis
TFRT-6069
ISSN 0280-5316

Department of Automatic Control
Lund University
Box 118
SE-221 00 LUND
Sweden

Printed in Sweden by Tryckeriet i E-huset
Lund 2018

Abstract

Earthquakes, wind and traffic may cause unwanted vibrations in buildings. Vibration control devices are often installed between floors to suppress such disturbances. In the context of buildings, controllers are commonly designed using methods which are subject to a number of limitations: we cannot discuss optimal performance or conclude that no controller exists which satisfies a given set of design specifications.

In this thesis, we address these shortcomings by considering convex optimisation for controller design in buildings. Given a restricted set of design specifications typical in vibration control, we show that the controller design problem may be formulated as a convex optimisation problem when the building is modelled as a chain of masses interconnected by linear springs and dampers. The design specifications for which this is shown are internal stability, achievability by some controller, and upper and lower bounds in the frequency and time domain.

This method is then demonstrated for a chain of five masses subjected to an impulse. A set of controller design problems are formulated and solved using CVX in Matlab. A specific design problem is also solved for different finite-dimensional approximations, and the result suggests convergence to some minimum. Tradeoff curves comparing actuator effort to intermass displacement in both the frequency and time domain are successfully computed. These results are then compared with the performance of the corresponding passive system in which dampers have been added. It is found in particular that the greatest damper force can be over 200 times larger than that of an optimal controller achieving the same performance.

Acknowledgements

I would like to thank my supervisor, Kaoru Yamamoto, for her insightful thoughts, her sensible advices, her patience with my many musings, and her constant support during this thesis work. I would also like to thank my examiner Anders Rantzer and my opponents Richard Bai and Karl Fredrik Erliksson for their very helpful comments on my work. Finally, I am grateful to the Department of Automatic Control at Lund University for allowing me this opportunity to work on this project.

Contents

Nomenclature

(C)	a convex optimisation problem
δ_i	the intermass displacement $x_i - x_{i-1}$
$\hat{H}_n$	a special function from $\mathbb{R}^{n_u n_y n}$ to $\mathscr{H}$
$\mathscr{D}_*$	the design specification that a given set of design specifications $\{\mathscr{D}_i\}$ be simultaneously satisfied
$\mathscr{D}_{\mathbb{C}stab}$	the design specification that $\mathscr{D}_*$ while excluding $\mathscr{D}_{stab}$
$\mathscr{D}_{\delta f,\alpha_{\delta f}}$	the design specification that $\phi_{\delta f}(H) < \alpha_{\delta f}$
$\mathscr{D}_{\delta t,\alpha_{\delta t}}$	the design specification that $\phi_{\delta t}(H) < \alpha_{\delta t}$
$\mathscr{D}_{fd*,\,n}$	the design specification that both $\mathscr{D}_{fdstab,\,n}$ and $\mathscr{D}_{\mathbb{C}stab}$
$\mathscr{D}_{fdstab,\,n}$	the design specification that $H \in \mathscr{H}$ belong to the finite-dimensional part of $\mathscr{H}_{stab}$ generated by means of the n first basis functions Q_k
$\mathscr{D}_i$	a design specification indexed by i
$\mathscr{D}_{stab}$	the design specification that $H \in \mathscr{H}$ be achieved by some stabilising controller
$\mathscr{D}_{ut,\alpha_{ut}}$	the design specification that $\phi_{ut}(H) < \alpha_{ut}$
$\mathscr{H}$	the set of all $n_z \times n_w$ transfer matrices
$\mathscr{H}_i$	the set of all $H \in \mathscr{H}$ for which $\mathscr{D}_i$ is true
$\mathscr{R}$	the set of all stable $n_u \times n_y$ transfer matrices
$\mathscr{V}_n$	the set of all $x \in \mathbb{R}^{n_u n_y n}$ such that $\hat{H}_n(x) \in \mathscr{H}_{\mathbb{C}stab}$
$\phi_{\delta f}$	a functional such that $\phi_{\delta f}(H) = \max\limits_{1 \le i \le N} \sup\limits_{\omega \in [0,\omega_0]} \lvert H_{\delta_i x_0}(i\omega) \rvert$

$\phi_{\delta t}$	a functional such that $\phi_{\delta t}(H) = \max_{1\leq i\leq N} \sup_{t\in[0,t_0]} \lvert\mathscr{L}^{-1}(H_{\delta_i x_0})(t)\rvert$
ϕ_{ut}	a functional such that $\phi_{ut}(H) = \max_{1\leq i\leq N} \sup_{t\in[0,t_0]} \lvert\mathscr{L}^{-1}(H_{u_i x_0})(t)\rvert$
ξ	the damping ratio
$\{\mathscr{D}_{\phi,\alpha}\}$	a design objective given an associated functional ϕ
d_i	the damping coefficient of the damper located between the i-1:th and the i:th mass
F_{d_i}	the force affecting the i:th mass as caused by the i:th damper
F_{k_i}	the force affecting the i:th mass as caused by the i:th spring
H	the closed-loop transfer matrix
H^0	$\hat{H}_n$ applied to the solution of (C)
I	the identity matrix
J	the dimension of the constraint set
K	the controller matrix
k_i	the spring constant of the spring located between the i-1:th and the i:th mass
m_i	the mass of the i:th storey
N	the number of masses included in the mass chain
n	the number of elements included in the finite sequence Q_k
n_w	the number of elements of the vector w
P_{yu}	the transfer matrix from u to y
Q_k	a basis function for any positive integer k
u	the actuator inputs: u_i is the force affecting m_i as caused by the i:th controller device
w	the exogenous inputs: $w = x_0$
x_0	the ground displacement
x_i	the displacement of the i:th mass
y	the sensed outputs: y_i is taken as δ_i
z	the regulated outputs: z_i is taken as δ_i and u_i

1

Introduction

Background

Earthquakes, traffic and wind are examples of disturbances which tend to cause unwanted vibrations in buildings. Dampers and vibration control devices are often installed between storeys to combat this issue. Requirements on the system behaviour are formulated and controllers are designed to meet these requirements. However, commonly used design methods are lacking in some important respects. For instance, if such methods are unsuccesful in generating an acceptable controller, one cannot conclude immediately that the constraints are too tight: a different approach could very well be more successful. Conversely, even if an acceptable controller were to be obtained, other more efficient choices may have been overlooked. It is useful for the designer to know, for instance, that there are other controllers which satisfy the constraints with much less effort; it is just as useful to know that a controller with simple structure performs almost as well as a best achiever.

In order to arrive at such a best achieving controller, it is natural to resort to some kind of minimisation over a set of candidate controllers. Local minima will not suffice, however, as the above shortcomings will have to be dealt with once more. Instead, we require global minima in order to properly evaluate such limits of performance. Convex optimisation is the key to this.

A convex optimisation problem is a type of minimisation problem for which there exist efficient numerical algorithms which guarantee convergence to a global minimum. More specifically, we are guaranteed to arrive to within an arbitrarily small distance of a global minimum in a finite number of iterations. Thus, if the particular controller design problem could somehow be translated into a convex optimisation problem, we would be in a position to remedy the above shortcomings. As it happens, recent theoretical developments has led [Boyd et al., 1990] to propose a programme for doing exactly this for a restricted set of systems and constraints. In this thesis, we adapt these results to the context of vibration control in buildings.

Purpose and Task

This thesis examines the circumstances under which it is possible to formulate the controller design problem for buildings as a convex optimisation problem. As a point of entry, the building is modelled as a chain of masses connected by linear springs and dampers. We verify that this system along with an accompanying set of requirements on system behaviour are on the right form to be amenable to the controller design procedure suggested in [Boyd et al., 1990]. The requirements which we consider are typical in the context of vibration control.

We then proceed to demonstrate the design method for a system of five masses subjected to an impulse disturbance. A set of controller design problems are formulated and restated as convex optimisation problems; these are subsequently solved using CVX in Matlab. The results are compared with the corresponding passive system in which external damping is added. We also evaluate the best achievable performance, comparing actuator effort to intermass displacement in both the frequency and time domain by means of tradeoff curves. Finally, a specific design problem is solved with different finite-dimensional approximations.

Outline

The thesis is structured as follows:

Chapter 2 is the background chapter in which we make important definitions and assumptions. In particular, the *mass chain model* for modelling buildings is introduced along with the related assumptions. Further, we review some basic concepts about structural engineering and control theory.

Chapter 3 describes the general procedure for translating a controller design problem into a convex optimisation problem. Here, we introduce *design specifications*, a formalisation of the demands on the system behaviour, and *convexity*, the crucial restriction which must be imposed on the design specifications in order to arrive at a convex optimisation problem.

Chapter 4 applies the design method outlined in Chapter 3 to the mass chain model described in Chapter 2. In particular, we put up the equations of motion for the mass chain system and describe it within the framework introduced in Chapter 2. This allows us to connect it to the contents of Chapter 3. We then consider various sets of material parameters and determine when the design method in question may be exploited to yield a convex optimisation problem.

Chapter 5 considers a specific mass chain system with five masses. In this chapter, the method detailed in Chapter 3 is put into practice. A set of controller design problems are posed and their corresponding convex optimisation problems are formulated and solved. In particular, we consider best achievable performances and compute tradeoff curves.

Chapter 6 discusses the results of Chapter 3, 4 and 5 and points at suggestions for future work on the subject.

2

Buildings and Control

In this section, we lay the foundation on which the remainder of the thesis depends. We discuss relevant facts about buildings and vibration suppression for later use. Further, we establish the conceptual framework within which we will carry out our analysis.

In particular, Section 2.1 discusses the physical reality of buildings and earthquakes. From this, various goals of regulation are anticipated and formulated as informal design specifications. The mass chain model for modelling buildings is then introduced in Section 2.2 alongside the necessary assumptions that such a model entails. Passive interconnections are considered in some detail. Finally, in Section 2.3 we review basic concepts and terminology from systems theory, including stability. In connection with this we also consider controller action. The definitions are adopted to a large extent from [Boyd et al., 1990].

2.1 Vibration Suppression in Buildings

There are many sources of disturbance associated with buildings, e.g. traffic, wind and earthquakes. Various means of disturbance rejection may be implemented either passively or actively. In the passive case, there is only dissipation and no external energy is added. A realisation of this appears, for instance, as a set of dampers between floors designed to absorb vibrational energy. By contrast, in the active case, energy is funnelled into the system from an external source. Controller devices positioned between floors could constitute this external agent. A combination of dampers and controllers is also employed, and a comparison between active and passive disturbance rejection is therefore interesting.

The difficulties we face as a result of vibrations will depend in general on the nature of the disturbance. For instance, surrounding traffic may be more of a comfort issue for the residents, whereas earthquakes will threaten the structural integrity of the building. The better known the disturbance profile, the more efficient the means of counteracting it will be. In the following, we therefore restrict ourselves

to earthquakes. This is because earthquakes constitute a recurring feature in some parts of the world, and their particular characteristics are well understood.

Earthquakes come in a variety of shapes and sizes. They may be analysed by means of their acceleration/velocity/displacement time histories and corresponding spectrums. There are many factors that come into play in determining the appearance of these graphs. For instance, the location of the hypocenter, the fault type and the distance away from the rupture will all be of consequence [USGS, 2018].

A particular kind of earthquake with which we concern ourselves in Chapter 4 can be caused by near-fault ground motion resulting from forward-directivity. This kind of ground motion is pulse-like and intense, and may be characterised by a small number of parameters [Mavroeidis and Papageorgiou, 2003]. However, under some circumstances, a pulse in the accelerogram associated with such ground motion may be idealised as an impulse [Kojima and Takewaki, 2015]. In the same vein, we allow an impulse to model the characteristic pulse in the displacement time history.

There are many aspects to take into account when reducing the impact of earthquakes. An important such factor is their low-frequency character: the highest activity in displacement tends to be located in the 0-10 rad/s frequency band [Yamamoto, 2016]. The natural frequencies of the building ought therefore not to coincide with that particular range. From a control perspective, the goal is to reduce the horizontal displacement between neighbouring floors: too great a distance would fracture the supporting columns and result in building collapse.

Special care has to be taken with the equipment designed to aid. For instance, in response to a violent quake, dampers themselves could damage the columns. The damping mechanism would have to be reworked such that nonlinearities be introduced to remedy this.

Controllers also require special attention. Minimising actuator effort is always beneficial, if not for damage prevention, then at least for the sake of economy. In response to large signals - a given with earthquakes - the controller device might overheat or incur mechanical damage. Saturation and the related windup phenomenon also have to be accounted for, and so forth.

In the following, we restrict ourselves to a particular subset of these pitfalls:

- Reduce horizontal displacement between adjacent floors in some appropriate sense.

- Reduce actuator effort in some appropriate sense.

Of course, the exact quantity we reduce will depend on what we wish to achieve, and there are different metrics for different purposes. This will be specified later on.

The choice in the list strikes a balance between the relative importance of these goals in engineering, and the mathematical width they offer. In the chapters to come, a suitable framework will be introduced in which these goals may be formalised as design specifications for controller design.

2.2 The Mass Chain Model

The main aim of this project is to examine the circumstances under which controller design via convex optimisation is possible in the domain of buildings. It therefore stands to reason that a simple model for buildings be used: should the results look promising, one may proceed to more sophisticated models. The model we will employ is called the *mass chain model*.

Considered through the mass chain model, the continuous mass distribution of a building is discretised and lumped into N point masses corresponding to the N floors of the building. The mass m_i corresponds to the mass of the i:th floor, $1 \leq i \leq N$. Torsion or rotation of the masses is assumed to be non-existent. These masses are then interconnected with passive or active components, in our case springs and dampers (passive) and controllers (active). An additional layer of these components separates the first floor m_1 from the disturbance source. The components are assumed to be connected in parallel. A schematic of this interconnection is shown in figure 2.1.

The disturbance is assumed to enter at ground level. It is further assumed to act horizontally in one dimension. This is not unreasonable for an earthquake. In response, the N masses are likewise assumed to move horizontally in one dimension. The default state in which there is no disturbance and the building is at rest is referred to as the *zero state*.

With these assumptions in mind, let $x_0 = x_0(t)$ denote the horizontal deviation of the ground from its position in the zero state. Further, let $x_i = x_i(t)$ denote the displacement of the i:th mass such that $x_i = 0$ corresponds to its position in the zero state. A visual representation of this is shown in figure 2.2.

We close this section on the definition of the mass chain model by commenting on its applicability. Despite its simplicity, the model is widely employed in the field of structural engineering, not least in relation to earthquakes, e.g. [Yamamoto, 2016]. However, the same model appears also in various other domains, such as in the study of vehicle platooning [Rajamani, 2005]. A platoon is a close-packed group of vehicles proceeding in the same direction, and they become relevant in the area of automated highway systems in which the vehicles are fully automated. Altogether, the work in this thesis may find use beyond vibration control in buildings.

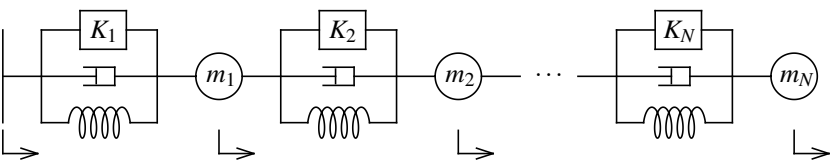

Figure 2.1 A schematic of the mass chain model. Here, K_i represent controller devices.

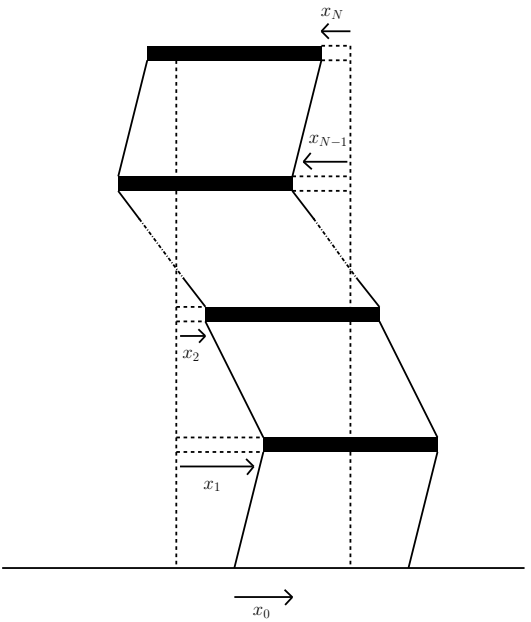

Figure 2.2 An abstraction of a building subjected to shear forces. The dashed lines represent the building at rest.

The Passive Interconnection

Consider now the springs and the dampers between the masses. The spring will model the inherent elasticity of the columns which support the structure. Due to gravity, the floors above a column will press down upon it. However, this vertical compression is not a parameter with which we will concern ourselves. Rather, we will consider only the *horizontal* column stretching caused by shear forces as the floors slide sideways.

This horizontal column stiffness will be modelled by linear springs obeying Hooke's law. The spring between m_i and m_{i-1} is therefore associated with a spring constant $k_i > 0$. In our case, the spring is at rest if its two ends overlap. This may be interpreted as the supporting column being completely vertical. When this state of rest is disturbed, the spring responds by generating a force proportional to the difference in horizontal position between its two ends. Seeing as these are attached to m_i and m_{i-1}, we may instead consider the *intermass displacement* $\delta_i = x_i - x_{i-1}$. Defining F_{k_i} as the force affecting the i:th mass m_i as caused by the i:th spring, Hooke's law then gives

$$F_{k_i} = -k_i(x_i - x_{i-1}) = -k_i\delta_i \tag{2.1}$$

Note that if m_i lies to the right of m_{i-1}, then $\delta_i > 0$ and so, by (2.1), $F_{k_i} < 0$. Hence,

the force exerted on m_i by the i:th spring will attempt to drag m_i to the left towards m_{i-1}, as it should.

As for damping, there are two main sources from which it is produced. One kind of damping stems from the material properties of columns, which dissipate energy by default. This is called *structural damping*. The other is an external addition of damping, perhaps in the form of a cylindrical device placed between floors containing a viscous fluid [Yamamoto, 2016].

We will assume a linear law for either form of damping, analogous to (2.1). The damper between m_i and m_{i-1} is therefore associated with a damping coefficient $d_i > 0$. Unlike in the case of a spring, however, we consider the difference in velocity between m_i and m_{i-1}. Defining F_{d_i} as the force affecting the i:th mass m_i as caused by the i:th damper, we have

$$F_{d_i} = -d_i(\dot{x}_i - \dot{x}_{i-1}) = -d_i\dot{\delta}_i \tag{2.2}$$

Note that if m_i moves faster than m_{i-1} to the right, then $\dot{\delta}_i > 0$ and so by the above $F_{d_i} < 0$. Hence, the force exerted on m_i by the i:th damper will attempt to increase the velocity of m_i to the left so as to match the velocity of m_{i-1}, as it should.

The damping coefficient may be further separated into

$$d_i = \frac{2\xi k_i}{\omega} \tag{2.3}$$

where the natural frequency ω is given by

$$\omega = \frac{2\pi}{0.1N} \tag{2.4}$$

ξ is called the damping ratio. It is customary in structural engineering to choose $\xi \in [0, 0.02]$ when modelling structural damping, and $\xi \in [0, 0.2]$ when modelling added external damping. Note that the denominator in (2.4) has been retrieved empirically.

2.3 Systems and Related Concepts

The system with its internal dynamics will be acted on by input signals to produce output signals. Sometimes, a controller is connected to modify the dynamics. An example is offered by the system detailed above. There, a reasonable input signal would be the ground movement x_0. By contrast, any signals of interest, such as the intermass displacements δ_i, would constitute the output signal. Controller action enters between floors, as shown in figure 2.1.

More formally, a distinction will be made between the signals that are directly related to the controller and those that are not. Input signals are divided into *actuator inputs* $u = u(t)$, signals that are generated by the controller and are funnelled into the system, and *exogenous inputs* $w = w(t)$, all remaining signals that somehow

affect the system from the outside. Output signals are similarly divided into *sensed outputs* $y = y(t)$, signals that the controller may access directly, i.e. controller inputs, and *regulated outputs* $z = z(t)$, which ideally consist of every signal required such that various performance measures may be fully evaluated. Therefore, z will often include the actuator inputs u and might overlap to some extent with the sensed outputs y. When no controller is connected, w and z are referred to simply as input and output signals respectively. In general, w, u, z and y are vector-valued, and their corresponding number of elements are denoted n_w, n_u, n_z and n_y respectively.

LTI Systems

We now turn to consider systems in general. The model of the system together with the signals w and z will be referred to as the *plant*, denoted P. In the following, the plant P and any controller K are assumed to be linear and time-invariant (LTI).

Consider a plant with a scalar input signal $w = w(t)$ and a scalar output signal $z = z(t)$. The LTI property implies that there exists a unique signal-independent function $p = p(t)$ such that for any w, we may obtain z by convolving w with p. p therefore describes the plant and its dynamics completely.

Now, in the Laplace domain, the convolution operation becomes simple multiplication. In other words, given an LTI system, there exists a unique signal-independent function $P = P(s)$, called the transfer function, such that for any w

$$Z(s) = P(s)W(s)$$

where $\mathscr{L}w(s) = W(s)$ and $\mathscr{L}z(s) = Z(s)$. We establish the convention that for any signal for which the Laplace transform $\mathscr{L}$ is well-defined, the transformed function is denoted by the corresponding capital letter.

For a general LTI plant with several input and output signals, the plant may be similarly described by a set of transfer functions, one for each input-output pair. The transfer function from w_j to z_i describes the relation between w_j and z_i in the Laplace domain, and is denoted $P_{z_i w_j}$. The transfer functions may be collected into a transfer matrix P to express the relation between input and output signals concisely as $Z = PW$. Assuming some of those signals are related to a controller according to the discussion above, this becomes

$$\begin{pmatrix} Z \\ Y \end{pmatrix} = P \begin{pmatrix} W \\ U \end{pmatrix}$$

where P may be partitioned as

$$P = \begin{pmatrix} P_{zw} & P_{zu} \\ P_{yw} & P_{yu} \end{pmatrix}$$

so that

$$Z = P_{zw}W + P_{zu}U \tag{2.5}$$
$$Y = P_{yw}W + P_{yu}U \tag{2.6}$$

The Closed-Loop System

The *closed-loop system* arises when a controller K is introduced to modify the dynamics of the plant, as shown in figure 2.3. This new system has as input signal the exogenous inputs of the plant w and as output signal the regulated outputs z. Closing the loop and invoking the assumption that the controller K be LTI, the actuator signal U may now be expressed as

$$U = KY \tag{2.7}$$

Inserting (2.7) into (2.6) yields

$$Y = P_{yw}W + P_{yu}KY$$

which may be solved for Y to obtain

$$Y = (I - P_{yu}K)^{-1}P_{yw}W$$

By substituting this new expression for Y into (2.7) and then plugging (2.7) into (2.5), we arrive at an expression for Z as a function of W

$$Z = (P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw})W$$

From this relation between W and Z we may identify the closed-loop transfer matrix H

$$H = P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw} \tag{2.8}$$

The matrix H is an $n_z \times n_w$ transfer matrix consisting of various transfer functions. The entry in the i:th row and j:th column will be denoted H_{ij}. H is fundamental for the kind of design procedures we intend to employ, as we shall find in the chapters to come.

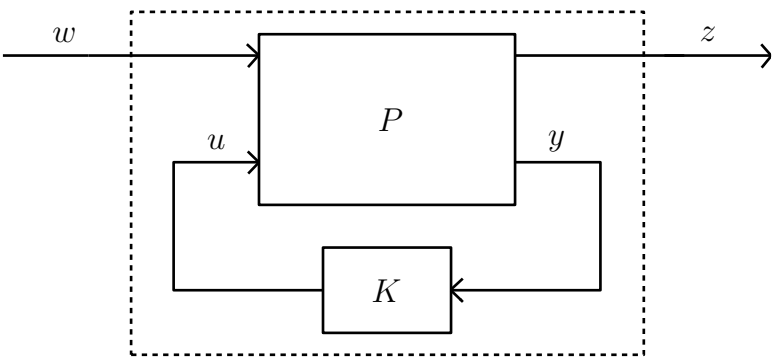

Figure 2.3 System, controller and signals visualised in a standard configuration. The dashed box represents the closed-loop system.

Stability

We now consider the important concept of stability for transfer functions and matrices.

There are various definitions of stability depending on purpose. In this thesis, we will directly adopt the definitions used in [Boyd et al., 1990]. A rational transfer function is said to be *stable* if there are no more zeros than poles and the poles have negative real part. A transfer matrix is said to be stable if all of its elements are stable.

Stability is preserved under matrix addition and multiplication. To see this, consider two rational functions $\frac{Q_{n1}}{Q_{d1}}$ and $\frac{Q_{n2}}{Q_{d2}}$ where the numerators and denominators are all polynomials and $Q_{d1}, Q_{d2} \neq 0$. Then

$$\frac{Q_{n1}}{Q_{d1}} + \frac{Q_{n2}}{Q_{d2}} = \frac{Q_{n1}Q_{d2} + Q_{n2}Q_{d1}}{Q_{d1}Q_{d2}}$$

and

$$\frac{Q_{n1}}{Q_{d1}} \cdot \frac{Q_{n2}}{Q_{d2}} = \frac{Q_{n1}Q_{n2}}{Q_{d1}Q_{d2}}$$

If the zeros of Q_{d1} and Q_{d2} have negative real part, then so do the zeros of the new pole polynomial $Q_{d1}Q_{d2}$. Further, if $\deg(Q_{n1}) \leq \deg(Q_{d1})$ and $\deg(Q_{n2}) \leq \deg(Q_{d2})$, then the resulting numerator degree after addition or multiplication is seen to be the same or less than the denominator degree. Hence, by the fundamental theorem of algebra, there are no more zeros than poles. The sums and the products of stable transfer functions are therefore stable. Now, since matrix addition and multiplication are just a combination of normal addition and multiplication in each element, the sums and products of stable transfer matrices also remain stable.

For a closed-loop system, *internal stability* is the property that the four transfer matrices

$$\begin{aligned} &K(I - P_{yu}K)^{-1}P_{yu} \\ &K(I - P_{yu}K)^{-1} \\ &(I - P_{yu}K)^{-1}P_{yu} \\ &(I - P_{yu}K)^{-1} \end{aligned} \tag{2.9}$$

be stable. Here, actuator and sensor noises are assumed to be added to the actuator inputs and the noiseless sensed outputs respectively. They are also assumed to be included in w, while u and y are assumed to be included in z, so that the four matrices in (2.9) appear in H. In this way, if the system is controllable and observable, internal stability is equivalent to the stability of H. Finally, if the closed-loop system is internally stable, K is said to stabilise the plant P.

3

Controller Design

In this chapter, we discuss controller design via convex optimisation in general terms. For a restricted set of systems and constraints on system behaviour, the controller design problem may be cast as a convex optimisation problem. This enables us to use efficient methods to solve the problem numerically. We will establish in this chapter under which circumstances a convex optimisation problem may be formulated, and how this is accomplished. The aim for the remainder of the thesis is to apply the contents of this chapter to the contents of the previous one.

An overview and some background to the subject is given in Section 3.1. Section 3.2 then goes on to establish the foundation on which the remainder of the chapter will depend, discussing constraints in a more formal manner. Section 3.3 may be regarded as a continuation of Section 3.2 and introduces a fundamental restriction on the constraints, namely convexity. In Section 3.4, the design procedure is outlined and the necessary theory is established. The material in this chapter is based on [Boyd et al., 1990].

3.1 Overview

There are many ways to go about controller design. The general objective is to find a controller K such that the dynamics of a given plant is modified to behave in a desired way. This desired behaviour is often expressed as a set of *design specifications* on, for instance, the performance and robustness of the system. Controllers may be subjected to constraints as well, e.g. on their structure or order.

Often, heuristic methods are employed to this end. However, the designer might be unable to find a suitable controller which satisfies the demands. A consequence of using various rules of thumb is that it is impossible to conclude whether the design method/designer is at fault or whether there is in fact no controller to be found. Conversely, even if a controller were to be obtained, other more efficient choices may have been overlooked. In order to address these and other shortcomings, it is reasonable to minimise some appropriate quantity under said constraints. An optimisation scheme for which only local minima are guaranteed would not be

acceptable, however, as one cannot know for certain if other approaches would have been more successful.

In this chapter, we outline a design method based on [Boyd et al., 1990] in which certain systems and constraints result in a particular type of optimisation problem for which there exist special numerical algorithms. These algorithms are special in the sense that convergence to a global minimum is guaranteed. The kind of optimisation problem for which this can be accomplished is called *a convex optimisation problem*. We state this problem formally.

THE CONVEX OPTIMISATION PROBLEM

$$\min_{x \in \mathscr{V}} f(x) \tag{C}$$

for some $f : \mathscr{V} \to \mathbb{R}$ and $\mathscr{V} \subseteq \mathbb{R}^J$ which are both convex.

Convexity for sets and functions is defined in Section 3.3. Here, f is referred to as the *objective function* and $\mathscr{V}$ as the *constraint set*.

In general, (C) will not yield an analytical solution and must be solved numerically. Note that (C) may still be amenable to the numerical treatment described above even if the assumptions are relaxed somewhat. For instance, f may be quasiconvex, see [Boyd and Barratt, 1991].

In order to systematically arrive at such a problem formulation, the system and the demands on system behaviour must themselves be subjected to constraints. In particular, we require that the system be LTI and that the constraints on the system be convex. In other words, given an LTI plant and a set of convex design specifications, the controller design problem can be cast as a convex optimisation problem [Boyd and Barratt, 1991].

Once (C) has been reached, there are many numerical methods dedicated to the task of solving it, see [Boyd and Barratt, 1991]. We will not concern ourselves with this aspect of controller design, however, and will instead regard the controller design problem as solved upon reaching (C).

As for the design procedure itself, it relies fundamentally on the inclusion of one particular design specification, namely achievability and internal stability. The set of closed-loop transfer matrices which satisfy this design specification may then be generated by the set of stable $n_u \times n_y$ transfer matrices, as is shown in Section 3.4. This set is readily restricted to a finite-dimensional set which may in turn be parameterised by a vector $x \in \mathbb{R}^J$.

From this point on it is assumed that a controller is connected to the system and that both system and controller are LTI. For a review of concepts in systems theory for which we will have use in the following, consult Section 2.3.

3.2 Design Specifications

We begin by considering various requirements on the system behaviour in a more formal setting. A *design specification* $\mathscr{D}$ is a statement about the closed-loop transfer matrix. In the following, it is assumed that all constraints on the system may be expressed in terms of the closed-loop transfer matrix. This is not unreasonable, as our main concern lies with the behaviour of the system *after* a controller has been connected to modify it in a manner which pleases us. Recall further that the regulated outputs z were defined in Section 2.3 in such a way that we were given the possibility to include the transfer functions of interest in H. In light of this assumption, all desires or requirements we might wish to impose on the system should correspond to a design specification.

Suppose now we are given a set of design specifications $\{\mathscr{D}_i\}$ indexed by i (not necessarily a number in this context). Our main objective at this point is to determine if there exists a closed-loop transfer matrix which satisfies all $\mathscr{D}_i$ and therefore behaves in a desired way. Further, if several such matrices exist, we would also like to find the best performing one in some appropriate sense. Only if such a matrix exists do we attempt to retrieve the corresponding controller which achieves the closed-loop system.

In order to confront this problem systematically, it must first be restated in a more direct form. To this end, let $\mathscr{H}$ denote the set of all $n_z \times n_w$ transfer matrices. Given an element of $\mathscr{H}$, by definition $\mathscr{D}_i$ should be either true or false for each i. Collecting all $H \in \mathscr{H}$ for which $\mathscr{D}_i$ is true into a set $\mathscr{H}_i \subseteq \mathscr{H}$, i.e.

$$\mathscr{H}_i = \{H \in \mathscr{H} \mid \mathscr{D}_i \text{ is true}\}$$

it is clear that given a $\mathscr{D}_i$, $\mathscr{H}_i$ is well-defined. Conversely, an arbitrary subset $\mathscr{H}_i$ of $\mathscr{H}$ defines a $\mathscr{D}_i$ through the statement $H \in \mathscr{H}_i$. In other words, there is a one-to-one correspondence between $\mathscr{H}_i$ and $\mathscr{D}_i$. When referring to a design specification then, we also refer to its corresponding subset in $\mathscr{H}$.

We may combine several design specifications into one. Demanding that $\{\mathscr{D}_i\}$ be simultaneously fulfilled is a design specification on its own – call it $\mathscr{D}_*$ – as it says something about H. As such, there is a corresponding subset $\mathscr{H}_* \subseteq \mathscr{H}$. Since $\mathscr{D}_*$ requires the simultaneous satisfaction of all $\mathscr{D}_i$, it demands that $H \in \mathscr{H}_i$ for all i. This in turn corresponds to an intersection in $\mathscr{H}$, i.e.

$$\mathscr{H}_* = \bigcap_i \mathscr{H}_i \tag{3.1}$$

We are now in a position to restate our main objective more directly as deciding whether $\mathscr{H}_*$ is empty or not. If $\mathscr{H}_*$ is non-empty, we say that $\mathscr{D}_i$ are achievable; if empty, they are said to be unachievable.

Design Objectives and Limits of Performance

Some desires on the behaviour of the system are not necessarily constraints and as such are not intended to be strictly observed. We will refer to these as *design objectives*. As an example of a design objective, consider keeping the highest magnitude of some transfer function over some interval as low as possible. In some contexts, this is desirable, and the lower the better. More generally, design objectives often have this structure: a transfer matrix is associated with some number $\alpha \in \mathbb{R}$, and the lower α is the better. In the following, we shall assume design objectives appear on this form.

What is actually discussed above is a function $\phi : \mathscr{H} \to \mathbb{R}$. Given a design objective defined by ϕ, we have a direct way of comparing the performance of two transfer matrices $\bar{H}, \tilde{H} \in \mathscr{H}$. If $\phi(\bar{H}) \leq \phi(\tilde{H})$, then $\bar{H}$ *performs better* than $\tilde{H}$ and hence is more desirable. Now, confronted by a set of design specifications and a design objective, the aim is to determine whether $\mathscr{H}_*$ in (3.1) is empty or not. If non-empty, we would further like to find an $H^0 \in \mathscr{H}_*$ such that $\phi(H^0) \leq \phi(H)$ for all $H \in \mathscr{H}_*$. The corresponding controller which achieves this H^0 – assuming there is such a controller – may be considered an optimal controller and a solution to the controller design problem.

There are often several conflicting design objectives at once. This means that lowering all of them at the same time may be impossible. Conversely, settling for high values is unnecessary if lower values overall may be attained. It is in this sense that we speak of the best achievable performance, or *limit of performance*, for several design objectives. This is best done in the framework of design specifications.

Given a function ϕ, for every $\alpha \in \mathbb{R}$ we define the design specification $\mathscr{D}_{\phi,\alpha}$ such that

$$\mathscr{H}_{\phi,\alpha} = \{H \in \mathscr{H} \mid \phi(H) \leq \alpha\}$$

Clearly, $\mathscr{H}_{\phi,\alpha_1} \subseteq \mathscr{H}_{\phi,\alpha_2}$ if $\alpha_1 \leq \alpha_2$, since $\phi(H) \leq \alpha_1 \leq \alpha_2$. In other words, reducing α corresponds to restricting $\mathscr{D}_{\phi,\alpha}$. In order to emphasise the connection between design specifications and design objectives, we will represent the latter as a set of design specifications $\{\mathscr{D}_{\phi,\alpha}\}$ parameterised by the real number α. Lowering/minimising a design objective $\{\mathscr{D}_{\phi,\alpha}\}$ means finding the lowest α such that $\mathscr{D}_{\phi,\alpha}$ is still achievable.

There are many ways in which to handle several competing design objectives. In this thesis, we compare two design objectives at most. In such cases, the best achievable performance is most conveniently represented in graphical form. To this end, suppose two design objectives $\{\mathscr{D}_{a,\alpha}\}$ and $\{\mathscr{D}_{b,\beta}\}$ are given. Each number pair $(\alpha_0, \beta_0) \in \mathbb{R}^2$ corresponds to the combined design specification that $\mathscr{D}_{a,\alpha_0}$ and $\mathscr{D}_{b,\beta_0}$ both be demanded. If this combined design specification is achievable, i.e. there is an $H \in \mathscr{H}$ for which $\mathscr{D}_{a,\alpha_0}$ and $\mathscr{D}_{b,\beta_0}$ are both satisfied, let the corresponding point (α_0, β_0) be shaded; otherwise, let it be white. Figure 3.1 illustrates how such a result might look. The boundary curve separating the white area from the shaded area is called a *tradeoff curve* and represents the limit of performance. Moving along it

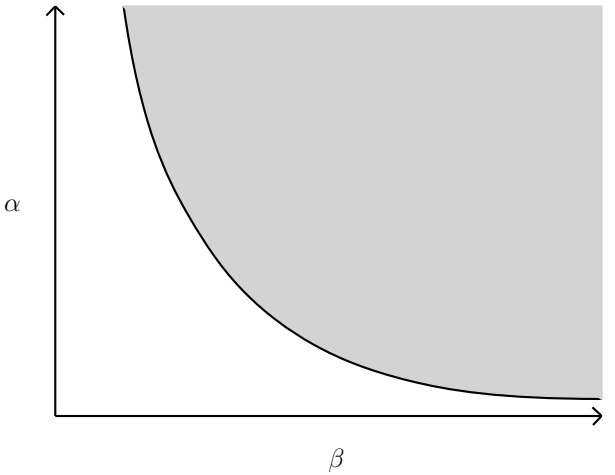

Figure 3.1 This curve represents a comparison between two conflicting design objectives. Each point in the shaded zone corresponds to an achievable design specification.

shows what must be surrendered in one design objective in order for the other to be lowered and vice versa. Similarly, proceeding from the top-right towards the tradeoff curve suggests a lowering of the two design objectives while still maintaining achievability: we improve the performance to the limit of achievability.

We now describe the strategy used in Chapter 5 for computing tradeoff curves. To this end, consider the related problem P of minimising $\{\mathscr{D}_{a,\alpha}\}$ under the constraint $\mathscr{D}_{b,\beta_0}$ for some fixed β_0. Let the minimum be given by α_0. This means that there exists a $H^0 \in \mathscr{H}$ which satisfies both $\mathscr{D}_{a,\alpha_0}$ and $\mathscr{D}_{b,\beta_0}$. This H^0 therefore also satisfies $\mathscr{D}_{a,\alpha}$ and $\mathscr{D}_{b,\beta_0}$ for $\alpha \geq \alpha_0$: by definition, $\mathscr{D}_{a,\alpha}$ is less tight than $\mathscr{D}_{a,\alpha_0}$. All points (α,β_0) for $\alpha \geq \alpha_0$ on the vertical line above (α_0,β_0) should therefore be shaded. Conversely, assume there is an $\alpha < \alpha_0$ such that both $\mathscr{D}_{a,\alpha}$ and $\mathscr{D}_{b,\beta_0}$ are achievable. Then the minimum of problem P is at most α and so cannot be α_0, a contradiction. All points (α,β_0) for $\alpha < \alpha_0$ on the vertical line below (α_0,β_0) should therefore be white. But this means that (α_0,β_0) lies exactly on the tradeoff curve. Solving problem P repeatedly for various β_0 now generates the tradeoff curve.

3.3 Convexity

The set $\mathscr{H}$ is a vector space under ordinary matrix addition and multiplication by scalar [Boyd et al., 1990]. It is therefore meaningful to discuss the geometry of its subsets. In particular, consider the concepts of infinite lines and line segments.

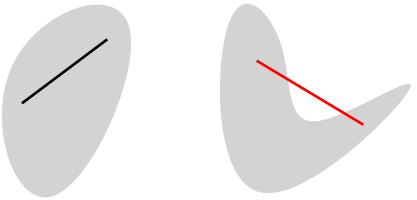

Figure 3.2 The convexity of sets illustrated in $\mathbb{R}^2$. The right set is not convex, since there is at least one pair of elements for which at least one convex combination is not contained in the set.

Take for instance the set $L = \{\lambda v + u \mid \lambda \in \mathbb{R}\}$ where $v, u \in \mathbb{R}^2$ are constants. If the elements of L were to be marked in the plane as λ ranges over the real numbers, we would obtain an infinite line through u. However, if instead v and u were abstract vectors in an arbitrary vector space, L would still be well-defined. This is because the vector space, by definition, is closed under addition and multiplication by scalar. It is therefore meaningful to speak about lines even in vector spaces.

Certain set geometries are associated with theoretical or computational benefits. Convexity is one such example. A subset $\mathscr{C}$ of a vector space is said to be *convex* if for any pair of vectors $u, v \in \mathscr{C}$ it is true that $\lambda u + (1-\lambda)v \in \mathscr{C}$ for all real numbers $\lambda \in [0,1]$. The new element $\lambda u + (1-\lambda)v$ is referred to as a *convex combination* of u and v. In essence, then, a set is convex if for any two of its elements, the entire line segment between them is contained in the set. This is illustrated in figure 3.2.

We now consider a useful property of convex sets concerning their combinations. For quick referencing, we gather it in the lemma below.

LEMMA 3.1 The intersection of convex sets is itself convex.

Proof Take any two elements of the intersection and consider the resulting convex combinations. Belonging to the intersection, the pair must belong to every set separately. Because these sets are all convex by assumption, all convex combinations will be contained in each set separately. The convex combinations must therefore belong to the intersection, which thus is convex. □

We also define convexity for real-valued functions defined over convex subsets $\mathscr{C}$ in arbitrary vector spaces. A function $f : \mathscr{C} \to \mathbb{R}$ is said to be convex if for any pair $u, v \in \mathscr{C}$, it is true that $f(\lambda u + (1-\lambda)v) \leq \lambda f(u) + (1-\lambda) f(v)$ for all real numbers $\lambda \in [0,1]$.

Recall now that for each design specification, there is a way of assigning a set of transfer matrices and vice versa. Thus, it is reasonable to extend the notion of convexity such that it also encompasses design specifications. We therefore say that a design specification $\mathscr{D}_i$ is convex if the corresponding set $\mathscr{H}_i \subseteq \mathscr{H}$ is convex. Given a set of convex design specifications $\{\mathscr{D}_i\}$, by Lemma 3.1 the equivalent design specification $\mathscr{D}_*$ defined by (3.1) is also convex.

Note that convexity has been defined for three separate entities: sets, functions and design specifications. We make the analogous definitions for *affinity*. This essentially corresponds to extending $\lambda \in [0,1]$ to $\lambda \in \mathbb{R}$ and exchanging $\leq$ for $=$ in the function definition. In particular, affinity implies convexity. Note that for functions, it is meaningful to extend the definition of affinity beyond real-valued functions.

We now turn to some important design specifications and show that they are convex. The performance criteria discussed below are relevant for the informal design goals listed in Section 2.1. They are also quite central on their own, independently of the context of this thesis. This means that many design goals likely fit into the general framework presented below.

We begin by considering bounds in the frequency domain.

THEOREM 3.1
Let the design specification $\mathscr{D}_{freq}$ be the statement that the magnitude of $H_{ij}(i\omega)$ be bounded above by some function $h : I \to \mathbb{R}^+$ over some frequency range $I \subseteq \mathbb{R}^+$. Then $\mathscr{D}_{freq}$ is convex.

Proof Let such a function h be given. We proceed by showing that the corresponding subset

$$\mathscr{H}_{freq} = \{H \in \mathscr{H} \mid |H_{ij}(i\omega)| \leq h(\omega),\ \omega \in I\}$$

is convex. To this end, let $\tilde{H}$ and $\bar{H}$ denote two elements of $\mathscr{H}_{freq}$. Since $\tilde{H}, \bar{H} \in \mathscr{H}_{freq}$ it is true that

$$\begin{aligned} |\tilde{H}_{ij}(i\omega)| &\leq h(\omega) \\ |\bar{H}_{ij}(i\omega)| &\leq h(\omega) \end{aligned} \tag{3.2}$$

for all $\omega \in I$.

Consider now the convex combination $H = \lambda\tilde{H} + (1-\lambda)\bar{H}$ for some $\lambda \in [0,1]$. By definition of addition and scalar multiplication for matrices, $H_{ij} = \lambda\tilde{H}_{ij} + (1-\lambda)\bar{H}_{ij}$. Taking the absolute value and exploiting the triangle inequality for complex numbers, we get

$$|H_{ij}(i\omega)| = |\lambda\tilde{H}_{ij}(i\omega) + (1-\lambda)\bar{H}_{ij}(i\omega)| \leq |\lambda||\tilde{H}_{ij}(i\omega)| + |(1-\lambda)||\bar{H}_{ij}(i\omega)|$$

Because $\lambda \in [0,1] \Leftrightarrow \lambda \geq 0$ and $1-\lambda \geq 0$, the absolute value bars may come off so that $|\lambda| = \lambda$ and $|1-\lambda| = 1-\lambda$. We substitute the inequalities (3.2) to obtain

$$\lambda|\tilde{H}_{ij}(i\omega)| + (1-\lambda)|\bar{H}_{ij}(i\omega)| \leq \lambda h(\omega) + (1-\lambda)h(\omega)$$

for all $\omega \in I$, where the cancellation of λ terms finally yields

$$|H_{ij}(i\omega)| \leq h(\omega)$$

This means that $H \in \mathscr{H}_{freq}$ for all $\lambda \in [0,1]$. Any convex combination of elements in $\mathscr{H}_{freq}$ must therefore belong to $\mathscr{H}_{freq}$. This shows that $\mathscr{H}_{freq}$, and hence $\mathscr{D}_{freq}$, is convex. □

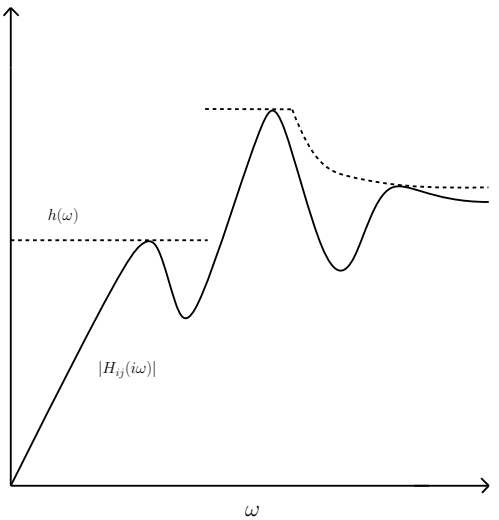

Figure 3.3 The magnitude of a transfer function bounded above.

The design specification $\mathscr{D}_{freq}$ is illustrated in figure 3.3.

We now turn to restrictions in the time domain. Often, one proceeds directly to the output signal to impose constraints on its shape, e.g. upper bounds for suppressing overshoots.

THEOREM 3.2
Let the design specification $\mathscr{D}_{time}$ be the statement that the response of H_{ij} to some scalar input w be bounded above and below by the functions $b_u : I \to \mathbb{R}$ and $b_l : I \to \mathbb{R}$ respectively over some time interval $I \subseteq \mathbb{R}^+$. Then $\mathscr{D}_{time}$ is convex.

Proof Let such functions b_u and b_l be given. As before, we show that the corresponding subset

$$\mathscr{H}_{time} = \{H \in \mathscr{H} \mid \ b_l(t) \leq \mathscr{L}^{-1}(H_{ij}W)(t) \leq b_u(t),\ t \in I\}$$

is convex.

Take two elements $\tilde{H}$ and $\bar{H}$ from $\mathscr{H}_{time}$. Since $\tilde{H}, \bar{H} \in \mathscr{H}_{freq}$ it is true that

$$\begin{aligned} b_l(t) &\leq \mathscr{L}^{-1}(\tilde{H}_{ij}W)(t) \leq b_u(t) \\ b_l(t) &\leq \mathscr{L}^{-1}(\bar{H}_{ij}W)(t) \leq b_u(t) \end{aligned} \tag{3.3}$$

for all $t \in I$.

Consider again the convex combination $H = \lambda\tilde{H} + (1-\lambda)\bar{H}$ for some $\lambda \in [0,1]$. As before, we have $H_{ij} = \lambda\tilde{H}_{ij} + (1-\lambda)\bar{H}_{ij}$. Multiplying by W and taking the inverse Laplace transform, we get

$$\mathscr{L}^{-1}(H_{ij}W)(t) = \mathscr{L}^{-1}((\lambda\tilde{H}_{ij} + (1-\lambda)\bar{H}_{ij})W)(t)$$

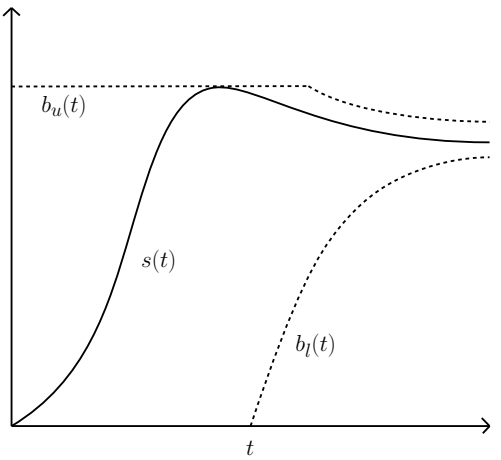

Figure 3.4 The step response $s(t)$ of a transfer function bounded above and below.

By the linearity property of $\mathscr{L}^{-1}$ inherited from the integral, the above expression may be further broken down to

$$\mathscr{L}^{-1}(\lambda\tilde{H}_{ij}W + (1-\lambda)\bar{H}_{ij}W)(t) = \lambda\mathscr{L}^{-1}(\tilde{H}_{ij}W)(t) + (1-\lambda)\mathscr{L}^{-1}(\bar{H}_{ij}W)(t)$$

Invoking the inequalities (3.3) and noting that $\lambda \in [0,1]$ then yields

$$\lambda\mathscr{L}^{-1}(\tilde{H}_{ij}W)(t) + (1-\lambda)\mathscr{L}^{-1}(\bar{H}_{ij}W)(t) \leq \lambda b_u(t) + (1-\lambda)b_u(t) = b_u(t)$$

and

$$\lambda\mathscr{L}^{-1}(\tilde{H}_{ij}W)(t) + (1-\lambda)\mathscr{L}^{-1}(\bar{H}_{ij}W)(t) \geq \lambda b_l(t) + (1-\lambda)b_l(t) = b_l(t)$$

for all $t \in I$. In other words,

$$b_l(t) \leq \mathscr{L}^{-1}(H_{ij}W)(t) \leq b_u(t)$$

and so, by definition of $\mathscr{H}_{time}$, $H \in \mathscr{H}_{time}$ for all $\lambda \in [0,1]$. Any convex combination of elements in $\mathscr{H}_{time}$ must therefore belong to $\mathscr{H}_{time}$. This shows that $\mathscr{H}_{time}$, and hence $\mathscr{D}_{time}$, is convex. □

The design specification $\mathscr{D}_{time}$ is illustrated for a step response in figure 3.4.

Transfer Matrix Achieved by Stabilising Controller

We now turn to the most important design specification of this chapter. It is fundamental not only from a design perspective, but also for the design method outlined in Section 3.4. It is in fact a combination of two crucial constraints on the closed-loop transfer matrix. We discuss them separately.

The vector space $\mathscr{H}$ generally consists of all kinds of matrices. Even if we were to find a closed-loop transfer matrix representing desirable dynamics, the question remains whether there is any controller for us to close the loop with that could possibly achieve such dynamics. Therefore, most $H \in \mathscr{H}$ would likely be useless, however favourable the dynamics represented by them. In other words, we would like to demand of H that it appear on the form $H = P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw}$ for some $n_u \times n_y$ transfer matrix K, see (2.8). This design specification $\mathscr{D}_{ach}$ corresponds to the subset

$$\mathscr{H}_{ach} = \{H \in \mathscr{H} \mid H = P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw} \text{ for some } K = K(s)\}$$

If $H \in \mathscr{H}_{ach}$, we say that H is achieved by some controller K.

There is also the issue of stability, a standard demand on a closed-loop system. It is desirable for such a system to be internally stable, as defined in Section 2.3. We wish to restrict $\mathscr{D}_{ach}$ and make the additional demand that H be internally stable. Thus, let $\mathscr{D}_{stab}$ denote the design specification that H should be achieved by some *stablising* controller K. In order for $\mathscr{D}_{stab}$ to be well-defined, the four transfer matrices in (2.9) must somehow be connected to H, so that $\mathscr{D}_{stab}$ may be expressed on H only. This is the case if the exogenous inputs are augmented by the actuator and sensor noise, and the regulated outputs are augmented by the actuator inputs and sensed outputs. Due to the structure imposed on H by $\mathscr{D}_{ach}$, the four tranfer matrices in (2.9) will now appear as blocks in H. Note that this inclusion of signals is quite consistent with the signal definitions in Section 2.3, since these are either signals which affect the system or signals in which we are interested. $\mathscr{D}_{stab}$ is now well-defined and corresponds to a subset $\mathscr{H}_{stab}$ of $\mathscr{H}$

$$\mathscr{H}_{stab} = \{H \in \mathscr{H} \mid H = P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw} \text{ for some stabilising } K = K(s)\}$$

We end this discussion by noting that more than just convexity holds for $\mathscr{D}_{stab}$.

THEOREM 3.3
$\mathscr{D}_{stab}$ is affine.

Proof The result appears in [Boyd et al., 1990]. As there is a lot of algebra involved, we omit the proof for convenience. The basic idea is simple enough, however. The affine combination of two elements in $\mathscr{H}_{stab}$ is formed. One then proceeds to rewrite the combination on the form (2.8), identifying an expression for the corresponding controller. With this at hand, the four transfer matrices (2.9) may be evaluated and shown to be stable by the stability of their constituents. The affine combinations therefore belong to $\mathscr{H}_{stab}$ and the result follows. □

Non-Convex Design Specifications

There are some important design specifications which are non-convex. We mention briefly two such cases which both concern controllers. Design specifications formulated through H can reach the controller via (2.8).

Consider the design specification that the controller be decentralised. This means that only y_i affects u_i, and so K is diagonal. This design specification is not convex.

Suppose instead we require the order of a controller, i.e. the largest degree of any polynomial found in K, to be at most some positive integer. This design specification is not convex either. In both cases, a simple counterexample suffices to disprove convexity, see [Boyd et al., 1990]. The demand that the controller be passive is likewise non-convex, as far as the author is aware.

Violations of either of these constraints are of enormous practical importance. The first means in essence that the controller input/output interconnection may be much more complex than what may be realisable in practice. As for the second, a controller with too great an order might not even be implementable. The fact that the constraints are not convex means that an optimal controller matrix may be dense and of high order, in which case it might be completely useless in practice.

3.4 The Design Procedure

In this section, we assume that we are given the design specification $\mathscr{D}_{stab}$ along with a separate set of convex design specifications $\{\mathscr{D}_i\}$ and a design objective $\{\mathscr{D}_{\phi,\alpha}\}$. These define our controller design problem according to Section 3.2. The objective is to arrive at the corresponding problem formulation (C). At this point, efficient numerical algorithms for solving the problem become available and so (C) is solved in principle.

Let $\mathscr{D}_{\mathbb{C}stab}$ denote the design specification that all design specifications of the set $\{\mathscr{D}_i\}$ be simultaneously fulfilled. The corresponding subset in $\mathscr{H}$ is therefore

$$\mathscr{H}_{\mathbb{C}stab} = \bigcap_i \mathscr{H}_i$$

By Lemma 3.1, $\mathscr{D}_{\mathbb{C}stab}$ is convex. Our demands on the system behaviour are now gathered concisely in the equivalent design specification $\mathscr{D}_*$, the combination of $\mathscr{D}_{stab}$ and $\mathscr{D}_{\mathbb{C}stab}$. In other words, the transfer matrices which satisfy our design specifications – if indeed there are any that do – are contained in

$$\mathscr{H}_* = \mathscr{H}_{stab} \cap \mathscr{H}_{\mathbb{C}stab} \tag{3.4}$$

cf. (3.1). Another use of Lemma 3.1 shows that this set is convex.

$\mathscr{H}_*$ is infinite-dimensional in general. In order to exploit the numerical tools at our disposal, the problem must be made finite-dimensional. This means we must truncate $\mathscr{H}_*$. There are many different ways in which to accomplish this, some more appropriate than others. For the particular design method outlined in this section, we will search for a finite-dimensional restriction inside $\mathscr{H}_{stab}$. In order to do this, however, $\mathscr{H}_{stab}$ must first be restated in a different, more useful form. Recall how

$\mathscr{H}_{ach}$ by definition is

$$\mathscr{H}_{ach} = \{P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw} \mid K\}$$

As it turns out, under certain circumstances, $\mathscr{H}_{stab}$ may be similarly expressed.

THEOREM 3.4
Given an LTI system, assume that P_{yu} is stable. Then

$$\mathscr{H}_{stab} = \{P_{zw} + P_{zu}RP_{yw} \mid R \text{ is stable}\} \tag{3.5}$$

Additionally, given such a stable R, the controller K which achieves the corresponding H is given by

$$K = (I + RP_{yu})^{-1}R \tag{3.6}$$

Proof Consider the definition of $\mathscr{H}_{stab}$ in the previous section. An equivalent but more explicit description is offered by

$$\mathscr{H}_{stab} = \left\{ P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw} \middle| \begin{matrix} K(I - P_{yu}K)^{-1}P_{yu} \\ K(I - P_{yu}K)^{-1} \\ (I - P_{yu}K)^{-1}P_{yu} \\ (I - P_{yu}K)^{-1} \end{matrix} \text{ stable for some } K \right\} \tag{3.7}$$

We now make the transformation

$$R = K(I - P_{yu}K)^{-1} \tag{3.8}$$

The expression (3.8) for R may now be inserted directly into the upper two expressions in (3.7) to yield RP_{yu} and R respectively. As for the remaining two,

$$\begin{aligned} I &= I \underbrace{-P_{yu}K + P_{yu}K}_{0} = (I - P_{yu}K) + P_{yu}K \underbrace{\Big((I - P_{yu}K)^{-1}(I - P_{yu}K)\Big)}_{I} \\ &= \Big(I + P_{yu}\underbrace{K(I - P_{yu}K)^{-1}}_{R}\Big)(I - P_{yu}K) = (I + P_{yu}R)(I - P_{yu}K) \end{aligned}$$

Multiplying by $(I - P_{yu}K)^{-1}$ from the right, we have $(I - P_{yu}K)^{-1} = I + P_{yu}R$. This accounts for the fourth expression in (3.7). Another multiplication from the right by P_{yu} accounts for the third. Altogether, because (3.8) suggests a one-to-one correspondence between R and K (see below), (3.7) may be rewritten in simpler terms according to

$$\mathscr{H}_{stab} = \left\{ P_{zw} + P_{zu}RP_{yw} \middle| \begin{matrix} RP_{yu} \\ R \\ (I + P_{yu}R)P_{yu} \\ I + P_{yu}R \end{matrix} \text{ stable for some } R \right\} \tag{3.9}$$

Suppose now that R is stable. By the assumption that P_{yu} be stable, we may conclude that the four transfer matrices in (3.9) are stable. This is because the sum and the product of two stable transfer matrices are stable, see Section 2.3. Conversely, if the four transfer matrices are stable for some R, then the second one in particular is as well, i.e. R is stable. In other words, when P_{yu} is stable, the condition in (3.9) is equivalent to the condition that R be stable, and so

$$\mathscr{H}_{stab} = \{P_{zw} + P_{zu}RP_{yw} \mid R \text{ is stable}\}$$

As for the second part of the theorem, assume that we are given a stable R. We multiply both sides of (3.8) from the right by $(I - P_{yu}K)$ and solve for K so that

$$K = (I + RP_{yu})^{-1}R$$

Note that this expression also confirms the one-to-one correspondence between R and K, which concludes the proof. Note that the idea of the proof is adopted from [Boyd et al., 1990]. □

The result of Theorem 3.4 tells us that $\mathscr{H}_{stab}$ is generated by sweeping over the set of all stable $n_u \times n_y$ transfer matrices $\mathscr{R}$. Instead of moving through the entire set, however, we restrict ourselves to a finite-dimensional part of it. The consequence is that only part of $\mathscr{H}_{stab}$ is generated, namely

$$\mathscr{H}_{fdstab} = \{P_{zw} + P_{zu}RP_{yw} \mid \text{only some } R \in \mathscr{R}\}$$

so that $\mathscr{H}_{fdstab} \subseteq \mathscr{H}_{stab}$. This means that $\mathscr{H}_{fdstab} \cap \mathscr{H}_{stab} = \mathscr{H}_{fdstab}$, and so the closed-loop transfer matrices which now satisfy our constraints are given by

$$\mathscr{H}_{fd*} = \mathscr{H}_{fdstab} \cap \mathscr{H}_{\mathbb{C}stab}$$

cf. (3.4). In essence, we have added a new design specification corresponding to $\mathscr{H}_{fdstab}$, which is not necessarily convex.

The $H \in \mathscr{H}$ which now satisfy our constraints are contained in $\mathscr{H}_{fd*}$. Since $\mathscr{H}_{fdstab} \subseteq \mathscr{H}_{stab}$, we have $\mathscr{H}_{fd*} \subseteq \mathscr{H}_{*}$. This means that if $\mathscr{H}_{fd*} \neq \varnothing$, we may take an element, express it on the form in (3.5) for some $R \in \mathscr{R}$, and use (3.6) to find the corresponding controller. This controller will then achieve a closed-loop system which is acceptable also with respect to $\mathscr{D}_{*}$. However, if $\mathscr{H}_{fd*} = \varnothing$, we know only that if $\mathscr{H}_{*}$ is non-empty, the elements do not belong to $\mathscr{H}_{fd*}$. This information can be useful in itself, but more can be said if $\mathscr{H}_{fdstab}$ is chosen appropriately.

We would like $\mathscr{H}_{fdstab}$ to take up as much of $\mathscr{H}_{stab}$ as possible in order to reduce the loss incurred by the truncation, thereby closing the gap between $\mathscr{H}_{fdstab}$ and $\mathscr{H}_{stab}$. In other words, we would like to use as much of $\mathscr{R}$ as possible. In order to discuss this more systematically and prepare for matters of convergence, we construct below an entire family of sets $\mathscr{H}_{fdstab,\,n}$ such that $\mathscr{H}_{fdstab,\,n} \subseteq \mathscr{H}_{fdstab,\,n+1}$ for positive integers n.

The Finite-Dimensional Approximation $\mathscr{H}_{fdstab,\,n}$

Consider the finite sequence

$$Q_k(s) = (s+a)^{-k} \qquad k = 1,\ldots,n \tag{3.10}$$

for some fixed complex number a, $\mathrm{Re}(a) > 0$. We refer to Q_k as *basis functions*. Note that the restriction on a means that all elements of the sequence are stable. Also, for the remainder of the section we assume that n is a positive integer.

We now proceed to define a function $\hat{R}_n : \mathbb{R}^{n_u n_y n} \to \mathscr{R}$ such that

$$\hat{R}_n(x) = \sum_{k=1}^{n} X_k(x) Q_k \tag{3.11}$$

for a fixed n. Here, each X_k is an $n_u \times n_y$ matrix defined by

$$X_k(x) = ((X_k)_{ij}) = (x_{(k-1)n_u n_y + (j-1)n_u + i})$$

where $x_{(k-1)n_u n_y+(j-1)n_u+i}$ is the $((k-1)n_u n_y + (j-1)n_u + i)$:th component of $x \in \mathbb{R}^{n_u n_y n}$. As an example, for $n_u = 4$ and $n_y = n = 3$, $X_2(x)Q_2$ becomes

$$X_2(x)Q_2 = \begin{pmatrix} x_{13} & x_{17} & x_{21} \\ x_{14} & x_{18} & x_{22} \\ x_{15} & x_{19} & x_{23} \\ x_{16} & x_{20} & x_{24} \end{pmatrix} (s+a)^{-2} = \begin{pmatrix} x_{13}(s+a)^{-2} & x_{17}(s+a)^{-2} & x_{21}(s+a)^{-2} \\ x_{14}(s+a)^{-2} & x_{18}(s+a)^{-2} & x_{22}(s+a)^{-2} \\ x_{15}(s+a)^{-2} & x_{19}(s+a)^{-2} & x_{23}(s+a)^{-2} \\ x_{16}(s+a)^{-2} & x_{20}(s+a)^{-2} & x_{24}(s+a)^{-2} \end{pmatrix}$$

It is clear from this and (3.10) that each $X_k Q_k$ is a stable transfer matrix for all $x \in \mathbb{R}^{n_u n_y n}$. Therefore, the sum $\hat{R}_n(x)$ is as well, and so $\hat{R}_n$ is well-defined.

A more natural function to define is $\hat{H}_{\mathscr{R}} : \mathscr{R} \to \mathscr{H}$ given by

$$\hat{H}_{\mathscr{R}}(R) = P_{zw} + P_{zu} R P_{yw}$$

Since it is well-defined on the entire $\mathscr{R}$, so is the composition $\hat{H}_n : \mathbb{R}^{n_u n_y n} \to \mathscr{H}$, $\hat{H}_n = \hat{H}_{\mathscr{R}} \circ \hat{R}_n$, i.e.

$$\hat{H}_n(x) = P_{zw} + P_{zu}\Big(\sum_{k=1}^{n} X_k(x) Q_k\Big) P_{yw}$$

Define now $\mathscr{H}_{fdstab,\,n}$ as

$$\mathscr{H}_{fdstab,\,n} = \{\hat{H}_n(x) \mid x \in \mathbb{R}^{n_u n_y n}\}$$

for fixed n. It is clear by the definition that $\mathscr{H}_{fdstab,\,n}$ is finite-dimensional. Because $\hat{R}_n$ is not surjective in general, only some $R \in \mathscr{R}$ may be reached from $\mathbb{R}^{n_u n_y n}$ and thus $\mathscr{H}_{fdstab,\,n} \subseteq \mathscr{H}_{stab}$ for all n. Further, given $\mathscr{H}_{fdstab,\,n}$ and $\mathscr{H}_{fdstab,\,n+1}$, every element of the former may be reached from the latter simply by extending $x \in \mathbb{R}^{n_u n_y n}$ with $n_u n_y$ zeros. This means that $\mathscr{H}_{fdstab,\,n} \subseteq \mathscr{H}_{fdstab,\,n+1}$, as desired.

The Convex Optimisation Problem

The original objective was to determine whether the convex design specifications $\mathscr{D}_{stab}$ and $\{\mathscr{D}_i\}$, i.e. $\mathscr{D}_*$, were achievable or not. If achievable, we also wished to find the best performing closed-loop system according to the design objective $\{\mathscr{D}_{\phi,\alpha}\}$. After having applied the necessary finite-dimensional design specification $\mathscr{D}_{fdstab,\,n}$, the new objective concerns the design specification $\mathscr{D}_{fd*,\,n}$, a restriction of $\mathscr{D}_*$.

Consider now the set $\mathscr{V}_n$ such that

$$\mathscr{V}_n = \{x \in \mathbb{R}^{n_u n_y n} \mid \hat{H}_n(x) \in \mathscr{H}_{\mathbb{C}stab}\}$$

for a fixed integer n and assume that it is non-empty. Thus, there exists an element $\bar{x} \in \mathscr{V}_n$. By definition of $\mathscr{V}_n$, we have $\hat{H}_n(\bar{x}) \in \mathscr{H}_{\mathbb{C}stab}$. Further, since $\mathscr{H}_{fdstab,\,n}$ was defined above to contain any $H \in \mathscr{H}$ for which $H = \hat{H}_n(x)$ for some $x \in \mathbb{R}^{n_u n_y n}$, we must also have $\hat{H}_n(\bar{x}) \in \mathscr{H}_{fdstab,\,n}$. Therefore, since $\mathscr{H}_{fd*,\,n} = \mathscr{H}_{fdstab,\,n} \cap \mathscr{H}_{\mathbb{C}stab}$, it follows that $\hat{H}_n(\bar{x}) \in \mathscr{H}_{fd*,\,n}$ and $\mathscr{H}_{fd*,\,n}$ is non-empty.

Conversely, if $\mathscr{H}_{fd*,\,n}$ is non-empty, then there is an element $H \in \mathscr{H}$ such that $H \in \mathscr{H}_{fdstab,\,n}$ and $H \in \mathscr{H}_{\mathbb{C}stab}$. By definition of $\mathscr{H}_{fdstab,\,n}$, the former implies there must exist some $\bar{x} \in \mathbb{R}^{n_u n_y n}$ such that $H = \hat{H}_n(\bar{x})$. The latter then gives $\hat{H}_n(\bar{x}) = H \in \mathscr{H}_{\mathbb{C}stab}$ and so $\bar{x} \in \mathscr{V}_n$ and $\mathscr{V}_n$ is non-empty.

We have just shown that $\mathscr{V}_n$ is non-empty if and only if $\mathscr{H}_{fd*,\,n}$ is non-empty. The objective of whether $\mathscr{D}_{fd*,\,n}$ is achievable or not is therefore translated to determining whether $\mathscr{V}_n$ is empty or not.

The question is now whether $\mathscr{V}_n$ is convex. In order to answer this, we first proceed to establish a special property of $\hat{H}_n$.

LEMMA 3.2 $\hat{H}_n$ is affine for all n.

Proof Suppose we are given $\bar{x}, \tilde{x} \in \mathbb{R}^{n_u n_y n}$. We form the affine combination

$$x = \lambda \bar{x} + (1-\lambda)\tilde{x}$$

for any $\lambda \in \mathbb{R}$. By addition and scalar multiplication for vectors,

$$\begin{aligned} x &= \lambda(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_{n_u n_y n}) + (1-\lambda)(\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_{n_u n_y n}) \\ &= (\lambda\bar{x}_1, \lambda\bar{x}_2, \ldots, \lambda\bar{x}_{n_u n_y n}) + ((1-\lambda)\tilde{x}_1, (1-\lambda)\tilde{x}_2, \ldots, (1-\lambda)\tilde{x}_{n_u n_y n}) \\ &= (\lambda\bar{x}_1 + (1-\lambda)\tilde{x}_1, \lambda\bar{x}_2 + (1-\lambda)\tilde{x}_2, \ldots, \lambda\bar{x}_{n_u n_y n} + (1-\lambda)\tilde{x}_{n_u n_y n}) \end{aligned}$$

and so

$$x_k = \lambda\bar{x}_k + (1-\lambda)\tilde{x}_k \tag{3.12}$$

Consider now the action of $\hat{R}_n$ on the affine combination. By definition of X_k, $(X_k)_{ij} = x_{(k-1)n_u n_y + (j-1)n_u + i}$. With $l(i,j,k) = (k-1)n_u n_y + (j-1)n_u + i$, inserting (3.12) yields

$$(X_k)_{ij} = x_l = \lambda\bar{x}_l + (1-\lambda)\tilde{x}_l$$

With this expression inside every element of X_k, we may separate X_k into two matrices with elements $\lambda\bar{x}_l$ and $(1-\lambda)\tilde{x}_l$, namely $X_k(\lambda\bar{x})$ and $X_k((1-\lambda)\tilde{x})$. Further, since every element has λ and $1-\lambda$ in common respectively, we may factor them out of the matrices so that (3.12) becomes

$$\begin{aligned}\hat{R}_n(x) &= \sum_{k=1}^{n} X_k(x)Q_k = \sum_{k=1}^{n} \big(\lambda X_k(\bar{x}) + (1-\lambda)X_k(\tilde{x})\big)Q_k \\ &= \lambda \sum_{k=1}^{n} X_k(\bar{x})Q_k + (1-\lambda)\sum_{k=1}^{n} X_k(\tilde{x})Q_k = \lambda\hat{R}_n(\bar{x}) + (1-\lambda)\hat{R}_n(\tilde{x})\end{aligned}$$

This shows that $\hat{R}_n$ is affine for all n.

Similarly, for any two $\bar{R},\tilde{R} \in \mathscr{R}$,

$$\begin{aligned}\hat{H}_{\mathscr{R}}(\lambda\bar{R}+(1-\lambda)\tilde{R}) &= P_{zw} + P_{zu}(\lambda\bar{R}+(1-\lambda)\tilde{R})P_{yw} \\ &= \underbrace{\lambda P_{zw} - \lambda P_{zw}}_{0} + P_{zw} + \lambda P_{zu}\bar{R}P_{yw} + (1-\lambda)P_{zu}\tilde{R}P_{yw} \\ &= \lambda(P_{zw} + P_{zu}\bar{R}P_{yw}) + (1-\lambda)(P_{zw} + P_{zu}\tilde{R}P_{yw}) \\ &= \lambda\hat{H}_{\mathscr{R}}(\bar{R}) + (1-\lambda)\hat{H}_{\mathscr{R}}(\tilde{R})\end{aligned}$$

which shows that $\hat{H}_{\mathscr{R}}$ is affine as well.

Finally, consider the composition $\hat{H}_n$. Since both parts out of which it is made are affine, we have

$$\begin{aligned}\hat{H}_n(x) &= \hat{H}_{\mathscr{R}}(\hat{R}_n(\lambda\bar{x}+(1-\lambda)\tilde{x})) = \hat{H}_{\mathscr{R}}(\lambda\hat{R}_n(\bar{x}) + (1-\lambda)\hat{R}_n(\tilde{x})) \\ &= \lambda\hat{H}_{\mathscr{R}}(\hat{R}_n(\bar{x})) + (1-\lambda)\hat{H}_{\mathscr{R}}(\hat{R}_n(\tilde{x})) = \lambda\hat{H}_n(\bar{x}) + (1-\lambda)\hat{H}_n(\tilde{x})\end{aligned}$$

for all $\bar{x},\tilde{x} \in \mathbb{R}^{n_u n_y n}$, all $\lambda \in \mathbb{R}$ and all n. This proves that $\hat{H}_n$ is affine for all n. □

With Lemma 3.2, $\mathscr{H}_{fdstab,\,n}$ is easily seen to be convex (and even affine).

The next theorem justifies our overall restriction to convex design specifications.

THEOREM 3.5

$$\mathscr{V}_n = \{x \in \mathbb{R}^{n_u n_y n} \mid \hat{H}_n(x) \in \mathscr{H}_{\mathbb{C}stab}\}$$

is convex for all n.

Proof Take two elements $\bar{x},\tilde{x} \in \mathscr{V}_n$ and consider the convex combination $x = \lambda\bar{x} + (1-\lambda)\tilde{x}$, $\lambda \in [0,1]$. By Lemma 3.2, we have

$$\hat{H}_n(x) = \hat{H}_n(\lambda\bar{x}+(1-\lambda)\tilde{x}) = \lambda\hat{H}_n(\bar{x}) + (1-\lambda)\hat{H}_n(\tilde{x})$$

Now, since $\bar{x}, \tilde{x} \in \mathscr{V}_n$, by definition of $\mathscr{V}_n$ we have $\hat{H}_n(\bar{x}), \hat{H}_n(\tilde{x}) \in \mathscr{H}_{\mathbb{C}_{stab}}$. Exploiting the convexity of $\mathscr{H}_{\mathbb{C}_{stab}}$ by Lemma 3.1, the convex combination $\lambda \hat{H}_n(\bar{x}) + (1 - \lambda)\hat{H}_n(\tilde{x})$ must therefore also belong to $\mathscr{H}_{\mathbb{C}_{stab}}$, i.e.

$$\hat{H}_n(x) = \lambda \hat{H}_n(\bar{x}) + (1 - \lambda)\hat{H}_n(\tilde{x}) \in \mathscr{H}_{\mathbb{C}_{stab}}$$

But by definition of $\mathscr{V}_n$, $\hat{H}_n(x) \in \mathscr{H}_{\mathbb{C}_{stab}}$ means $x \in \mathscr{V}_n$. This shows that the convex combination of any two elements of $\mathscr{V}_n$ also belongs to $\mathscr{V}_n$. $\mathscr{V}_n$ is therefore convex.□

We now proceed to the second objective and assume that $\mathscr{H}_{fd*,n}$ is non-empty. Consider the design objective $\{\mathscr{D}_{\phi,\alpha}\}$ and its associated function ϕ. As assumed in Section 3.2, a lower value of $\phi(H)$ for some $H \in \mathscr{H}$ is considered better. To find the best value over $\mathscr{H}_{fd*,n}$, we may consider the function $f : \mathbb{R}^{n_u n_y n} \to \mathbb{R}$ such that $f = \phi \circ \hat{H}_n$ and assume that there is an $x_0 \in \mathscr{V}_n$ (it is non-empty since $\mathscr{H}_{fd*,n}$ is) such that $f(x_0) \leq f(x)$ for all $x \in \mathscr{V}_n$. Then, with $H^0 = \hat{H}_n(x_0)$ and $H = \hat{H}_n(x)$, we have

$$\phi(H^0) = f(x_0) \leq f(x) = \phi(H)$$

Now, by definition of $\mathscr{V}_n$, we have $\mathscr{H}_{fd*,n} = \hat{H}_n(\mathscr{V}_n)$, and so sweeping through all $x \in \mathscr{V}_n$ implies sweeping through all $H \in \mathscr{H}_{fd*,n}$. Therefore, $\phi(H^0) \leq \phi(H)$ for all $H \in \mathscr{H}_{fd*,n}$ and so $H^0 = \hat{H}_n(x_0)$ is the optimal performer over $\mathscr{H}_{fd*,n}$. In other words, the function $\hat{H}_n$ preserves the minimum attained over $\mathscr{V}_n$.

Altogether, the controller design problem defined by $\mathscr{D}_{fd*,n}$ and $\{\mathscr{D}_{\phi,\alpha}\}$ may be formulated as a convex optimisation problem

$$\min_{x \in \mathscr{V}_n} \phi(\hat{H}_n(x))$$

assuming that ϕ is convex (the composition will then be convex due to $\hat{H}_n$ being affine). $\hat{H}_n$ may be used to find the corresponding optimal performing closed-loop system, and (3.6) may then be used to obtain the optimal controller which achieves such a system. Note that softwares such as CVX also examine achievability when confronted with (C). In practice, then, the above formulation solves both of the objectives stated in Section 3.2.

4

Vibration Control in Buildings

In this chapter, we examine the possibility of designing controllers for vibration suppression in buildings using the theory of Chapter 3. In particular, for a building modelled by the mass chain model, we formalise the informal design goals listed in Section 2.1 and establish when exactly it is possible to employ the design procedure outlined in Section 3.4. Alongside this, we also discuss passive control.

The aim of Section 4.1 is to integrate the building model from Section 2.2 with the framework presented in Section 2.3. In Section 4.2, we proceed to exploit the machinery developed in Chapter 3 in the context of vibration control. Note that the system we refer to from this point on is the mass chain model, and that all the related notation and assumptions of Chapter 2 are observed throughout this chapter.

4.1 The Plant

We begin this section by considering the various signals associated with the system.

Input and Output Signals

The input signals will consist of both actuator inputs u and exogenous inputs w, as detailed in Section 2.3. Beginning with the latter, the most obvious choice for a signal affecting the system, unrelated to the controller, is the ground movement x_0. In light of the definition of internal stability, we should also include actuator and sensor noise in w. As we will see in Section 4.2, however, this turns out to be a formality we need not observe, given that we are assuming a noiseless environment. We therefore set $w = x_0$ and let the exogenous input be a scalar signal.

As for the actuator inputs, we assume that there is a controller device installed between each floor as in Section 2.2. Their influence on the building enters via the opposite forces they exert on two neighbouring floors. For the sake of convenience, we refer directly to these forces as actuator inputs. Specifically, we denote by u_i

the force that affects m_i as caused by the i:th controller device; u is the column vector which contains these scalar signals u_i, $1 \leq i \leq N$. Finally, we assume that the controller is LTI.

The output signals are similarly partitioned into two categories: sensed outputs y and regulated outputs z. The latter includes the intermass displacement $\delta_i = x_i - x_{i-1}$ and the actuator signals u_i. The reason for this is that when we formalise the informal goals of Section 2.1 as design specifications, we must do so through the closed-loop transfer matrix, see Section 3.2. To this end, let $z_i = \delta_i$ and $z_{N+i} = u_i$ for $1 \leq i \leq N$. We now lump these scalar signals into the column vector z. Note that u contains N entries whereas z contains $2N$.

Finally, given our interest in the intermass displacement, it is reasonable to measure this quantity as well, so that the controller may have access to this information when issuing commands. It is likely also a more practical unit to measure than, say, the individual displacements x_i of each floor. We therefore set $y_i = \delta_i$, $1 \leq i \leq N$, and collect these scalar signals into the column vector y.

The System

We now turn to the system itself. The aim of this section is to connect the above signals on the form (2.5) and (2.6). In order to accomplish this, we first derive the equations of motion for our system.

Recall from Chapter 2 that the building is modelled by the mass chain model. As such, each floor is a mass point limited to one-dimensional movement. Further, as a consequence of the assumptions in Section 2.2, there are only three kinds of forces acting on the masses: spring F_{k_i}, damper F_{d_i} and actuator u_i forces. These were defined in Section 2.2.

With Newton's second law dictating that the force sum be proportional to the acceleration of the body, the above suggests the following N equations.

$$\begin{cases} m_i\ddot{x}_i = F_{k_i} - F_{k_{i+1}} + F_{d_i} - F_{d_{i+1}} + u_i - u_{i+1}, & 1 \leq i < N \\ m_N\ddot{x}_N = F_{k_N} + F_{d_N} + u_N \end{cases} \tag{4.1}$$

Note that as there are no mechanical devices above the top floor, m_N is subjected to half as many forces.

We would now like to be more specific about the impact of the passive elements. To this end, we invoke (2.1) and (2.2). The related discussion in Chapter 2 suggests the following way of rewriting (4.1).

$$\begin{cases} m_i\ddot{x}_i = -k_i(x_i - x_{i-1}) + k_{i+1}(x_{i+1} - x_i) - d_i(\dot{x}_i - \dot{x}_{i-1}), & 1 \leq i < N \\ \qquad\quad + d_{i+1}(\dot{x}_{i+1} - \dot{x}_i) + u_i - u_{i+1} \\ m_N\ddot{x}_N = -k_N(x_N - x_{N-1}) - d_N(\dot{x}_N - \dot{x}_{N-1}) + u_N \end{cases}$$

Rearranging the terms slightly and dividing by m_i and m_N, we finally obtain the

equations of motion for the system.

$$\begin{cases} \ddot{x}_i = \dfrac{k_i}{m_i}x_{i-1} + \dfrac{d_i}{m_i}\dot{x}_{i-1} - \dfrac{k_i+k_{i+1}}{m_i}x_i - \dfrac{d_i+d_{i+1}}{m_i}\dot{x}_i, \quad 1 \le i < N \\ \qquad + \dfrac{k_{i+1}}{m_i}x_{i+1} + \dfrac{d_{i+1}}{m_i}\dot{x}_{i+1} + \dfrac{1}{m_i}u_i - \dfrac{1}{m_i}u_{i+1} \\ \ddot{x}_N = \dfrac{k_N}{m_N}x_{N-1} + \dfrac{d_N}{m_N}\dot{x}_{N-1} - \dfrac{k_N}{m_N}x_N - \dfrac{d_N}{m_N}\dot{x}_N + \dfrac{1}{m_N}u_N \end{cases} \tag{4.2}$$

As we can see from (4.2), the plant is LTI.

We now proceed by making the transformation

$$\bar{x}_{2i-1} = x_i,\ \bar{x}_{2i} = \dot{x}_i, \quad 1 \le i \le N \tag{4.3}$$

The derivatives of these new functions depend on whether the subscript is odd or even. For odd subscripts, we have simply $\dot{\bar{x}}_{2i-1} = \dot{x}_i = \bar{x}_{2i}$, using both expressions in (4.3). For even subscripts, we have $\dot{\bar{x}}_{2i} = \ddot{x}_i$ and therefore have to invoke (4.2). The result is

$$\begin{cases} \dot{\bar{x}}_{2i-1} = \bar{x}_{2i} \\ \dot{\bar{x}}_{2i} = \dfrac{k_i}{m_i}\bar{x}_{2i-3} + \dfrac{d_i}{m_i}\bar{x}_{2i-2} - \dfrac{k_i+k_{i+1}}{m_i}\bar{x}_{2i-1} - \dfrac{d_i+d_{i+1}}{m_i}\bar{x}_{2i} \\ \qquad + \dfrac{k_{i+1}}{m_i}\bar{x}_{2i+1} + \dfrac{d_{i+1}}{m_i}\bar{x}_{2i+2} + \dfrac{1}{m_i}u_i - \dfrac{1}{m_i}u_{i+1} \end{cases} \tag{4.4}$$

for $1 \le i \le N$, except for the two special cases $\dot{\bar{x}}_2$ and $\dot{\bar{x}}_{2N}$. These are collected separately below.

$$\begin{cases} \dot{\bar{x}}_2 = -\dfrac{k_1+k_2}{m_1}\bar{x}_1 - \dfrac{d_1+d_2}{m_1}\bar{x}_2 + \dfrac{k_2}{m_1}\bar{x}_3 + \dfrac{d_2}{m_1}\bar{x}_4 \\ \qquad + \dfrac{k_1}{m_1}w + \dfrac{d_1}{m_1}\dot{w} + \dfrac{1}{m_1}u_1 - \dfrac{1}{m_1}u_2 \\ \dot{\bar{x}}_{2N} = \dfrac{k_N}{m_N}\bar{x}_{2N-3} + \dfrac{d_N}{m_N}\bar{x}_{2N-2} - \dfrac{k_N}{m_N}\bar{x}_{2N-1} - \dfrac{d_N}{m_N}\bar{x}_{2N} + \dfrac{1}{m_N}u_N \end{cases} \tag{4.5}$$

Recall that $w = x_0$.

We now apply the Laplace transform to the expressions in (4.4) and (4.5). Since k_i, d_i and m_i are constants, by the linearity of the transform, the transformed expressions will barely have changed. The exception is the occasional derivative turning into a multiplication by s. Note that $\mathscr{L}(\frac{k_1}{m_1}w + \frac{d_1}{m_1}\dot{w}) = (\frac{k_1}{m_1} + \frac{d_1}{m_1}s)W$.

The terms in each expression are now readily structured into a sum of matrix products according to

$$s\bar{X} = A\bar{X} + B_w W + B_u U \tag{4.6}$$

Here, the newly defined states have been collected into a $2N \times 1$ column vector $\bar{x}$ with Laplace transform $\bar{X}$. A and B_u are constant matrices of size $2N \times 2N$ and $2N \times N$ respectively, whereas B_w is of size $2N \times 1$ and depends on s.

(4.6) is easily solved for $\bar{X}$ to yield

$$\bar{X} = (sI - A)^{-1} B_w W + (sI - A)^{-1} B_u U \tag{4.7}$$

In order to form the sensed outputs Y, we must extract the relevant parts out of $\bar{X}$. Simple multiplication by a certain matrix C achieves this: recall that the i:th row of Y is defined as the intermass displacement $X_i - X_{i-1} = \bar{X}_{2i-1} - \bar{X}_{2i-3}$. There is an exception in the first row, namely $X_1 - X_0 = \bar{X}_1 - W$, and so we must add a term $C_w W$ so that

$$Y = C\bar{X} + C_w W \tag{4.8}$$

Substituting (4.7) into (4.8), we get

$$Y = \underbrace{\left(C(sI - A)^{-1} B_w + C_w\right)}_{P_{yw}} W + \underbrace{C(sI - A)^{-1} B_u}_{P_{yu}} U \tag{4.9}$$

from which we may identify P_{yw} and P_{yu}.

As for the regulated outputs Z, recall that it was defined to contain both the intermass displacements, now collected in Y, and the actuator inputs U. We therefore have

$$Z = \begin{pmatrix} Y \\ U \end{pmatrix} = \underbrace{\begin{pmatrix} P_{yw} \\ 0 \end{pmatrix}}_{P_{zw}} W + \underbrace{\begin{pmatrix} P_{yu} \\ I \end{pmatrix}}_{P_{zu}} U \tag{4.10}$$

With (4.9) and (4.10) at hand, we finally have the corresponding expressions for (2.6) and (2.7), and the four important transfer matrices which describe the plant are given by

$$\begin{cases} P_{yw} = C(sI - A)^{-1} B_w + C_w \\ P_{yu} = C(sI - A)^{-1} B_u \\ P_{zw} = \begin{pmatrix} P_{yw} \\ 0 \end{pmatrix} \\ P_{zu} = \begin{pmatrix} P_{yu} \\ I \end{pmatrix} \end{cases} \tag{4.11}$$

From the way in which the important matrices A, B_u, B_w, C, C_w were defined in (4.6) and (4.8) based on (4.5), it is clear that the A matrix especially will have a somewhat complex structure. We therefore postpone displaying them until Chapter 5, at which point they will be shown explicitly for the $N = 5$ case.

4.2 Vibration Control

We now close the loop and consider the closed-loop transfer matrix which follows from (2.8). The aim of this section is to formalise the informal design goals listed in Section 2.1 and establish under what circumstances the controller design problem may be formulated on the form (C) according to Section 3.4. This controller design method may be contrasted with other alternative means of control, such as passive control, see e.g. [Yamamoto, 2016]. We touch upon the subject of passive control at the end of this section.

As stated in the list in Section 2.1, we would like to reduce $|\delta_i(t)|$ and $|u_i(t)|$ for $1 \leq i \leq N$ in some appropriate sense. In order to show that constraints in both the frequency and time domain are supported by the theory, we involve both kinds. This is also reasonable from a control perspective: different kinds of regulation are achieved depending on where the constraint is applied.

We begin with constraints in the frequency domain. Recall that in the spectrum of an earthquake, the highest activity in displacement tends to occur in the $0-10$ rad/s frequency band. Now, the connection between x_0 and δ_i is represented by the closed-loop transfer function $H_{\delta_i x_0} = H_{z_i w}$. One way to suppress the influence of low-frequency vibrations is to limit the low-frequency part of $|H_{\delta_i x_0}(i\omega)|$ and demand that it not exceed a given number $0 < \alpha_{\delta f} < 1$ over some frequency range $I = [0, \omega_0]$. According to Theorem 3.1, this is a convex design specification (take $h(\omega) = \alpha_{\delta f}$). Demanding that this hold for all $1 \leq i \leq N$ is also a convex design specification by Lemma 3.1. The corresponding subset in $\mathscr{H}$ is given by

$$\mathscr{H}_{\delta f, \alpha_{\delta f}} = \{H \in \mathscr{H} \mid \ |H_{z_i w}(i\omega)| \leq \alpha_{\delta f},\ 1 \leq i \leq N,\ \omega \in [0, \omega_0]\}$$

We now introduce the function $\phi_{\delta f} : \mathscr{H} \to \mathbb{R}$ given by

$$\phi_{\delta f}(H) = \max_{1 \leq i \leq N} \sup_{\omega \in [0, \omega_0]} |H_{\delta_i x_0}(i\omega)|$$

to the effect that $\mathscr{H}_{\delta f, \alpha_{\delta f}}$ may be expressed more concisely as

$$\mathscr{H}_{\delta f, \alpha_{\delta f}} = \{H \in \mathscr{H} \mid \ \phi_{\delta f}(H) \leq \alpha_{\delta f}\} \tag{4.12}$$

We assume that the supremum taken in $\phi_{\delta f}$ exists.

Similarly, reducing actuator effort could involve the same type of design specification over different closed-loop transfer matrices, namely $H_{u_i x_0} = H_{z_{N+i} w}$ for all $1 \leq i \leq N$: these form the connection between x_0 and u_i. The corresponding subset in $\mathscr{H}$ is given by

$$\mathscr{H}_{uf, \alpha_{uf}} = \{H \in \mathscr{H} \mid \ |H_{z_{N+i} w}(i\omega)| \leq \alpha_{uf},\ 1 \leq i \leq N,\ \omega \in [0, \omega_0]\} \tag{4.13}$$

for some constant number α_{uf}. By defining an analogous function ϕ_{uf}, (4.13) can be restated concisely just as (4.12). Note also that by varying $\alpha_{\delta f}$ and α_{uf}, we generate different design specifications.

The exact manner in which design specifications on the form (4.12) and (4.13) affect the regulated outputs in the time domain is not clear in practice. Reducing $|H_{ij}(i\omega)|$ does not necessarily imply that big transient spikes are suppressed. It is not difficult to imagine that such violent pulse-like behaviour may in fact be quite devastating. Too large $|\delta_i(t)|$, even for a brief moment, could fracture the supporting columns, and too large $|u_i(t)|$ could damage the columns or the controller devices and render the latter inactive for the remainder of the earthquake, see Section 2.1. We therefore proceed more directly and apply constraints in the time domain as well.

The aim of this kind of constraint would be to control big transient spikes after short but particularly intense quakes. According to Section 2.1, such a quake can be approximated by an impulse so that $W = 1$. For the intermass displacement, we thus demand that the magnitude of the impulse response $|\mathscr{L}^{-1}(H_{\delta_i x_0})(t)|$ not exceed a given number $0 < \alpha_{\delta t}$ over the time interval $I = [0, t_0]$. By Theorem 3.2, this is a convex design specification (take $b_u(t) = \alpha_{\delta t}$ and $b_l(t) = -\alpha_{\delta t}$). Further, demanding that this hold for all $1 \leq i \leq N$ is once more also a convex design specification by Lemma 3.1. The corresponding subset in $\mathscr{H}$ is given by

$$\mathscr{H}_{\delta t, \alpha_{\delta t}} = \{H \in \mathscr{H} \mid \ |\mathscr{L}^{-1}(H_{z_i w})(t)| \leq \alpha_{\delta t},\ 1 \leq i \leq N,\ t \in [0, t_0]\}$$

The analogous demand for actuator effort is given by

$$\mathscr{H}_{ut, \alpha_{ut}} = \{H \in \mathscr{H} \mid \ |\mathscr{L}^{-1}(H_{z_{N+i} w})(t)| \leq \alpha_{ut},\ 1 \leq i \leq N,\ t \in [0, t_0]\}$$

We now define two functions $\phi_{\delta t}, \phi_{ut} : \mathscr{H} \to \mathbb{R}$ such that

$$\phi_{\delta t}(H) = \max_{1 \leq i \leq N} \sup_{t \in [0, t_0]} |\mathscr{L}^{-1}(H_{\delta_i x_0})(t)|$$

and

$$\phi_{ut}(H) = \max_{1 \leq i \leq N} \sup_{t \in [0, t_0]} |\mathscr{L}^{-1}(H_{u_i x_0})(t)|$$

$\mathscr{H}_{\delta t, \alpha_{\delta t}}$ and $\mathscr{H}_{ut, \alpha_{ut}}$ are now expressed concisely as

$$\mathscr{H}_{\delta t, \alpha_{\delta t}} = \{H \in \mathscr{H} \mid \ \phi_{\delta t}(H) \leq \alpha_{\delta t}\} \tag{4.14}$$

and

$$\mathscr{H}_{ut, \alpha_{ut}} = \{H \in \mathscr{H} \mid \ \phi_{ut}(H) \leq \alpha_{ut}\} \tag{4.15}$$

Note that $\phi_{\delta t}$, $\phi_{\delta f}$, ϕ_{ut} and ϕ_{uf} are convex functions.

Finally, as our last design specification, we will demand $\mathscr{D}_{stab}$. Not only is internal stability an important performance criterion in itself, but the entire design procedure outlined in Section 3.4 rests on its inclusion through Theorem 3.4. For the implications of that theorem to hold, however, P_{yu} must be stable. The next theorem verifies that this is the case for our system regardless of parameter choice, as long as there is stiffness and damping.

THEOREM 4.1
P_{yu} is stable for all $N \geq 1$ and all material constants $m_i, k_i, d_i > 0$ for $1 \leq i \leq N$.

Proof Consider the definition of P_{yu} from (4.11)

$$P_{yu} = C(sI - A)^{-1}B_u$$

As seen above, the matrices B_u and C are constant. As for $(sI-A)^{-1}$, recall that it is given by $\frac{1}{\det(sI-A)}\text{adj}(sI-A)$. The latter matrix, called the adjugate, is simply a collection of determinants of parts of $sI-A$. Since $sI-A$ contains only polynomials (A is constant), so will the adjugate: the determinant of a matrix is merely a particular sum of products of its elements. Therefore, $C\text{adj}(sI-A)B_u$ contains only polynomials as well, and the final multiplication by $\frac{1}{\det(sI-A)}$ to obtain P_{yu} turns every polynomial into a rational polynomial. In other words, if a transfer function element of P_{yu} has a pole, it must be one of the zeros of $\det(sI-A)$.

Consider now the system $\dot{\bar{x}} = A\bar{x}$ obtained by setting $w = 0$ and $u = 0$. Because of the springs and dampers between each floor, there will be a constant energy dissipation and $\bar{x}$ will get arbitrarily close to $\bar{x} = 0$ from any initial state, given enough time. In other words, $\dot{\bar{x}} = A\bar{x}$ is globally asymptotically stable in a Lyapunov sense, and all eigenvalues of A must therefore have negative real part. But the eigenvalues of A are given exactly by the zeros of $\det(sI-A)$, and so, by the above, every pole of the transfer function entries of P_{yu} must have negative real part. Further, the adjugate contains only determinants of parts of $sI-A$, and so the orders will not surpass that of $\det(sI-A)$. Therefore, by the definition of stability for rational transfer matrices, P_{yu} is stable as long as $m_i, k_i, d_i > 0$ for $1 \leq i \leq N$. □

Altogether, combining the design specifications $\mathscr{D}_{stab}$, $\mathscr{D}_{\delta t,\alpha_{\delta t}}$, $\mathscr{D}_{uf,\alpha_{uf}}$, $\mathscr{D}_{ut,\alpha_{ut}}$, $\mathscr{D}_{\delta f,\alpha_{\delta f}}$ with the corresponding design objectives, we may formulate the controller design problem on the form (C) according to Section 3.4 for all number of masses N and every material parameter configuration, as long as $m_i, k_i, d_i > 0$ for $1 \leq i \leq N$.

We close this chapter by commenting briefly on a particular kind of control strategy: passive control. In our case, passive control refers to the inclusion of dampers between floors. As explained in Section 2.2, this addition of external damping manifests as an increase in the damping ratio ξ. By (2.3), F_{d_i} may be separated into one part representing structural damping and another part representing the action of the damper devices, the latter of which is consequently identified as u_i in (4.1). This means that u_i has the form

$$u_i = c_i(\dot{x}_i - \dot{x}_{i-1})$$

for some real constants c_i. Given our choice of sensed outputs (see Section 4.1), $U_i = c_i s(X_i - X_{i-1}) = c_i s Y_i$ and so damper action is equivalent to controller action governed by a particular kind of decentralised controller matrix K such that $K_{ii} = c_i s$. The force exerted by the controller devices should therefore correspond to that exerted by the dampers, i.e. the part of F_{d_i} which is not due to structural damping. It is in this way that passive and active control are compared in the next chapter.

5

The Five-Storey Building

Recall that in Chapter 4, we showed that the design procedure outlined in Section 3.4 could be employed for all chains of N masses and all choices of material parameters with $m_i, k_i, d_i > 0$, given a restricted set of design specifications. In this chapter, we demonstrate the design method outlined in Section 3.4 for a building of reasonable size ($N = 5$) subjected to some of the design specifications formulated in Section 3.3. Various design problems are stated and solved using the method in question to yield optimally performing closed-loop systems. The performance of these optimal systems is then compared to the performance of the corresponding passive system when external damping is added.

In Section 5.1, we describe the system explicitly and formulate some design problems as convex optimisation problems. The solutions to these problems are then reported in Section 5.2. Matlab was used for all computations involved and the convex optimisation problems were solved using CVX, see [CVX]. For a description of the notation used to formulate the design problems, see Section 3.2.

5.1 The N = 5 System

We begin by describing the system. This means we specify the template for the plant supplied in Chapter 4. Setting $N = 5$, we must decide on 15 material constants, namely m_i, k_i, d_i for $1 \leq i \leq 5$. The spring constants are adopted from [Léger and Dussault, 1992], where they had been retrieved experimentally for a specific set of masses, which we also use here. The damping coefficients are then calculated from (2.3). ξ is set to 0.02 for structural damping. The resulting parameter choices are summarised in table 5.1 below.

Given this set of material parameter choices, we may proceed to (4.2) for the equations of motion of the system. From these, the four important transfer matrices P_{zw}, P_{zu}, P_{yw} and P_{yu} may be derived, as shown in Chapter 4. By (4.11), they are

$i=$	1	2	3	4	5
m_i (kg)	10^5	10^5	10^5	10^5	10^5
k_i (MN/m)	245.7	210.6	175.5	140.4	105.3
d_i (kNs/m)	782.09	670.36	558.63	446.91	335.18

Table 5.1 The material parameter choices for the system.

described by the matrices A, B_w, B_u, C, C_w, displayed below for the $N=5$ case.

$$A=\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{k_1+k_2}{m_1} & -\frac{d_1+d_2}{m_1} & \frac{k_2}{m_1} & \frac{d_2}{m_1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{k_2}{m_2} & \frac{d_2}{m_2} & -\frac{k_2+k_3}{m_2} & -\frac{d_2+d_3}{m_2} & \frac{k_3}{m_2} & \frac{d_3}{m_2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{k_3}{m_3} & \frac{d_3}{m_3} & -\frac{k_3+k_4}{m_3} & -\frac{d_3+d_4}{m_3} & \frac{k_4}{m_3} & \frac{d_4}{m_3} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{k_4}{m_4} & \frac{d_4}{m_4} & -\frac{k_4+k_5}{m_4} & -\frac{d_4+d_5}{m_4} & \frac{k_5}{m_4} & \frac{d_5}{m_4} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{k_5}{m_5} & \frac{d_5}{m_5} & -\frac{k_5}{m_5} & -\frac{d_5}{m_5} \end{pmatrix},$$

$$B_w=\begin{pmatrix} 0 \\ \frac{d_1}{m_1}s+\frac{k_1}{m_1} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad B_u=\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{1}{m_1} & -\frac{1}{m_1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{m_2} & -\frac{1}{m_2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{m_3} & -\frac{1}{m_3} & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{m_4} & -\frac{1}{m_4} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{m_5} \end{pmatrix}$$

$$C=\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \end{pmatrix}, \quad C_w=\begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

With the above matrices set, the $N=5$ plant is completely characterised.

Some Controller Design Problems

We now turn to some controller design problems that we formulate as convex optimisation problems. For the notation and terminology used below, see Section 3.2.

A. Establish whether each set of design specifications below is achievable or not. If achievable, minimise the given design objective.

1) $\mathscr{D}_{stab}$ and $\mathscr{D}_{ut,10\,\mathrm{GN}}$ with the design objective $\{\mathscr{D}_{\delta t,\alpha_{\delta t}}\}$
2) $\mathscr{D}_{stab}$, $\mathscr{D}_{ut,10\,\mathrm{GN}}$ and $\mathscr{D}_{\delta t,\alpha_{\delta t}}$ for $\alpha_{\delta t} = 17, 20$ m with the design objective $\{\mathscr{D}_{\delta f,\alpha_{\delta f}}\}$.

B. Establish whether $\mathscr{D}_{stab}$ and $\mathscr{D}_{ut,1\,\mathrm{GN}}$ is achievable or not. If achievable, minimise $\{\mathscr{D}_{\delta f,\alpha_{\delta f}}\}$. Repeat this procedure for $n = 2,4,6,8,10,12,14$ when applying the finite-dimensional constraint $\mathscr{D}_{fdstab,\,n}$.

C. Given $\mathscr{D}_{stab}$, compute the tradeoff curve between

1) $\{\mathscr{D}_{\delta f,\alpha_{\delta f}}\}$ and $\{\mathscr{D}_{ut,\alpha_{ut}}\}$
2) $\{\mathscr{D}_{\delta t,\alpha_{\delta t}}\}$ and $\{\mathscr{D}_{ut,\alpha_{ut}}\}$ for $n = 2,5$.

In each problem, $\xi = 0.02$, $n = 5$, $t_0 = 2$ s and $\omega_0 = 20$ rad/s were chosen, see Section 4.2 for the latter two. The exception is problem B and C2, in the sense that n is allowed to vary. The constant a in the basis functions Q_k was taken as $a = 2$. Note that a problem on the form (C) is assumed to solve the achievability problem in practice, see 3.4. Finally, recall that if the demands are achievable, then there is a closed-loop transfer matrix with good enough behaviour for which (3.6) may be used to compute the controller which achieves it.

We now proceed to formulate the corresponding convex optimisation problem (C) for each problem listed above according to the instructions in Section 3.4.

Problem A By definition of $\mathscr{H}_{\mathbb{C}stab}$, we have

$$\mathscr{H}_{\mathbb{C}stab} = \mathscr{H}_{ut,10\,\mathrm{GN}}$$

for the setup A1. In other words,

$$\mathscr{H}_{\mathbb{C}stab} = \{H \in \mathscr{H} \mid \phi_{ut}(H) \leq 10\,\mathrm{GN}\}$$

With the finite-dimensional approximation $\mathscr{H}_{fdstab,\,5}$ applied, we therefore have $\mathscr{V}_5$ more explicitly as

$$\mathscr{V}_5 = \{x \in \mathbb{R}^{n_u n_y n} \mid \hat{H}_5(x) \in \mathscr{H}_{\mathbb{C}stab}\} = \left\{x \in \mathbb{R}^{125} \,\middle|\, \phi_{ut}(\hat{H}_5(x)) \leq 10\,\mathrm{GN}\right\}$$

As for the design objective $\{\mathscr{D}_{\delta t,\alpha_{\delta t}}\}$, it corresponds to the objective function $f(x) = \phi_{\delta t}(\hat{H}_5(x))$. With $\mathscr{V}_5$ and f well-defined, (C) finally becomes

$$\min_{x \in \mathscr{V}_5} \phi_{\delta t}(\hat{H}_5(x))$$

As for the second setup A2,

$$\mathscr{H}_{\mathbb{C}stab} = \mathscr{H}_{ut,10\,\mathrm{GN}} \cap \mathscr{H}_{\delta t,\alpha_{\delta t}}$$

for $\alpha_{\delta t} = 17,20$ m and so

$$\mathscr{V}_5 = \left\{ x \in \mathbb{R}^{125} \,\middle|\, \begin{array}{r} \phi_{ut}(\hat{H}_5(x)) \leq 10 \text{ GN} \\ \phi_{\delta t}(\hat{H}_5(x)) \leq \alpha_{\delta t} \end{array} \right\}$$

The design objective $\{\mathscr{D}_{\delta f, \alpha_{\delta f}}\}$ corresponds to the objective function $f(x) = \phi_{\delta f}(\hat{H}_5(x))$. Thus, once more we have (C)

$$\min_{x \in \mathscr{V}_5} \phi_{\delta f}(\hat{H}_5(x))$$

Note that the convex optimisation problem is computation-ready only when $\alpha_{\delta t} = 17$ m or $\alpha_{\delta t} = 20$ m is chosen.

Problem B For this problem, consider the design specifications $\mathscr{D}_{stab}$ and $\mathscr{D}_{ut,1\text{ GN}}$ and the design objective $\{\mathscr{D}_{\delta f, \alpha_{\delta f}}\}$. To formulate (C), we proceed precisely as in problem A2. With one less constraint, $\mathscr{H}_{\mathbb{C}stab}$ becomes

$$\mathscr{H}_{\mathbb{C}stab} = \mathscr{H}_{ut,1\text{ GN}}$$

The significant difference compared to A2 is that we now allow n to vary. Thus, $J = n_u n_y n = 5 \cdot 5 \cdot n = 25n$. We therefore have

$$\mathscr{V}_n = \left\{ x \in \mathbb{R}^{25n} \,\middle|\, \phi_{ut}(\hat{H}_n(x)) \leq 1 \text{ GN} \right\}$$

With objective function $f(x) = \phi_{\delta f}(\hat{H}_n(x))$, (C) becomes

$$\min_{x \in \mathscr{V}_n} \phi_{\delta f}(\hat{H}_n(x))$$

Note that for each n, there is an optimisation problem; we pose seven of these by allowing n to range over the even integers up to $n = 14$.

Problem C The setup C1 is handled first. Confronted by two design objectives, we consider instead the related setup $\mathscr{D}_{ut,\alpha_{ut}}$ and $\{\mathscr{D}_{\delta f, \alpha_{\delta f}}\}$. (C) is formed as

$$\min_{x \in \mathscr{V}_5} \phi_{\delta f}(\hat{H}_5(x))$$

for

$$\mathscr{V}_5 = \left\{ x \in \mathbb{R}^{125} \,\middle|\, \phi_{ut}(\hat{H}_5(x)) \leq \alpha_{ut} \right\}$$

Here, a distinct convex optimisation problem is generated for each α_{ut}: we sweep over many α_{ut} and match them with the corresponding minima. By Section 3.2, these pairs will form the tradeoff curve between $\{\mathscr{D}_{\delta f, \alpha_{\delta f}}\}$ and $\{\mathscr{D}_{ut,\alpha_{ut}}\}$.

Similarly, in the C2 case, we consider the related setup $\mathscr{D}_{ut,\alpha_{ut}}$ and $\{\mathscr{D}_{\delta t, \alpha_{\delta t}}\}$, with (C) for $n = 2,5$ formed as

$$\min_{x \in \mathscr{V}_n} \phi_{\delta t}(\hat{H}_n(x))$$

for

$$\mathscr{V}_n = \left\{ x \in \mathbb{R}^{25n} \,\middle|\, \phi_{ut}(\hat{H}_n(x)) \leq \alpha_{ut} \right\}$$

5.2 Results

In this section, we gather the computation results to the problems A, B and C of the previous section. Before doing so, however, we turn briefly to the corresponding passive system and remark on some values against which later results will be compared.

The Passive System

The passive system is obtained by taking $x \in \mathbb{R}^{n_u n_y n}$ as the zero vector, thereby effectively breaking the loop: there is no longer any active component involved. Below, we consider two passive systems in parallel: the default system ($\xi = 0.02$) in which damping stems only from the columns, and the system ($\xi = 0.2$) in which dampers have been added between floors. This external addition of damping is modelled precisely by raising the damping ratio ξ from 0.02 to 0.2, see Section 2.1.

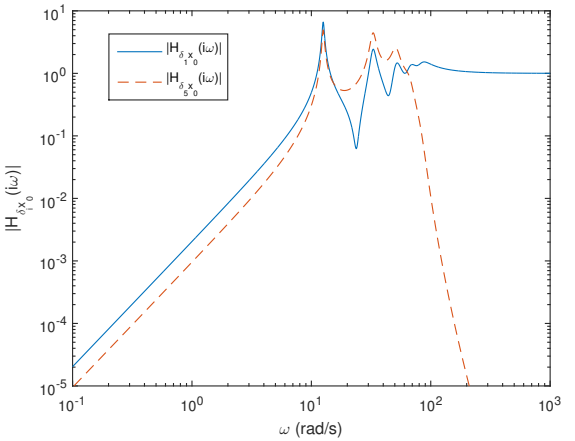

Figure 5.1 The magnitude of $H_{\delta_i x_0}(i\omega)$ for $i = 1$ (blue) and $i = 5$ (dashed orange) when $\xi = 0.02$ in the passive case.

Consider first the greatest value of the frequency response magnitude $|H_{\delta_i x_0}(i\omega)|$ on the interval $[0, \omega_0]$ as $1 \leq i \leq 5$ when ξ is taken as 0.02 and 0.2 respectively. These values are gathered in table 5.2 below. The frequency response magnitudes for $i = 1$ and $i = 5$ are shown in figure 5.1 for the case $\xi = 0.02$. Note that $\xi = 0.2$ means that the damping coefficients d_i in table 5.1 are increased by a factor 10, see (2.3).

Consider now instead the response of the masses and the dampers when subjected to an impulse in ground movement, see Section 4.2. The greatest values of the impulse response $\mathscr{L}^{-1}(H_{\delta_i x_0})(t)$ on the interval $[0, t_0]$ as $1 \leq i \leq 5$ when ξ is

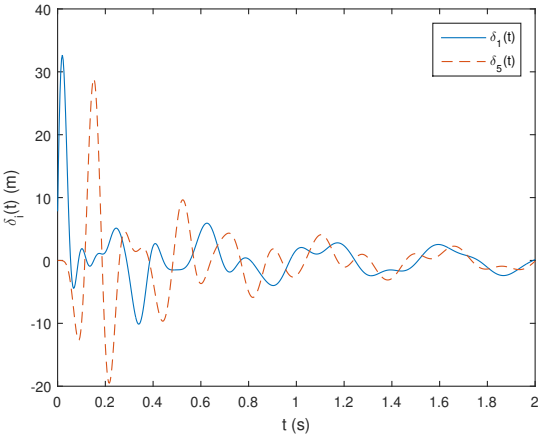

Figure 5.2 The impulse response $\mathscr{L}^{-1}(H_{\delta_i x_0})(t)$ for $i = 1$ (blue) and $i = 5$ (dashed orange) when $\xi = 0.02$ in the passive case.

i	1	2	3	4	5
$\xi = 0.02$	6.6158	7.2211	7.4199	6.8847	4.9614
$\xi = 0.2$	0.6942	0.7413	0.7515	0.6919	0.4966

Table 5.2 The greatest value of $|H_{\delta_i x_0}(i\omega)|$ on the interval $[0, \omega_0]$ for $1 \leq i \leq 5$ when ξ is varied.

taken as 0.02 and 0.2 are collected in table 5.3 below. The impulse responses for $i = 1$ and $i = 5$ are shown in figure 5.2 when $\xi = 0.02$.

In the case of $\xi = 0.2$, the greatest force produced by the added dampers in the time interval $[0, t_0]$ is 85.342 GN. This value, paired together with the greatest value of $|H_{\delta_i x_0}(i\omega)|$ – 0.7515 according to table 5.2 – is marked by a red cross in figure 5.10 for a comparison with the corresponding active system.

i	1	2	3	4	5
$\xi = 0.02$	32.6600 m	25.5540 m	22.0200 m	21.7474 m	28.8980 m
$\xi = 0.2$	78.2087 m	78.2087 m	14.7659 m	7.6747 m	4.7627 m

Table 5.3 The greatest value of $|\mathscr{L}^{-1}(H_{\delta_i x_0})(t)|$ on the interval $[0, t_0]$ for $1 \leq i \leq 5$ when ξ is varied.

Some Convex Optimisation Problems

We now turn to the computation results of the problems A, B and C. The solutions to each convex optimisation problem was computed using CVX, see [CVX Research Inc., 2017], a Matlab-based modelling system for convex optimisation. The computation time associated with various results was obtained using the tic-toc command in Matlab. Furthermore, all attempts to compute the controller matrix K corresponding to the acquired minima failed due to overflow. This was attempted using the Control System Toolbox in Matlab.

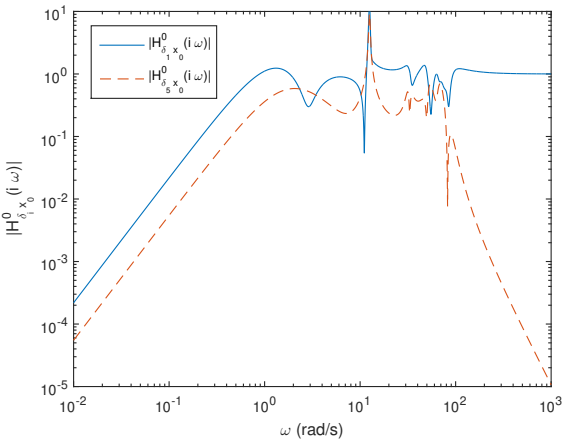

Figure 5.3 The magnitude of $H^0_{\delta_i x_0}(i\omega)$ for $i = 1$ (blue) and $i = 5$ (dashed orange) in problem A1.

Problem A In the first problem A1, the resulting minimum was 17.4666 m. The coordinate x^0 for which this miminum was attained corresponds to the closed-loop transfer matrix $H^0 = \hat{H}_5(x^0)$. The functionals $\phi_{\delta f}$, $\phi_{\delta t}$ and ϕ_{ut} were then evaluated at $H = H^0$ with their values gathered in table 5.4. The magnitude of $H^0_{\delta_i x_0}(i\omega)$ and the impulse response of $H^0_{u_i x_0}$ are plotted in figures 5.3 and 5.4 respectively for $i = 1$ and $i = 5$.

$\phi_{\delta f}(H^0)$	$\phi_{\delta t}(H^0)$	$\phi_{ut}(H^0)$
13.4103	17.4666 m	10 GN

Table 5.4 $\phi_{\delta f}$, $\phi_{\delta t}$ and ϕ_{ut} evaluated at $H = H^0$ for problem A1.

In the case of A2, the design specifications with $\alpha_{\delta t} = 20$ m yielded the minimum 0.1851, whereas those with $\alpha_{\delta t} = 17$ m yielded unachievability. The coor-

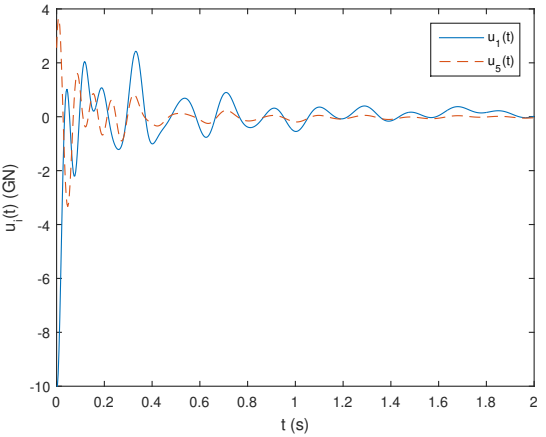

Figure 5.4 The impulse response $\mathscr{L}^{-1}(H^0_{u_i x_0})(t)$ for $i = 1$ (blue) and $i = 5$ (dashed orange) in problem A1.

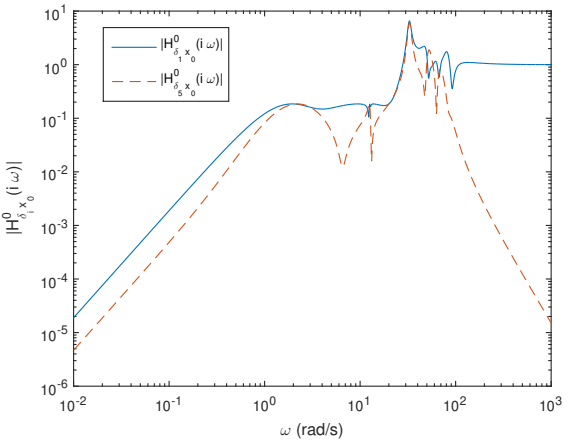

Figure 5.5 The magnitude of $H^0_{\delta_i x_0}(i\omega)$ for $i = 1$ (blue) and $i = 5$ (dashed orange) in problem A2.

dinate for which the minimum 0.1851 was attained once more corresponds to a closed-loop transfer matrix H^0. Table 5.5 collects the values of the functions $\phi_{\delta f}$, $\phi_{\delta t}$ and ϕ_{ut} at $H = H^0$. Further, figures 5.5 and 5.6 show the magnitude of $H^0_{\delta_i x_0}$ and the impulse response of $H^0_{\delta_i x_0}$ respectively for $i = 1$ and $i = 5$.

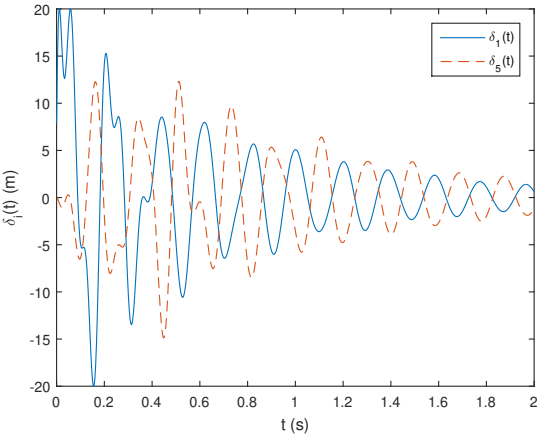

Figure 5.6 The impulse response $\mathscr{L}^{-1}(H^0_{\delta_{i,x_0}})(t)$ for $i = 1$ (blue) and $i = 5$ (dashed orange) in problem A2.

$\phi_{\delta f}(H^0)$	$\phi_{\delta t}(H^0)$	$\phi_{ut}(H^0)$
0.1851	20.0000 m	9.6161 GN

Table 5.5 $\phi_{\delta f}$, $\phi_{\delta t}$ and ϕ_{ut} evaluated at $H = H^0$ for problem A2.

Problem B For this problem, seven convex optimisation problems were solved for $n = 2,4,6,8,10,12,14$. The computation times for the optimisation process ranged from about 1 minute (61 s) in the $n = 2$ case to about 15 minutes (891 s) in the $n = 14$ case. The minima corresponding to each n are gathered in table 5.6 below, and a plot thereof is shown in figure 5.7.

n	2	4	6	8	10	12	14
minimum	0.3276	0.3094	0.2945	0.2818	0.2772	0.2762	0.2762

Table 5.6 The minima for the seven convex optimisation problems in B.

Problem C Consider first the setup C1 for the problem of computing tradeoff curves. 100 points were used in generating the resulting tradeoff curve which is shown in figure 5.8. The computation time was about 5 hours.

As for the setup C2, the two resulting tradeoff curves appear together in figure 5.9. Similarly, the curves consist of 100 points each. The computation times were about 1 hours and 2.5 hours in the $n = 2$ and $n = 5$ case respectively. Note that in figure 5.9, although seemingly overlapping, the orange curve is in fact situated slightly above the blue curve at all time.

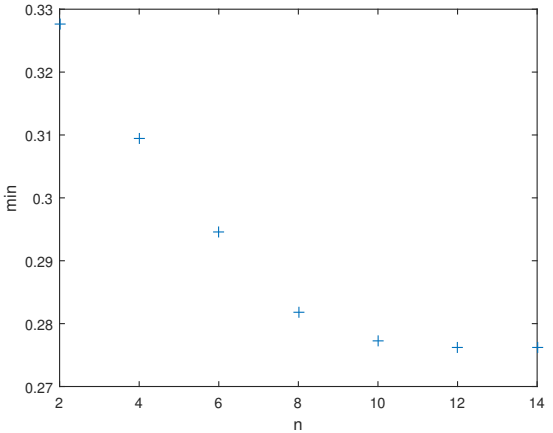

Figure 5.7 The minima for the convex optimisation problems in B.

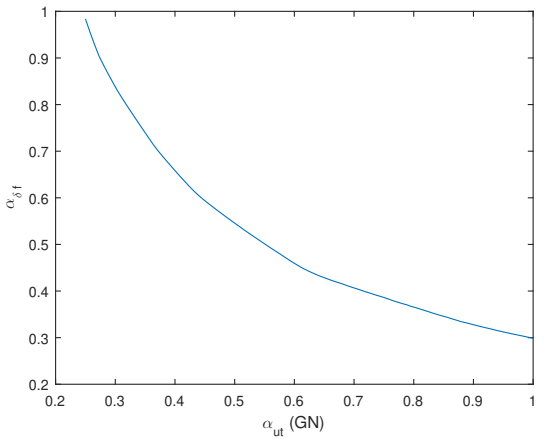

Figure 5.8 The tradeoff curve for $\{\mathscr{D}_{\delta f,\alpha_{\delta f}}\}$ and $\{\mathscr{D}_{ut,\alpha_{ut}}\}$ in problem C1.

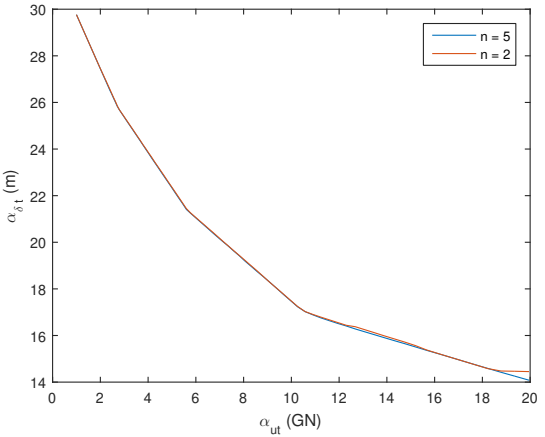

Figure 5.9 The tradeoff curve for $\{\mathscr{D}_{\delta t,\alpha_{\delta t}}\}$ and $\{\mathscr{D}_{ut,\alpha_{ut}}\}$ when $n = 5$ (blue) and $n = 2$ (orange) in problem C2.

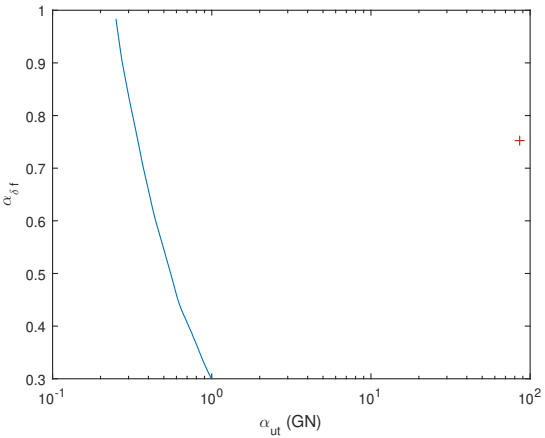

Figure 5.10 The tradeoff curve for $\{\mathscr{D}_{\delta f,\alpha_{\delta f}}\}$ and $\{\mathscr{D}_{ut,\alpha_{ut}}\}$ in figure 5.8. The performance of the corresponding passive system with damping ratio $\xi = 0.2$ is marked by a red cross.

6

Discussion and Conclusion

In this final chapter, we analyse the contents of the previous chapters and conclude the project. In Section 6.1, we perform a critical assessment of the thesis material and make suggestions for future work on the subject. Section 6.2 then summarises the important conclusions reached in the previous section.

6.1 Discussion

We begin our analysis by considering the first major objective of this thesis, namely that of formulating the controller design problem as a convex optimisation problem.

The Controller Design Problem

In Chapter 3 and 4 we showed that given a restricted set of controller design problems for a mass chain system, they could be formulated as convex optimisation problems. This could be accomplished for any number of masses joined together by linear springs and dampers, as long as the spring constants and damping coefficients were all positive. The controller design problems amenable to this procedure are those defined by the convex design specifications presented in Chapter 3. In particular, we may introduce an arbitrary real-valued function as an upper bound to the magnitude of any closed-loop transfer function. Similarly, we may confine the response in the time domain to an arbitrary input signal between two arbitrary real-valued functions. The closed-loop system may also be required to be internally stable and achievable by some controller.

As far as performance criteria are concerned, these are arguably quite exhaustive. However, important design specifications relating to such qualities as robustness and controller structure/order are neglected in the above. The latter was mentioned in Chapter 3, where it was identified to be non-convex. In other words, we cannot demand that the controller be either low ordered or decentralised if we wish to employ the above method for arriving at a convex optimisation problem. The optimal controllers obtained by the method could therefore be impossible to implement in practice. On the other hand, there is the option of resorting to various model

reduction techniques for simplifying the controller, e.g. [Glad and Ljung, 2003]. If this could be accomplished without too great a sacrifice in performance, the design method would find more use. For the purpose of evaluating the best achievable performance, however, the above performance specifications will suffice.

Another important limitation to the design method concerns the plant. The mass chain model is a simple description of a building; they are often represented by more complex models in practice. The mass chain model could therefore be made more complex to increase its availability as an approximation. Now that the model has been shown to be compatible with our design method, verifying compatibility for a more complex variant would be a natural next step. Of course, the plant has to remain LTI for this to be successful.

In such cases of linear approximation, robustness would be valuable in suppressing model perturbations caused by nonlinearities. This is especially true if model reduction were to yield a realisable controller. As it happens, a considerable amount of robustness specifications are in fact convex, see [Boyd et al., 1990]. Robustness specifications relevant to vibration control are therefore likely convex as well. Altogether, it might be interesting for future projects to establish whether relevant robustness specifications are convex or not, to what extent model reduction techniques may be applied to controllers and if the mass chain model can be made more complex while still retaining compatibility with the design method in question.

Finally, note that the internal stability demand does not immediately imply that all closed-loop transfer functions are stable. As explained in Chapter 2, we must further require of the system that it be observable and controllable. Because the main purpose of internal stability is to allow a free parameter representation of $\mathscr{H}_{stab}$ according to Theorem 3.4, we will not verify observability/controllability. We note that if it should not hold, the measuring/controller equipment may be improved, e.g. by measuring the individual mass displacements or even the velocities to the effect that the matrix C changes. Only the matrices B_u and C are affected by such changes and so in general, P_{yu} should still be stable. Note that for this particular plant setup, the proof in Theorem 4.1 may be extended to encompass parts of the remaining three transfer matrices in (4.11) to the effect that H can easily be seen to have all its poles confined to the strict left half-plane.

We now turn to discuss the results in Chapter 5. Recall that we stated some problems A, B and C for a system of five masses. The corresponding convex optimisation problems were successfully formulated and solved, and the optimal closed-loop transfer matrices were obtained. However, the associated controllers could not be retrieved in symbolic form using the Control System Toolbox. This was due to overflow. Studying the formula $K = (1 + RP_{yu})^{-1}R$, however, this should be expected. Even for a modest amount of basis functions, we expect polynomials of very high order. The controller would be useless in practice for this reason alone, and so in this respect, retrieving it is not essential. For the purpose of examining its structure, there may be more efficient ways in which to accomplish this, something an extension of the project could be concerned with.

The Passive System

Consider first the results for the corresponding passive system with damping ratio $\xi = 0.02$. This damping ratio corresponds to structural damping, i.e. no added damping. The behaviour of the transfer functions in figure 5.1 is expected: sending in a low frequency wave, the masses should have an easy time to locate each other, i.e. keeping the magnitude of the intermass displacement low. Additionally, high-frequency ground movement should affect the masses closest to the ground more.

The peaks are also expected. The greatest appears at around 12 rad/s. This is consistent with (2.4), an empirical formula for the natural frequency of the mass chain model. Setting N to 5, (2.4) gives 12.5664... rad/s. We compare the peak values with the second row of table 5.2 and note that this first peak appears to represent the greatest value. In particular, it appears to be present in all five transfer functions. Recall now that the spectrum of the displacement for earthquakes tends to have the highest activity in the 0-10 rad/s frequency band. It is therefore crucial that this particular peak is suppressed.

Adding external damping by increasing the damping ratio to $\xi = 0.2$ appears to achieve just this: the peak is reduced by approximately as much as the damping ratio is increased – a factor 10. This is shown in the third row of table 5.2. The transfer functions are effectively smoothened out.

Returning to the case with structural damping, we now consider the impulse response. The behaviour of the curves in figure 5.2 is expected in the sense that an impulse should produce a violent reaction at the beginning; dampers between the floors dissipate energy and so the oscillations are expected to gradually become smaller. Because m_1 is closer to the ground, we expect its first major peak to occur before that of m_5 as the violent reaction spreads down the mass chain. This is consistent with the behaviour of the curves.

The values themselves do not seem reasonable, however: 30 m in horizontal distance between adjacent floors is not realistic. On the other hand, neither is the ground moving as an impulse. The large values could be a consequence thereof. As explained in Section 2.1, the impulse is a crude approximation of a particular kind of pulse-like ground motion called forward-directivity near-fault ground motion. For more realistic models of such earthquakes, see e.g. [Mavroeidis and Papageorgiou, 2003] or [Kojima and Takewaki, 2015]. One of these could be employed instead of the impulse for more realistic results.

The other anomaly we must address appears in table 5.3. We would expect a higher ξ to better dampen the impulse response. This also appears to be the case for $i = 3,4,5$. However, not only are the values more than twice as large for $i = 1,2$ compared to the case of structural damping (cf. 32.6600 m and 78.2087 m), but they are identical to four decimals. After some closer investigation of the values involved, a possible explanation is as follows. Increasing the damping ratio ξ increases d_1 in particular. This makes the structure between m_1 and the ground more

rigid, and so m_1 should better track the ground impulse, thereby resulting in a larger displacement. This observation is further supported by increasing d_1. Interestingly, it appears as though added external damping, while efficient in quelling oscillations, could likewise lead to a more violent response shortly after an abrupt change in ground displacement.

Regarding the value 78.2087 m, it also appears to be the first value in the data vector corresponding to $t = 0$. For $\mathscr{L}^{-1}(H_{\delta_2 x_0})(t)$, the corresponding value is -78.2087 m (78.2087 m with the absolute value taken). Now, as it happens, the same phenomenon occurs for $\xi = 0.02$. In that case, though, the phenomenon is masked because higher values are reached for $t > 0$. In other words, we seem to have $x_1 - x_0 = -(x_2 - x_1)$ at $t = 0$ in Matlab. But this is reasonable if the ground is assumed to have just completed its impulse. The enormous force exerted on m_1 by the ground leads to instantaneous response. However, m_1 itself does not move as an impulse, and so the force exerted on m_2 is not comparable to that of the ground exerted on m_1. Therefore, the speed built up by m_2 in that flash of a moment around $t = 0$ will be negligible compared to that of m_1, i.e. $x_2(0) = 0$ and so $\delta_1 = -\delta_2$. Note that $t = 0$ refers to the first element in the time vector in Matlab.

Controller Design and Problem A

We consider now the controller design problems formulated in Section 5.1, beginning with problem A1. It appears to have been successfully solved, reducing the largest impulse response peak from 32.6600 m in the passive case (table 5.3) to 17.4666 m (table 5.4), while confining the actuator effort to within 10 GN. Figure 5.4 shows how the greatest force produced by the first controller device reaches 10 GN precisely. This appearance of the curve is likewise reasonable, and the large numbers are accounted for as in the above discussion.

On comparing this closed-loop system with the passive system, however, we see that the overall low-frequency magnitude of the transfer functions has increased, cf. figure 5.1 and 5.3. In particular, the peak at around 12 rad/s has increased too. This bodes ill in conjunction with the low-frequency profile of earthquakes. Hence, we try to reduce at least the peak as much as possible in problem A2 while demanding an intermass displacement of at most 20 m in absolute value, close to the current minimum 17.4666 m.

The result is quite encouraging for the transfer functions of interest: the largest value on the interval $0 - 20$ rad/s for all five transfer functions has been reduced from around 10 (at least) to 0.1851. We see clearly in figure 5.5 how the magnitude of two of the transfer functions stays below this value in the range $0 - 20$ rad/s. Compare this with the corresponding value 0.7515 when external damping was added in the passive case (table 5.2). At the same time, we have successfully kept the impulse responses down to below 20 m, cf. 78.2087 m for the case of added external damping (table 5.3). Figure 5.6 shows how $|\delta_1|$ just barely reaches 20 m several times. Note how the demand to stay below 17 m resulted in unachievability,

as consistent with the result of problem A1.

On the other hand, the oscillations which follow in the wake of the impulse are not damped as well as they are in the passive case for $\xi = 0.2$. As an example of this, consider $|\delta_5|$, which remains below 5 m and 12 m in the passive and active case respectively, cf. table 5.3 and figure 5.6. Altogether, controller action appears to complement damper action even when demanding only design specifications with simple structure. Note that the upper and lower bound functions which define the time domain design specification can be made more complex while still preserving convexity, see Theorem 3.2. This could be useful in addressing the oscillations.

Finally, we observe that the *second* peak in the frequency domain appears to have risen slightly as compared with the passive case, cf. figure 5.1 and 5.5. This is an example of a general phenomenon in which suppressing one part of the magnitude of a transfer function tends to amplify another part. Fortunately, the design specification relating to the frequency domain can be made more complex to account for this while still preserving convexity: the upper bound function h may be customised accordingly, see Theorem 3.1. Comparing the results of A1 to A2, we observe that we do not necessarily have to sacrifice much by applying tighter constraints in order to see dramatic improvements.

Basis Functions and Problem B

In problem B, the same controller design problem is solved for different finite-dimensional approximations. The number of basis functions Q_k included in the approximation is increased from $n = 2$ to $n = 14$ (even numbers only). The resulting minima in table 5.6 are plotted against the number of Q_k in figure 5.7.

The general downward slope of the curve is reasonable: a higher n corresponds to a less tight finite-dimensional approximation, and so there are more closed-loop systems to choose from, including those generated in tighter approximations. As a result, the minimum corresponding to some n_0 must be equal to or lower than the minimum corresponding to any $n < n_0$.

The curve further suggests some form of convergence as n increases. Whether this is due to failed solution attempts for higher n or an underlying mathematical behaviour is difficult to say. Numerical difficulties could be expected on the basis of high-order rational functions like $(s+2)^{-14}$ alone.

At the same time, the convergence behaviour is not wholly unexpected: there are convergence results relating to the sequence Q_k in the one-dimensional case [Boyd and Barratt, 1991]. The free parameter R was constructed so as to allow independent linear combinations of Q_k in each matrix element for a chance at convergence. Recall that $\mathscr{H}_{stab}$ is generated by the set of stable $n_u \times n_y$ transfer matrices. Ideally, we would be able to get arbitrarily close to any such transfer matrix in some appropriate sense by choosing a high enough n. This would mean escaping the inconvenient constraint imposed on the design problem by the necessary finite-dimensional truncation.

There is also the possibility of generating the matrix R in a different manner entirely. In particular, Q_k can be chosen differently, e.g. by changing the constant a in its definition. However, there is also the possibility of opting for a different structure altogether for Q_k than the one suggested in Chapter 3. As an example, the related Laguerre filters could possibly prove to be a more suitable candidate for Q_k, see [Heuberger et al., 2005]. Either way, this topic of convergence and choosing the free parameter R appropriately could benefit from future examination.

Limits of Performance and Problem C

Consider now problem C and the comparison between actuator effort and inter-mass displacement in both the frequency and time domain. The tradeoff curve for $\{\mathscr{D}_{\delta f,\alpha_{\delta f}}\}$ and $\{\mathscr{D}_{ut,\alpha_{ut}}\}$ under the constraint $\mathscr{D}_{stab}$ is shown in figure 5.8. In essence, this curve attempts to approximate the boundary between that which is achievable and not achievable, thereby representing the best achievable performance. For instance, choosing $(\alpha_{\delta f},\alpha_{ut}) = (0.4, 0.4\text{ GN})$, the point is seen to lie below the curve. Hence, there is likely no controller which can achieve a closed-loop system such that $|H_{\delta_i x_0}(i\omega)| \leq 0.4$ for $1 \leq i \leq 5$ and $\omega \in [0,20]$, while simultaneously maintaining internal stability, achievability by a controller and confinement of the actuator effort to within 0.4 GN in the time interval $0-2$ s.

In practice, due to the finite-dimensional approximation, the actual boundary between achievability and non-achievability will lie at or below the one in figure 5.8. However, the discussion of problem B above and C2 below suggests this boundary should be close in some sense to the curve in figure 5.8.

The curve itself behaves in a reasonable way in the sense that it appears to be convex, as it should be. Further, it begins with large values for small α_{ut} while sloping downward for increasing α_{ut}. This represents the sacrifice we have to make in one design specification in order to tighten the other: it is intuitive that more actuator effort results in better peak suppression. By keeping the actuator effort below some critical value in the time domain, we may ensure that no column or controller damage is incurred. In choosing a safe upper bound, the actuator effort would arguably remain low even if the ground were not to move as an impulse. Conversely, knowing that we wish to stay below a certain $\alpha_{\delta f}$, we could reinforce the columns and the controller devices accordingly.

The analogous tradeoff curves between $\{\mathscr{D}_{\delta t,\alpha_{\delta t}}\}$ and $\{\mathscr{D}_{ut,\alpha_{ut}}\}$ shown in figure 5.9 have the same properties as the C1 curve, including convexity. Further, the two curves in figure 5.9 appear to be almost identical. With the discussion of problem B in mind, this suggests that the real tradeoff curve (when the finite-dimensional constraint is lifted) is also very similar. The fact that the blue curve appears to be situated just below the orange curve at all time is a direct consequence of the relation between n and the minima of a problem, see the discussion of problem B above.

Unlike the C1 curve, however, the two curves appear to be less smooth and almost linear over some intervals. The cause for this is not known, but it could

be related to the computation of the curves, see below. Note that the curves were generated by 100 points each, so a lack thereof cannot be the cause.

Another observation is that of the orange curve abruptly becoming horizontal at approximately $\alpha_{ut} = 19$ GN after nearly having overlapped with the blue curve. The natural interpretation is the following. If we were to solve the corresponding problem D of reducing $\{\mathscr{D}_{\delta t,\alpha_{\delta t}}\}$ while demanding only $\mathscr{D}_{stab}$, the minimum m_0 (assuming it exists) would correspond to a closed-loop system with some maximum actuator effort value $\alpha_{ut}^{(0)}$ on the interval $0-2$ s. Now, gradually loosening the restriction on actuator effort for the corresponding convex optimisation problem for the setup C2, we will eventually reach the point when the max value of $\alpha_{ut}^{(0)}$ is allowed. The corresponding minimum at this point must be the same as m_0, or else one of them is not an actual minimum, a contradiction. Proceeding by raising the max value α_{ut} allowed cannot further reduce the minimum, however, for in this case m_0 would no longer be the minimum to problem D, another contradiction. Similarly, the minimum cannot increase (take the solution to problem D). The curve therefore has to be horizontal from this point on. We argue that $\alpha_{ut}^{(0)} \approx 19$ GN.

The fact that the blue curve continues downwards is reasonable, again due to differences in the tightness of the finite-dimensional constraint: the minimum in the $n = 5$ variant of problem D should be equal to or lower than m_0, cf. the discussion of problem B. However, we still expect an abrupt change for high enough α_{ut}.

Altogether, with the exception of the lack of smoothness of the C2 tradeoff curves, the actuator effort and the intermass displacement have been successfully compared in both the frequency and time domain by means of reasonable-looking tradeoff curves. This representation of various limits of performance was one of our main goals according to Chapter 1. As noted there, the strength of these tradeoff curves lies mainly in comparisons with alternative controller design methods. To this end, as a first application of this, we compare directly the method of passive control with the limits of performance of the system.

According to Section 4.2, the two are directly comparable: the controller device corresponds to the added damper device. Figure 5.10 shows the result of such a comparison: it consists of the tradeoff curve of figure 5.8 and a red cross marking the performance of the corresponding passive system. We see that while the dampers exert forces greater than 80 GN to achieve the result in table 5.2, an optimal controller will require at most 0.4 GN for the same kind of performance. In this sense, an optimal controller can be over 200 times more efficient than a damper.

Some Notes on Computation

In general, the computation was prone to instability in the sense that small changes in input value could result in dramatic changes in CVX solution status. For instance, changing an upper bound constraint by a decimal could render CVX completely incapable of solving the problem, returning "Status: Failed" with minimum "NaN" when before it had returned "Status: Solved" along with a number. Changes in the

code, such as expressing as many operations as possible in matrix multiplications, helped remedy this to some extent. All computations performed for the result section had "Status: Solved".

A similar phenomenon occurred when the input values were either too small or too great. In this case, though, the status returned was often "Inaccurate/solved", and the corresponding minima were unreasonable, unlike in the previous case. In practice, then, solving design problems works only for some mid-range inputs.

The choice of basis functions Q_k is yet another factor which comes into play in deciding the quality of computation. The constant a in the definition of Q_k was not chosen arbitrarily, and some tentative tests were run prior to the main ones to decide on an a which would yield stable results. As it turns out, even small differences in the constant a could influence the result considerably. In order to improve computation in general, then, attention should be paid to the particular choice of Q_k. It appears to be well worth exploring alternative options for Q_k, possibly even beyond the structure suggested in this thesis, because the choice affects the results in more than one way, see e.g. the discussion about problem B.

Computation times tended to be excessively long, cf. several hours to compute a tradeoff curve. Prior to our attempts to make the code more efficient, they had been substantially longer. The computation time also increased with the number of basis functions n included in generating the finite-dimensional approximation, which is natural. However, figures 5.7 and 5.9 suggest that n may be kept low for better solution times without much difference in minima obtained. Note that the difference in computation time between adjacent n would likely increase for more complex systems, as the solution vector $x \in \mathbb{R}^{n_u n_y n}$ increases by $n_u \cdot n_y$ elements.

Finally, we remark on the value $\phi_{ut}(H^0) = 9.6161$ GN in table 5.5. Recall that the max actuator effort allowed was 10 GN. This suggests that the minimum could possibly have been lowered slightly by maximising the actuator effort within the allowed limits. This in turn could be confirmed for cases in which two very similar design problems a) and b) with the fixed constraint $\mathscr{D}_{\delta t, \alpha_{\delta t}}$ were solved, as in A2. The problem b), which allowed more actuator effort, resulted in a lower minimum. However, the allowed actuator effort was not fully exploited in the optimal solution, and the solution satisfied the tighter actuator effort constraints of problem a), whose minimum was slightly higher. This implies that the minimum of problem a) was no actual minimum: it should be at most the minimum reported for problem b).

The phenomenon in question tended to occur for large numbers when $\mathscr{D}_{\delta t}$ was somehow involved. The cause for this could be computational, since improving the code tended to reduce the likelihood of this phenomenon. As it happens, the code was necessarily more complex for time domain evaluations of intermass displacements. The linear intervals in the tradeoff curves of figure 5.9 could similarly be related to the code: note how the interval lengths increase with α_{ut}.

Altogether, the coding required much attention in order for well-behaved solutions to be attained in CVX. Future work on the subject could investigate the exact causes of the above anomalies and further improve the code for wider applicability.

6.2 Conclusion

In this thesis, it was shown that under special circumstances, the controller design problem for a mass chain system may be formulated as a convex optimisation problem. This is true for chains consisting of any number of masses interconnected by linear springs and dampers for which the corresponding spring constants and damping coefficients are all positive. Additionally, the constraints on the system behaviour have to be on a special form, namely as arbitrary real-valued functions acting as bounds in the frequency and time domain. The closed-loop system may also be required to be internally stable and achievable by some controller.

The design method in question was then demonstrated on a system of five masses subjected to an impulse disturbance. A set of controller design problems were posed, and the corresponding convex optimisation problems were successfully formulated and solved. Tradeoff curves were generated comparing actuator effort with intermass displacement in both the frequency and time domain. The optimal closed-loop systems were probed in various ways and the appearance of the resulting graphs was judged to be reasonable and consistent with theory.

On the other hand, the controller matrices which achieved the optimal closed-loop systems were not succesfully obtained: the structure was too complex and the order too high for computation. Such closed-loop systems would therefore not be realisable in practice. Also, the displacements and forces in the resulting simulations were rather large if considered in the context of buildings. This was mainly attributed to crude approximations, such as the use of an impulse as an earthquake.

The optimal solutions were subsequently compared with the performance of the corresponding passive system in which dampers were added. As expected, the optimal closed-loop systems performed several times better than the passive system in every comparison. In particular, it was found that the greatest damper force can be over 200 times larger than that of an optimal controller achieving the same performance. It was also found that added external damping results in more violent behaviour in the first few milliseconds of some impulse responses. However, this adverse effect of increased damping can be negated using a simple form of the permitted constraints.

A specific controller design problem was also solved for different finite-dimensional approximations. The resulting minima indicated convergence to some specific value as the number of basis functions was increased. Additionally, the minima were close enough together so as to suggest that the advantage of choosing a large number of basis functions over a small one is modest at best. In exchange for making the approximation more crude, the computation is stabilised and the computation times are lowered significantly.

As for the computation involved, it was prone to instability and was less well-behaved for more complex problems. Enhancements to the code were noted to improve the situation considerably, but the solver still failed for low enough and high enough input values.

Bibliography

Boyd, S. and C. Barratt (1991). *Linear Controller Design: Limits of Performance*. Prentice-Hall.

Boyd, S., C. Barratt, and S. Norman (1990). "Linear controller design: limits of performance via convex optimization". *Proceedings of the IEEE* **78** (3), pp. 529–579.

CVX Research Inc. (2017). *CVX: matlab software for disciplined convex programming, version 2.1*. URL: `http://cvxr.com/cvx`.

Glad, T. and L. Ljung (2003). *Reglerteori – Flervariabla och olinjära metoder*. Studentlitteratur.

Heuberger, P. S., P. M. V. den Hof, and B. Wahlberg (2005). *Modelling and Identification with Rational Orthogonal Basis Functions*. Springer.

Kojima, K. and I. Takewaki (2015). "Critical earthquake response of elastic–plastic structures under near-fault ground motions (part 2: forward-directivity input)". *Frontiers in Built Environment* **1**, p. 13.

Léger, P. and S. Dussault (1992). "Seismic-energy dissipation in MDOF structures". *J. Struct. Eng.* **118**:5, pp. 1251–1269.

Mavroeidis, G. P. and A. S. Papageorgiou (2003). "A mathematical representation of near-fault ground motions". *Bulletin of the Seismological Society of America* **93**:3, 1099–1131.

Rajamani, R. (2005). *Vehicle Dynamics and Control*. Springer and Mechanical Engineering Series.

USGS (2018). *Earthquakes*. URL: `https://earthquake.usgs.gov/` (visited on 2018-10-03).

Yamamoto, K. (2016). *Disturbance attenuation in mass chains with passive interconnection* (Doctoral Thesis). URL: `https://doi.org/10.17863/CAM.26393`.

Lund University
Department of Automatic Control
Box 118
SE-221 00 Lund Sweden

Document name
MASTER'S THESIS

Date of issue
October 2018

Document Number
TFRT-6069

Author(s)
Emil Vladu

Supervisor
Kaoru Yamamoto, Dept. of Automatic Control, Lund University, Sweden
Anders Rantzer, Dept. of Automatic Control, Lund University, Sweden (examiner)

Title and subtitle

Controller Design for Multistorey Buildings via Convex Optimisation

Abstract

Earthquakes, wind and traffic may cause unwanted vibrations in buildings. Vibration control devices are often installed between floors to suppress such disturbances. In the context of buildings, controllers are commonly designed using methods which are subject to a number of limitations: we cannot discuss optimal performance or conclude that no controller exists which satisfies a given set of design specifications.

In this thesis, we address these shortcomings by considering convex optimisation for controller design in buildings. Given a restricted set of design specifications typical in vibration control, we show that the controller design problem may be formulated as a convex optimisation problem when the building is modelled as a chain of masses interconnected by linear springs and dampers. The design specifications for which this is shown are internal stability, achievability by some controller, and upper and lower bounds in the frequency and time domain.

This method is then demonstrated for a chain of five masses subjected to an impulse. A set of controller design problems are formulated and solved using CVX in Matlab. A specific design problem is also solved for different finite-dimensional approximations, and the result suggests convergence to some minimum. Tradeoff curves comparing actuator effort to intermass displacement in both the frequency and time domain are successfully computed. These results are then compared with the performance of the corresponding passive system in which dampers have been added. It is found in particular that the greatest damper force can be over 200 times larger than that of an optimal controller achieving the same performance.

Keywords

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title
0280-5316

ISBN

Language
English

Number of pages
1-65

Recipient's notes

Security classification