Gain-Scheduled Control of a Quadcopter UAV

by

Shaun Sawyer

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Applied Mathematics

Waterloo, Ontario, Canada, 2015

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

In this thesis we develop a gain-scheduled control law for the quadcopter unmanned aerial vehicle (UAV). Techniques from linear control theory are introduced and used to construct adaptive proportional and proportional-integral control laws for use with both state and observer-based output feedback. The controller monitors the yaw angle of the quadcopter and updates a gain matrix as the system evolves through operating points. To demonstrate the effectiveness of the gain-scheduled controller, trajectories involving significant variation in the yaw angle are tracked by the quadcopter, including a helix and Lissajous curve. We consider physical implementation of the controller, and offer suggestions for improvement and future work.

Acknowledgements

I would like to thank my supervisor Dr. Dong Eui Chang for mentoring me throughout my graduate studies, offering invaluable advice, encouragement, and being a great person to work with.

Thank you also to my committee members Dr. Xinzhi Liu and Dr. Soo Jeon for reading my thesis, and offering your comments and criticism.

Dedication

This thesis is dedicated to my family. Thank you for always being there for me, even after I left Windsor.

Table of Contents

List of Figures

Chapter 1

Introduction

The control of nonlinear systems is a topic of great importance in the present time. Unlike linear systems, nonlinear systems have no defined structure and so the task of using feedback to alter the dynamics in a favourable way is not as simple. As such, mathematicians and control engineers have developed a variety of techniques to apply to nonlinear systems in order to control them. Commonly employed control strategies for nonlinear systems include linearization, sliding mode control, backstepping, feedback linearization, Lyapunov redesign, high-gain observers, control using differential flatness, passivity-based control, and gain scheduling.

Appealing to the defined structure and desirable properties that linear systems possess, it is often the case that control engineers will first linearize a nonlinear system to analyze local behavior about operating points of interest to them. If local control about these points is all that is desired, then a linear controller developed using the linearized model can be applied to the nonlinear model to stabilize the full system. The same linear controller can also be used to track a reference trajectory, provided that the reference does not vary too rapidly and any parameters that appear in the linear dynamics do not vary as the reference varies. This detail is subtle but important, and if neglected can result in the failure of the controller. As such, the approach of developing a single linear control to guide a system along certain desired trajectories breaks down for a lot of systems. If the controller could adapt to this change in the linearized system as the dynamics evolved, we may be able to preserve its linear structure. We can use gain scheduling to achieve this goal.

A gain-scheduled controller is a natural extension of the *fixed-gain* controller. Unlike the aforementioned controller guaranteed to only work in a small neighbourhood of a specified setpoint, a gain-scheduled controller can extend the linearization approach to be valid for

a range of setpoints. This improvement is accomplished by tuning the gains, in accordance with how the linearized dynamics are evolving, in a way that optimizes the controller for each setpoint. In a physical sense, this technique of control is best applied to processes that require varying amounts of effort from the controller as the system evolves. Generally, one or more parameters called *scheduling variables* are chosen to monitor and are used to determine how and when the control should be altered.

One such system in which gain scheduling control would be appropriate is the four rotor helicopter known as a *quadcopter*. To control this unmanned aerial vehicle, physical control inputs are assigned to the torque of each rotor, enabling translation in three spatial directions, as well as rotation in three dimensions. This configuration makes the quadcopter highly maneuverable. In particular, the quadcopter is capable of tracking a trajectory in x, y, z space while simultaneously following an independent trajectory for its heading or yaw angle configuration. The desired trajectory for the yaw angle to follow will contribute to how the control inputs should vary in the tracking of a reference signal. This property is exhibited when linearizing the nonlinear dynamics of the quadcopter about any hovering configuration. It then becomes apparent that the linearized dynamics depend continuously on the desired yaw angle, hence making the quadcopter an ideal candidate for the application of gain-scheduled control.

Chapter 2

Theory

Although the quadcopter is in fact a nonlinear system, we can still use results from linear control theory in our efforts to control it by appealing to the linearization of the system about its equilibrium points. As such, before we develop the various controllers, we first go over the necessary results derived from linear control theory.

A common approach employed in the control of linear and nonlinear systems is to use some variation of what is known as Proportional-Integral-Derivative (PID) control. Such a controller takes the form

$$\mathbf{u} = -\mathbf{K}_1\mathbf{e}(t) - \mathbf{K}_2 \int \mathbf{e}(t)\, dt - \mathbf{K}_3 \frac{d\,\mathbf{e}(t)}{dt},$$

where the error $\mathbf{e}(t)$ is the difference between the actual and desired states of the system, and the *gains*, or the $\mathbf{K}_i$'s, are tuned to achieve the desired performance from the system. In the case of having a multidimensional control input where $\mathbf{u}$ is a vector, the gains are in fact matrices. It is often the case that only some combination of the terms in the control law will be used. For our purposes we will only consider the cases when $\mathbf{K}_1 \neq 0, \mathbf{K}_2 = \mathbf{K}_3 = 0$ and $\mathbf{K}_1, \mathbf{K}_2 \neq 0, \mathbf{K}_3 = 0$, or P and PI control. In the selection of the gains, we will be employing a *linear quadratic regulator* to arrive at an "optimal" feedback control, where we choose the criteria for optimality. The question of whether or not a suitable controller of this form can be developed will be posed by the notion of *controllability*.

We also consider when the full state variable is not available for feedback, leaving us with only a subset of the state to draw measurements from. In this case we make use of an *observer* - an exogenous system that is designed to converge to the actual state of the system. Similar to the notion of controllability, the question of whether or not a suitable observer exists will be posed by the notion of *observability*.

2.1 Linear Control Systems

Linear control systems are attractive to mathematicians and control engineers because of their concise algebraic structure. This feature allows them to be easily manipulated in the case of control design. There are two main types of linear control systems, and a third type of which we will discuss later on. We begin with a couple of definitions.

Definition 2.1.1. A *linear control system* is a dynamical system of the form

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

where

$\mathbf{x}(t)$ is the $n \times 1$ state vector,
$\mathbf{y}(t)$ is the $p \times 1$ output vector,
$\mathbf{u}(t)$ is the $m \times 1$ control vector,
$\mathbf{A}(t)$ is the $n \times n$ plant matrix,
$\mathbf{B}(t)$ is the $n \times m$ control matrix,
$\mathbf{C}(t)$ is the $p \times n$ output matrix,
$\mathbf{D}(t)$ is the $p \times m$ feedforward matrix.

As seen, the matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ generally depend on the time variable t. However, we will be primarily concerned with linear systems that do *not* explicitly depend on time, hence motivating the next definition.

Definition 2.1.2. A *linear time invariant (LTI) control system* is a dynamical system of the form

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

where $\mathbf{x}(t) \in \mathbb{R}^n, \mathbf{y}(t) \in \mathbb{R}^p, \mathbf{u}(t) \in \mathbb{R}^m$ are time-dependent vectors, and $\mathbf{A} \in \mathbb{R}^{n\times n}, \mathbf{B} \in \mathbb{R}^{n\times m}, \mathbf{C} \in \mathbb{R}^{p\times n}, \mathbf{D} \in \mathbb{R}^{p\times m}$ are constant matrices.

We will concern ourselves only with autonomous linear systems, or LTI systems, of the form in definition 2.1.2, and, unless otherwise stated, is to be understood as such when any reference to linear systems is made henceforth. Also, we will assume that the feedforward matrix $\mathbf{D} = 0$, so that the system can now be written more concisely as

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \\ \mathbf{y} &= \mathbf{C}\mathbf{x}. \end{aligned} \tag{2.1}$$

For simplicity, we will refer to a linear system of the form (2.1) as $(\mathbf{A}, \mathbf{B}, \mathbf{C})$.

2.2 Stabilization of Linear Systems

We first discuss the problem of finding a *state feedback control*, where we assume that the output of the linear system (2.1) is the entire state. That is, $\mathbf{y} = \mathbf{x}$, or $\mathbf{C} = \mathbf{I}_n$ where $\mathbf{I}_n$ is the $n \times n$ identity matrix. We formulate the problem as follows.

Given the linear system

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{2.2}$$

find a feedback controller

$$\mathbf{u} = \gamma(\mathbf{x})$$

such that the origin $\mathbf{x} = 0$ is an asymptotically stable equilibrium point of the closed-loop system

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\gamma(\mathbf{x}).$$

This can be done by taking the feedback control to be

$$\mathbf{u} = -\mathbf{K}\mathbf{x}$$

for some suitably chosen *gain matrix* $\mathbf{K}$. Substituting this control into (2.2) results in the closed-loop system

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}(-\mathbf{K}\mathbf{x}) \\ &= (\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}. \end{aligned}$$

Choosing $\mathbf{K}$ such that the matrix $\mathbf{A} - \mathbf{B}\mathbf{K}$ is *Hurwitz*, where all of its eigenvalues have negative real part, will result in the global asymptotic stability of the origin $\mathbf{x} = 0$. Moreover, the origin will be globally exponentially stable.

Often, it is the case where the entire state is not available on-line for the use of feedback, and only a subset of the state variables can be used in the design of the controller. Here the output vector $\mathbf{y}$ is of dimension less than the state $\mathbf{x}$. As such, we instead consider the problem of designing an *output feedback control*, which we formulate as follows.

Given the linear system

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \\ \mathbf{y} &= \mathbf{C}\mathbf{x} \end{aligned} \tag{2.3}$$

find a static feedback controller

$$\mathbf{u} = \gamma(\mathbf{y})$$

or a dynamic feedback controller

$$\begin{aligned}\mathbf{u} &= \gamma(\hat{\mathbf{x}}),\\ \dot{\hat{\mathbf{x}}} &= \mathbf{g}(\mathbf{y}, \mathbf{u})\end{aligned}$$

such that the origin is an asymptotically stable equilibrium point of the closed-loop system. In the dynamic feedback controller case, the origin to be stabilized is now $(\mathbf{x} = 0, \hat{\mathbf{x}} = 0)$, where $\hat{\mathbf{x}}$ is an estimate of the full state variable to be obtained by a suitable *state observer*.

A state observer is simply an external system that can be used to estimate the state variable from the input $\mathbf{u}$ and output $\mathbf{y}$. We will be using an observer known as the *Luenberger* observer introduced by David Luenberger [1]. It takes the form

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}\mathbf{u} + \mathbf{L}(\mathbf{y} - \mathbf{C}\hat{\mathbf{x}}), \tag{2.4}$$

where $\mathbf{L}$ is the *observer gain* to be suitably chosen by us. It is not difficult to show that the error dynamics satisfy

$$\dot{\mathbf{e}} = (\mathbf{A} - \mathbf{L}\mathbf{C})\mathbf{e},$$

where $\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}}$. By choosing $\mathbf{L}$ such that the matrix $\mathbf{A} - \mathbf{L}\mathbf{C}$ is Hurwitz, we will have as $t \to \infty$ that $\mathbf{e} \to 0$, or equivalently $\hat{\mathbf{x}} \to \mathbf{x}$. We can then take the control law as

$$\begin{aligned}\mathbf{u} &= -\mathbf{K}\hat{\mathbf{x}},\\ \dot{\hat{\mathbf{x}}} &= \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}\mathbf{u} + \mathbf{L}(\mathbf{y} - \mathbf{C}\hat{\mathbf{x}})\end{aligned}$$

with $\mathbf{A} - \mathbf{B}\mathbf{K}$ Hurwitz, resulting in the augmented closed-loop dynamics

$$\begin{pmatrix}\dot{\mathbf{x}}\\ \dot{\mathbf{e}}\end{pmatrix} = \begin{pmatrix}\mathbf{A} - \mathbf{B}\mathbf{K} & \mathbf{B}\mathbf{K}\\ 0 & \mathbf{A} - \mathbf{L}\mathbf{C}\end{pmatrix}\begin{pmatrix}\mathbf{x}\\ \mathbf{e}\end{pmatrix}. \tag{2.5}$$

The upper triangular matrix in (2.5) shows that the gain matrices $\mathbf{K}$ and $\mathbf{L}$ can be designed independently[1] of one another without affecting the overall stability of the system[2]. Whether or not we can actually find such matrices as described above depends on the notions of *controllability* and *observability*.

[1] $\mathbf{K}$ and $\mathbf{L}$ are typically chosen so that the observer converges to the state quicker than the state itself converges to the origin.

[2] This is known as the *Separation Principle* (see, for example, [2]).

2.3 Controllability and Observability

We now discuss two very important concepts of modern control theory introduced by Kalman in 1960 [3], namely, *controllability* and *observability*. These concepts are the outcome of attempting to answer the following two questions [4]:

1. Is it possible to find a control $\mathbf{u}(t)$ to bring the initial state $\mathbf{x}(t_0)$ to any desired final state $\mathbf{x}(t_f)$ in finite time?
2. Is it possible to determine the state of the system by measuring only the system output over a finite interval of time?

If the answer to the first question is yes, then we say that the system is controllable. Similarly, if the answer to the second question is yes, then we say that the system is observable.

For LTI systems, the condition for checking controllability and observability is very simple and uses the following definitions.

Definition 2.3.1. The *controllability matrix* $\mathcal{C}$ of an LTI system is defined to be

$$\mathcal{C}(\mathbf{A},\mathbf{B}) = \begin{pmatrix}\mathbf{B} & \mathbf{AB} & \mathbf{A}^2\mathbf{B} & \cdots & \mathbf{A}^{n-1}\mathbf{B}\end{pmatrix}.$$

Definition 2.3.2. The *observability matrix* $\mathcal{O}$ of an LTI system is defined to be

$$\mathcal{O}(\mathbf{A},\mathbf{C}) = \begin{pmatrix}\mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{n-1}\end{pmatrix}.$$

Using these matrices we can now state the conditions required for the controllability and observability of LTI systems.

Theorem 2.3.1. *The pair* $(\mathbf{A},\mathbf{B})$ *is controllable if and only if the rank of* $\mathcal{C}(\mathbf{A},\mathbf{B})$ *is equal to* n*. That is,* $\mathcal{C}$ *has full row rank.*

Theorem 2.3.2. *The pair* $(\mathbf{A},\mathbf{C})$ *is observable if and only if the rank of* $\mathcal{O}(\mathbf{A},\mathbf{C})$ *is equal to* n*. That is,* $\mathcal{O}$ *has full column rank.*

The proofs of these theorems are well known and can be found in many textbooks on linear control theory, such as [2].

What this means for us is that the closed-loop eigenvalues in the augmented system (2.5) can be arbitrarily assigned (with the restriction that the poles not on the real axis are complex conjugate pairs) by choosing appropriate $\mathbf{K}$ and $\mathbf{L}$ matrices.[3]

2.4 Linear Quadratic Regulator

Let as assume that we have the linear system

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$

where the pair $(\mathbf{A}, \mathbf{B})$ is controllable. From the above discussion, we know that we can choose a gain matrix $\mathbf{K}$ to arbitrarily place the eigenvalues of the matrix $\mathbf{A} - \mathbf{B}\mathbf{K}$ so that the state feedback control $\mathbf{u} = -\mathbf{K}\mathbf{x}$ stabilizes the origin of the corresponding closed-loop system. The question now is how to proceed in choosing which $\mathbf{K}$ to use.

The method we will be using is known as an *infinite horizon continuous time linear quadratic regular*. The goal is to find the corresponding control $\mathbf{u}$ such that the cost

$$J = \int_0^\infty \left(\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u}\right)\, dt \tag{2.6}$$

is minimized, where $\mathbf{Q} \in \mathbb{R}^{n\times n}$ and $\mathbf{R} \in \mathbb{R}^{m\times m}$ are positive definite symmetric matrices chosen by us. The term $\mathbf{x}^T\mathbf{Q}\mathbf{x}$ can be thought of as a weighted sum, with weight $\mathbf{Q}$, representing the deviation of the state value from the origin. The term $\mathbf{u}^T\mathbf{R}\mathbf{u}$ can be thought of similarly as the magnitude of the control action. Roughly speaking, increasing elements of $\mathbf{Q}$ results in quicker convergence of the corresponding states, and increasing elements of $\mathbf{R}$ limits the energy that can be expended by the control. Varying $\mathbf{Q}$ and $\mathbf{R}$ consequently vary the locations of the closed-loop poles, thereby affecting the dynamic performance of the system. See [2] for more details.

It is well known that the solution to this problem is the feedback control

$$\mathbf{u} = -\mathbf{K}\mathbf{x},$$

[3]This procedure is known as *pole placement*. In simulation, we will use the `place` command in MATLAB, which uses the algorithm discussed in [5].

with

$$\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}, \tag{2.7}$$

where $\mathbf{P}$ in the unique symmetric positive definite matrix that solves the *algebraic Ricatti equation*

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} = 0. \tag{2.8}$$

In the case of observer feedback control of a system $(\mathbf{A}, \mathbf{B}, \mathbf{C})$, where we have the additional task of designing an observer gain $\mathbf{L}$ so that $\mathbf{A} - \mathbf{L}\mathbf{C}$ is Hurwitz, we can simply use pole placement to design $\mathbf{L}$ such that the eigenvalues of $\mathbf{A} - \mathbf{L}\mathbf{C}$ are some factor larger (in magnitude) than the eigenvalues of $\mathbf{A} - \mathbf{B}\mathbf{K}$ resulting from the above procedure.

2.5 Stabilization Through Linearization

We now extend our scope to that of nonlinear systems. The techniques that we developed for linear systems guarantee global exponential stability of a desired setpoint. Unfortunately, when we apply these techniques to nonlinear systems, we first must linearize the system so as to approximate the local behaviour of the system about a certain point. Doing so and applying the above techniques will result in local asymptotic stability.

We begin with outlining the procedure of designing a linear state feedback controller to achieve local stabilization about the origin of a nonlinear system. Suppose that we have a nonlinear system

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}), \tag{2.9}$$

with $\mathbf{f}(\mathbf{x}, \mathbf{u})$ continuously differentiable in a domain $D_x \times D_\mathbf{u} \in \mathbb{R}^n \times \mathbb{R}^p$ containing the origin $(\mathbf{x} = 0, \mathbf{u} = 0)$ and $\mathbf{f}(0, 0) = 0$ (so that the origin is an equilibrium point). Linearizing (2.9) about $(0, 0)$ give us the linear system

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \tag{2.10}$$

where $\mathbf{A}$ and $\mathbf{B}$ are the Jacobians of $\mathbf{f}$ with respect to $\mathbf{x}$ and $\mathbf{u}$ evaluated at the origin, namely,

$$\mathbf{A} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}, \mathbf{u})\bigg|_{(\mathbf{x},\mathbf{u})=(0,0)}, \qquad \mathbf{B} = \frac{\partial \mathbf{f}}{\partial \mathbf{u}}(\mathbf{x}, \mathbf{u})\bigg|_{(\mathbf{x},\mathbf{u})=(0,0)}.$$

Then (2.10) approximates the local behaviour of (2.9) about $(0, 0)$. Assuming that the pair $(\mathbf{A}, \mathbf{B})$ is controllable, we can proceed as in section 2.2 to develop a linear feedback control

$\mathbf{u} = -\mathbf{K}\mathbf{x}$ with $\mathbf{A} - \mathbf{B}\mathbf{K}$ Hurwitz. Under this controller, the closed-loop nonlinear system is

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, -\mathbf{K}\mathbf{x}). \tag{2.11}$$

We note that the origin $\mathbf{x} = 0$ is an equilibrium point of the closed-loop system, and linearization about the origin results in

$$\begin{aligned}\dot{\mathbf{x}} &= \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}, -\mathbf{K}\mathbf{x})\bigg|_{\mathbf{x}=0} \mathbf{x} + \frac{\partial \mathbf{f}}{\partial \mathbf{u}}(\mathbf{x}, -\mathbf{K}\mathbf{x})\bigg|_{\mathbf{x}=0} \frac{\partial \mathbf{u}}{\partial \mathbf{x}}\bigg|_{\mathbf{x}=0} \mathbf{x} \\ &= \mathbf{A}\mathbf{x} + \mathbf{B}(-\mathbf{K})\mathbf{x} \\ &= (\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}.\end{aligned}$$

Since $\mathbf{A} - \mathbf{B}\mathbf{K}$ is Hurwitz, it follows that the origin of (2.11) is asymptotically stable. This can be verified using a suitable Lyapunov function $V = \mathbf{x}^T\mathbf{P}\mathbf{x}$ where the unique symmetric positive definite matrix $\mathbf{P}$ solves the Lyapunov equation

$$\mathbf{P}(\mathbf{A} - \mathbf{B}\mathbf{K}) + (\mathbf{A} - \mathbf{B}\mathbf{K})^T\mathbf{P} + \mathbf{Q} = 0,$$

with $\mathbf{Q}$ any positive definite symmetric matrix. Such a matrix $\mathbf{P}$ is guaranteed to exist so long as $\mathbf{A} - \mathbf{B}\mathbf{K}$ is Hurwitz. We will then have that $\dot{V} = -\mathbf{x}^T\mathbf{Q}\mathbf{x}$ along trajectories of (2.10), establishing asymptotic stability of the origin of (2.11). This fact is a consequence of Lyapunov's Indirect Method (see [6] for a proof of this). The Lyapunov function $V(\mathbf{x})$ can be used to estimate the region of attraction of the origin of the closed-loop system.

2.6 Integral Action Through Linearization

So far the only form of controller we have considered is one that applies a corrective effort that is proportional to the error, namely $\mathbf{u}(t) = -\mathbf{K}\mathbf{e}(t)$, or $\mathbf{u}(t) = -\mathbf{K}\mathbf{x}(t)$, where $\mathbf{e}(t) = \mathbf{x}(t) - 0$ in the case of stabilizing the origin. This control law is known as proportional (P) control and is suitable for use under certain design specifications. However, we may be interested in increasing the speed of convergence without having to continuously increase the gain $\mathbf{K}$. Also, there may be the existence of an undesired steady-state error depending on the reference we are tracking and if there are constant parameter perturbations in our model. To deal with this, we introduce the concept of integral action. Our goal will be to design a linear feedback controller that will ensure asymptotic regulation under parameter perturbations for which the closed-loop system remains stable. We will follow [6] in our approach.

Suppose we now have the nonlinear system

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}), \\ \mathbf{y} &= \mathbf{h}(\mathbf{x}, \mathbf{w}), \end{aligned} \tag{2.12}$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^p$, $\mathbf{y} \in \mathbb{R}^p$ is the controlled output, and $\mathbf{w} \in \mathbb{R}^l$ is a constant vector of unknown perturbations and disturbances. We assume $\mathbf{f}$ and $\mathbf{h}$ are continuously differentiable in $\mathbf{x}$ and $\mathbf{u}$, and continuous in $\mathbf{w}$ in a domain $D_{\mathbf{x}} \times D_{\mathbf{u}} \times D_{\mathbf{w}} \subset \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^l$.

Suppose we want to achieve asymptotic regulation of the output $\mathbf{y}$ to a constant reference $\mathbf{r} \in D_{\mathbf{r}} \subset \mathbb{R}^p$ so that $\mathbf{y} \to \mathbf{r}$ as $t \to \infty$. Let

$$\mathbf{v} = \begin{pmatrix} \mathbf{r} \\ \mathbf{w} \end{pmatrix} \in D_{\mathbf{r}} \times D_{\mathbf{w}}.$$

Assume that for each $\mathbf{v} \in D_{\mathbf{r}} \times D_{\mathbf{w}}$ there is a corresponding $(\mathbf{x}_{ss}, \mathbf{u}_{ss})$ depending continuously on $\mathbf{v}$ and satisfying

$$\begin{aligned} 0 &= \mathbf{f}(\mathbf{x}_{ss}, \mathbf{u}_{ss}, \mathbf{w}), \\ 0 &= \mathbf{h}(\mathbf{x}_{ss}, \mathbf{w}) - \mathbf{r}. \end{aligned}$$

We now implement integral action. Define the new variable σ to be the integral of the error $\mathbf{e} = \mathbf{y} - \mathbf{r}$, or equivalently

$$\dot{\sigma} = \mathbf{h}(\mathbf{x}, \mathbf{w}) - \mathbf{r}$$

and augment this to the state dynamics (2.12) to obtain the augmented dynamics

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}), \\ \dot{\sigma} &= \mathbf{h}(\mathbf{x}, \mathbf{w}) - \mathbf{r}. \end{aligned} \tag{2.13}$$

Our goal now is to develop the feedback controller to stabilize the system (2.13) at an equilibrium point $(\mathbf{x}_{ss}, \sigma_{ss})$ such that σ_{ss} results in the desired $\mathbf{u}_{ss}$. Linearizing about $(\mathbf{x}_{ss}, \sigma_{ss})$ results in

$$\begin{aligned} \dot{\mathbf{x}_\delta} &= \mathbf{A}\mathbf{x}_\delta + \mathbf{B}\mathbf{u}_\delta, \\ \dot{\sigma_\delta} &= \mathbf{C}\mathbf{x}_\delta, \end{aligned} \tag{2.14}$$

where $\mathbf{x}_\delta = \mathbf{x} - \mathbf{x}_{ss}$, $\mathbf{u}_\delta = \mathbf{u} - \mathbf{u}_{ss}$, $\sigma_\delta = \sigma - \sigma_{ss}$, and

$$\mathbf{A} = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{ss}, \qquad \mathbf{B} = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{ss}, \qquad \mathbf{C} = \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{ss}.$$

We can now define

$$\xi_\delta = \begin{pmatrix} \mathbf{x}_\delta \\ \sigma_\delta \end{pmatrix}$$

so that (2.14) can be written

$$\dot{\xi}_\delta = \mathcal{A}\xi_\delta + \mathcal{B}\mathbf{u}_\delta, \tag{2.15}$$

where $\mathcal{A}, \mathcal{B}$ are block matrices of appropriate dimension given by

$$\mathcal{A} = \begin{pmatrix} \mathbf{A} & 0 \\ \mathbf{C} & 0 \end{pmatrix}, \qquad \mathcal{B} = \begin{pmatrix} \mathbf{B} \\ 0 \end{pmatrix}. \tag{2.16}$$

It is now clear that choosing $\mathbf{u}_\delta = -\mathcal{K}\xi_\delta$ such that $\mathcal{A} - \mathcal{B}\mathcal{K}$ is Hurwitz will result in the asymptotically stable closed-loop system

$$\dot{\xi}_\delta = (\mathcal{A} - \mathcal{B}\mathcal{K})\xi_\delta.$$

Note that in order for such a $\mathcal{K}$ to be guaranteed to exist we require that $(\mathcal{A}, \mathcal{B})$ is controllable. This will be the case so long as $(\mathbf{A}, \mathbf{B})$ is controllable, and[4] we have that

$$\operatorname{rank} \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & 0 \end{pmatrix} = n + p. \tag{2.17}$$

See section B.1 of appendix B for a proof of this. We will then have

$$\xi_\delta \to 0 \Rightarrow \begin{array}{c} \mathbf{x}_\delta \to 0 \\ \sigma_\delta \to 0 \end{array} \Rightarrow \begin{array}{c} \mathbf{x} \to \mathbf{x}_{ss} \\ \sigma \to \sigma_{ss} \end{array}.$$

From (2.13) we will thus have by continuity of $\mathbf{h}(\mathbf{x}, \mathbf{w})$ on $\mathbf{x}$ that

$$\mathbf{h}(\mathbf{x}, \mathbf{w}) - \mathbf{r} \to \mathbf{h}(\mathbf{x}_{ss}, \mathbf{w}) - \mathbf{r} \to 0$$

so that $\mathbf{y} \to \mathbf{r}$, as desired.

By expressing the gain matrix $\mathcal{K}$ as the block matrix

$$\mathcal{K} = \begin{pmatrix} \mathbf{K}_1 & \mathbf{K}_2 \end{pmatrix}$$

we can write the control law as

$$\mathbf{u}_\delta = -\mathbf{K}_1\mathbf{x}_\delta - \mathbf{K}_2\sigma_\delta,$$

[4]The condition (2.17) guarantees that the model $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ has no transmission zeros [6].

or equivalently

$$\mathbf{u} = -\mathbf{K}_1(\mathbf{x} - \mathbf{x}_{ss}) - \mathbf{K}_2(\sigma - \sigma_{ss}) + \mathbf{u}_{ss}. \tag{2.18}$$

Note that $\mathcal{A}-\mathcal{B}\mathcal{K}$ being Hurwitz implies that $\mathbf{K}_2$ is nonsingular; this is shown in section B.2 of appendix B. We can then take[5]

$$\sigma_{ss} = -\mathbf{K}_2^{-1}\mathbf{u}_{ss}$$

so that (2.18) becomes

$$\mathbf{u} = -\mathbf{K}_1(\mathbf{x} - \mathbf{x}_{ss}) - \mathbf{K}_2\sigma \tag{2.19}$$

which is the conventional PI controller

$$\mathbf{u}(t) = -\mathbf{K}_1\mathbf{e}(t) - \mathbf{K}_2 \int_0^t \mathbf{e}(\tau)\, d\tau.$$

The effects of implementing integral action in the control law will be seen in chapter 5.

In the case of output feedback, we now couple the control (2.19) with an observer as in (2.4) so that the control law now takes the form

$$\begin{aligned} \mathbf{u} &= -\mathbf{K}_1(\hat{\mathbf{x}} - \mathbf{x}_{ss}) - \mathbf{K}_2\sigma, \\ \dot{\hat{\mathbf{x}}} &= \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}\mathbf{u} + \mathbf{L}(\mathbf{y} - \mathbf{C}\hat{\mathbf{x}}), \end{aligned} \tag{2.20}$$

where $\mathbf{L}$ is designed such that $\mathbf{A} - \mathbf{LC}$ is Hurwitz. The closed-loop augmented dynamics are now

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, -\mathbf{K}_1(\hat{\mathbf{x}} - \mathbf{x}_{ss}) - \mathbf{K}_2\sigma, \mathbf{w}), \\ \dot{\hat{\mathbf{x}}} &= \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}(-\mathbf{K}_1(\hat{\mathbf{x}} - \mathbf{x}_{ss}) - \mathbf{K}_2\sigma) + \mathbf{L}(\mathbf{y} - \mathbf{C}\hat{\mathbf{x}}), \\ \dot{\sigma} &= \mathbf{y} - \mathbf{r}. \end{aligned}$$

Linearizing about $(\mathbf{x}_{ss}, \mathbf{x}_{ss}, \sigma_{ss})$ where $\sigma_{ss} = -\mathbf{K}_2^{-1}\mathbf{u}_{ss}$ results in

$$\begin{aligned} \dot{\mathbf{x}}_\delta &= \mathbf{A}\mathbf{x}_\delta - \mathbf{B}\mathbf{K}_1\hat{\mathbf{x}}_\delta - \mathbf{B}\mathbf{K}_2\sigma_\delta, \\ \dot{\hat{\mathbf{x}}}_\delta &= \mathbf{L}\mathbf{C}\mathbf{x}_\delta + (\mathbf{A} - \mathbf{B}\mathbf{K}_1 - \mathbf{L}\mathbf{C})\hat{\mathbf{x}}_\delta - \mathbf{B}\mathbf{K}_2\sigma_\delta, \\ \dot{\sigma}_\delta &= \mathbf{C}\mathbf{x}_\delta, \end{aligned}$$

[5]By using the control law in (2.19), we do not need to specify the value of σ_{ss}. It comes naturally from our pole placement procedure for the matrix $\mathcal{A} - \mathcal{B}\mathcal{K}$.

which we can write as

$$\begin{pmatrix} \dot{\mathbf{x}}_\delta \\ \dot{\hat{\mathbf{x}}}_\delta \\ \dot{\sigma}_\delta \end{pmatrix} = \begin{pmatrix} \mathbf{A} & -\mathbf{B}\mathbf{K}_1 & -\mathbf{B}\mathbf{K}_2 \\ \mathbf{L}\mathbf{C} & \mathbf{A} - \mathbf{B}\mathbf{K}_1 - \mathbf{L}\mathbf{C} & -\mathbf{B}\mathbf{K}_2 \\ \mathbf{C} & 0 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{x}_\delta \\ \hat{\mathbf{x}}_\delta \\ \sigma_\delta \end{pmatrix}, \tag{2.21}$$

where the matrix in (2.21) is Hurwitz by construction. Hence as before we will have that $\mathbf{y} \to \mathbf{r}$ as desired.

Chapter 3

Gain Scheduling

Developing a linear control to stabilize the origin of a linear system as we have done above guaranteed us global asymptotic (in fact exponential) stability. However, as we have seen, the linearization about a certain point of a nonlinear system is only valid *near* that point. As such, we can only guarantee local asymptotic stability. By using gain scheduling, we can extend the region of stability to a range of operating points. This is done by allowing the controller to vary as the system advances through operating points.

3.1 Background

Gain scheduling as a control technique for nonlinear systems has been around for a while, being used relatively early in military applications. Early examples of gain scheduling include application to flight control and automotive engine control as addressed in [7]. Unfortunately, it was difficult and expensive to realize in hardware before the advent of digital computers, hindering its implementation in commercial use. However, as digital controllers increased in popularity, so did the use of gain scheduling.

Before the 1990s, theoretical treatments of gain scheduling as a nonlinear control technique were rare. Its use was often justified through simulations alone, ensuring the controller maintained satisfactory performance for the desired range of system parameters. Recently, authors have been taking to a more analytical treatment of the subject in developing a framework for gain-scheduled control. One author in particular, Wilson J. Rugh, has made many contributions in this time (see [7, 8, 9, 10, 11]).

3.2 Linear Parameter-Varying Systems

Before we discuss the procedure of gain scheduling, we will first introduce the concept of linear parameter-varying systems.

As we have seen, the stabilization via linearization approach discussed in section 2.5 works well when we are only interested in stabilizing about a specific operating point, or tracking a reference signal with which the matrices $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ in the corresponding linearized system do not vary. However, we may be interested in tracking a parameterized reference signal, resulting in a linearized model that may be dependent on this parameter. These models are known as *linear parameter-varying systems*. Such systems take the form

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}(\alpha)\mathbf{x} + \mathbf{B}(\alpha)\mathbf{u}, \\ \mathbf{y} &= \mathbf{C}(\alpha)\mathbf{x}, \end{aligned} \tag{3.1}$$

where α is an exogenous parameter that can generally depend on time. It is this form of linear system that we are most interested in, and is the foundation behind the modern development of gain-scheduled control.

3.3 Procedure

The term "gain scheduling" has been associated with many different design notions. It is sometimes simply a point of view taken by the control engineer as to whether or not their controller is operating under some gain-scheduled procedure. The most common approach, and the one that we will be employing in the control of the quadcopter, is summarized as follows.

Step 1. Construct a linear parameter-varying model of the plant, usually done by computing the Jacobian linearization of the nonlinear dynamics about a family of equilibrium points, also known as operating or set points. The parameters that appear in the model are typically the *scheduling variables* — variables whose values will be monitored to determine how and when the controller will switch.

Step 2. Use techniques from linear control theory to develop a family of stabilizing controllers for the linear parameter-varying model of the plant. Typically, the controllers are developed so that the closed-loop linear system should exhibit the desired performance for each frozen value of the parameter.

Step 3. Determine *how* the controller will switch, or how the gains will be scheduled. The control law can consist of a set of *point designs* that are linearized about the specified operating points, and will switch between these point designs as the scheduling variable changes. Alternatively, the point designs can be interpolated in some way, through linear interpolation of the gains or other methods (some examples are given in [10, 11]).

Step 4. Assess the performance of the controller. Doing this in a rigorous fashion may be simple or difficult, depending on whether or not analytical performance guarantees were taken as part of the design process in the previous step. However, in many situations this is not the case and further investigation is required in assessing the local and nonlocal performance.

These are the steps as outlined by Rugh in [7]. Note that it is not obvious how to go about implementing step 3. Certain techniques that work well for some systems may fail when applied to others.

3.4 Advantages and Disadvantages

The main advantage that gain scheduling has over other nonlinear control methods is that it is an application of linear design techniques to a nonlinear system, and linear control theory is already very well understood. We can use the tools that we have for linear control systems, including quadratic performance measures, output feedback techniques, and frequency domain methods. Of course, their effect when applied to nonlinear systems will be less understood. Additionally, systems operating under gain-scheduled control typically respond quickly when faced with changing operating conditions, provided that the scheduling variable is chosen to reflect these changes. Finally, gain scheduling is often less intensive computationally when compared to other nonlinear control techniques.

Gain scheduling does come with some disadvantages, the biggest disadvantage being a consequence of the main advantage stated above. The fact that linear control techniques are applied results in stability results that are only local in nature. Though, in many physical applications nonlinear control implementations will be local as well. Further, as mentioned in step 3 of section 3.3, gain scheduling can be very *ad hoc*. Many decisions to be made are system-dependent, including the choice of an appropriate scheduling variable and scheduling procedure.

3.5 Gain-Scheduled Control via Point Design

When designing a control with the linearization approach discussed in section 2.5, we were faced with the limitation of local stabilization. That is, the controller we developed was only guaranteed to work close to a single operating point. If we wanted to switch to another operating point, or a sequence of operating points, we may not be able to do so. As mentioned previously, it may be the case that the linearized dynamics vary as we continue through our sequence of operating points, requiring the development of a sequence of different linear controllers in order to maintain stability.

Gain scheduling can extend the stabilization through linearization approach to a wider range of operating points by implementing new controllers as necessary. A nonlinear controller is constructed for a nonlinear system by patching a series of linear controllers together. Each linear controller is constructed for a specific value of some parameter affecting the linearized dynamics of the nonlinear system. This parameter is known as the *scheduling variable*, and the controller associated with each operating point is called a *point design* [7]. The choice of which linear controller to use at any given time is made by monitoring the scheduling variable.

We illustrate this design procedure for a general nonlinear system. Suppose that we have the system

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}), \tag{3.2}$$

with $\mathbf{f}(\mathbf{x}, \mathbf{u})$ continuously differentiable in a domain $D_{\mathbf{x}} \times D_{\mathbf{u}} \in \mathbb{R}^n \times \mathbb{R}^p$ containing the origin ($\mathbf{x} = 0, \mathbf{u} = 0$) and $\mathbf{f}(0, 0) = 0$ (so that the origin is an equilibrium point).

Now let us attempt to design a state feedback control so that $\mathbf{x}$ tracks a reference signal $\mathbf{r}$. Suppose that $\mathbf{r} = \alpha_0$ where α_0 is constant. Then the desired steady state value of $\mathbf{x}$ is α_0 and the desired steady state value of the control input is $\mathbf{u}_{ss}(\alpha_0)$ where $\mathbf{u}_{ss}(\alpha_0)$ satisfies

$$0 = \mathbf{f}(\alpha_0, \mathbf{u}_{ss}(\alpha_0)).$$

Now define

$$\mathbf{x}_\delta = \mathbf{x} - \alpha_0, \qquad \mathbf{u}_\delta = \mathbf{u} - \mathbf{u}_{ss}(\alpha_0),$$

so that $(\mathbf{x}_\delta, \mathbf{u}_\delta) = (0, 0)$ is an equilibrium of (3.2). Linearizing about this point results in

$$\dot{\mathbf{x}_\delta} = \mathbf{A}(\alpha_0)\mathbf{x}_\delta + \mathbf{B}(\alpha_0)\mathbf{u}_\delta \tag{3.3}$$

where

$$\mathbf{A}(\alpha_0) = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{(\mathbf{x},\mathbf{u})=(\alpha_0,\mathbf{u}_{ss}(\alpha_0))}, \qquad \mathbf{B}(\alpha_0) = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{(\mathbf{x},\mathbf{u})=(\alpha_0,\mathbf{u}_{ss}(\alpha_0))}.$$

Assuming that the pair $(\mathbf{A}(\alpha_0), \mathbf{B}(\alpha_0))$ is controllable, we can now continue to develop a feedback control

$$\mathbf{u}_\delta = -\mathbf{K}(\alpha_0)\mathbf{x}_\delta \tag{3.4}$$

such that $\mathbf{A}(\alpha_0) - \mathbf{B}(\alpha_0)\mathbf{K}(\alpha_0)$ is Hurwitz. Applying this to (3.3) results in the closed-loop linearized system

$$\dot{\mathbf{x}_\delta} = (\mathbf{A}(\alpha_0) - \mathbf{B}(\alpha_0)\mathbf{K}(\alpha_0))\mathbf{x}_\delta.$$

Hence under the control (3.4), which we can write as

$$\mathbf{u}(\mathbf{x}; \alpha_0) = -\mathbf{K}(\alpha_0)(\mathbf{x} - \alpha_0) + \mathbf{u}_{ss}(\alpha_0), \tag{3.5}$$

the closed-loop dynamics will be given by

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, -\mathbf{K}(\alpha_0)(\mathbf{x} - \alpha_0) + \mathbf{u}_{ss}(\alpha_0)) \tag{3.6}$$

and we will have that $\mathbf{x}_\delta \to 0$, or equivalently $\mathbf{x} \to \alpha_0$ in a neighbourhood of $\mathbf{x} = \alpha_0$.

Let us denote the region of attraction of $\mathbf{x} = \alpha_0$ of the closed-loop system (3.6) as $\mathcal{R}_{\alpha_0} \subset D$. Suppose now that we wish to switch the reference signal to $\mathbf{r} = \alpha_1$ at time t_1, where $\alpha_0, \mathbf{x}(t_1) \in \mathcal{R}_{\alpha_1}$. We can repeat the above procedure using the linearization around $(\alpha_1, \mathbf{u}_{ss}(\alpha_1))$ to obtain a new controller

$$\mathbf{u}(\mathbf{x}; \alpha_1) = -\mathbf{K}(\alpha_1)(\mathbf{x} - \alpha_1) + \mathbf{u}_{ss}(\alpha_1)$$

to asymptotically stabilize the closed-loop system about $\mathbf{x} = \alpha_1$.

In this fashion, we can track a reference of a sequence of operating points $\mathbf{r} = \{\alpha_i\}$, so long as $\alpha_{k-1}, \mathbf{x}(t_k) \in \mathcal{R}_{\alpha_k} \subset D$ where t_k denotes the time of the k^{th} switch. This concept is demonstrated in figure 3.1.

We note that because we are only using the point designs in our control law, the controller is discontinuous, which is not ideal in physical implementation. We consider the effects of this in section 5.6.

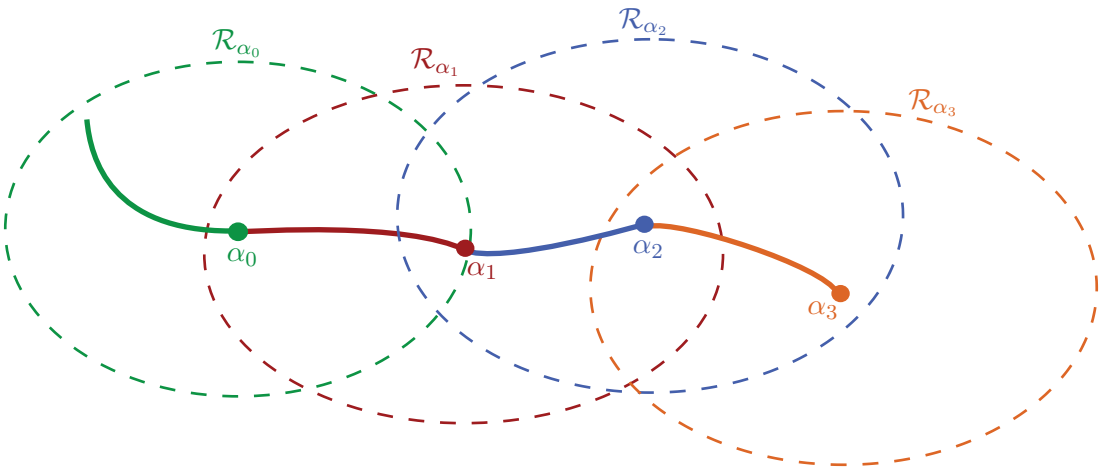


Figure 3.1: Conceptual illustration of the evolution of a trajectory under gain-scheduled control. Each colour corresponds to a separate point design.

Chapter 4

The Quadcopter UAV

The quadcopter is a helicopter with four rotors arranged in a planar fashion. It is a dynamic vehicle with four input forces and six output coordinates, thus having an underactuated degree of two. The input forces are assigned to the torques of each of the rotors. Two of the rotors opposite to each other rotate in a clockwise direction, and the other two rotate in a counter-clockwise direction. It can be controlled by changing the rotor speeds, which provides lift forces experienced by each rotor, as well as the overall torque experienced by the quadcopter itself. This configuration makes the quadcopter a highly maneuverable vehicle capable of tracking a wide range of trajectories.

4.1 Dynamic Model

Our model choice for the quadcopter follows that used by [12, 13, 14]. The primary motions of the quadcopter can be described referring to figure 4.1.

We denote the reference frame relative to the center of mass of the quadcopter by (x_b, y_b, z_b), and an inertial world frame by (x, y, z). The quadcopter is controlled by supplying four torques T_1, T_2, T_3, T_4 to the rotors of the quadcopter, which in turn produces four thrust forces F_1, F_2, F_3, F_4 along the z_b axis. As such, vertical motion can be achieved by varying all of the rotor speeds simultaneously. Motion along the x_b axis is related to a rotation about the y_b axis and is obtained by increasing (or decreasing) the speeds of rotors 1 and 2 and decreasing (or increasing) the speeds of rotors 3 and 4. Similarly, motion along the y_b axis is related to a rotation about the x_b axis and is obtained by increasing (or decreasing) the speeds of rotors 2 and 3 and decreasing (or increasing) the speeds of rotors 1 and 4. Rotation about the z_b axis is a result of the net moment produced by the spinning rotors.

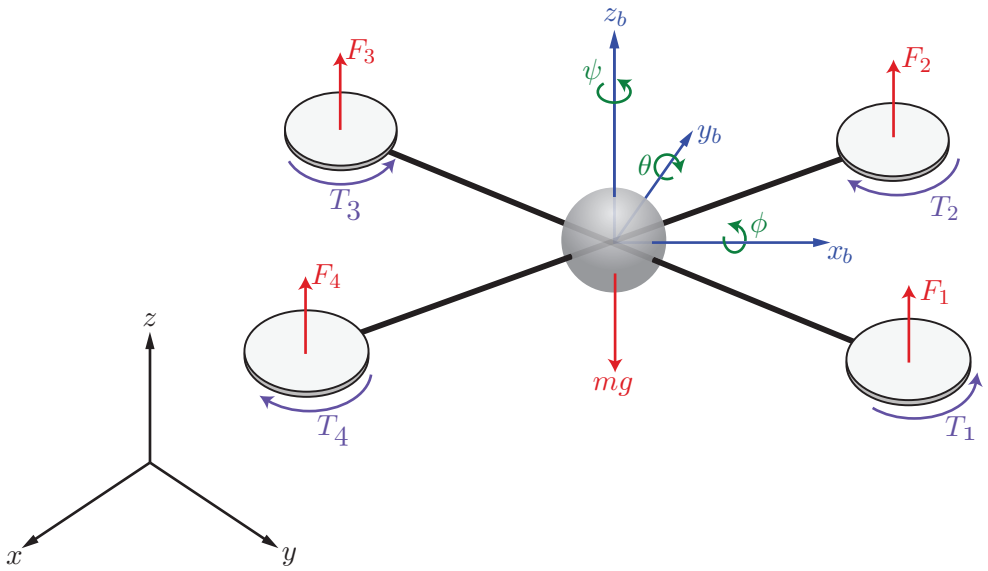


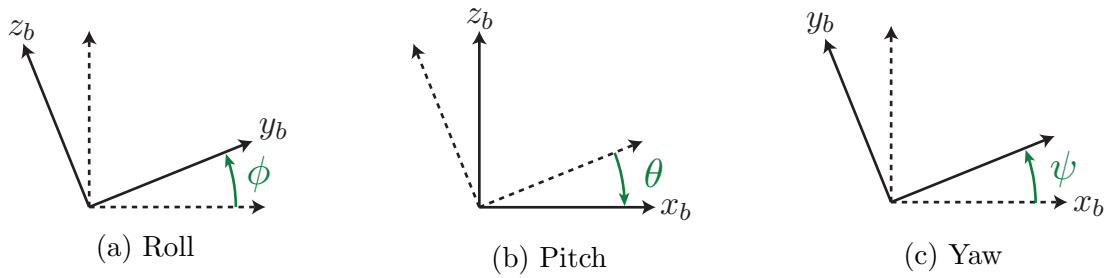
Figure 4.1: Schematic diagram of a quadcopter.

To represent the rotation configuration of the quadcopter relative to the world frame, we choose to use the ZYX Euler angle convention. This choice results in a singularity when $\theta = \pm\frac{\pi}{2}$, which is not problematic for any trajectory we plan as θ will remain small.

The roll, pitch, and yaw angles as they relate to rotations in the $y_b z_b$-plane, $x_b z_b$-plane, and $x_b y_b$-plane, respectfully, can be seen in figure 4.2. From here we can arrive at the necessary rotation matrix Γ, where

$$\Gamma = \begin{pmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi \\ \sin\psi\cos\theta & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{pmatrix}. \tag{4.1}$$

As such, premultiplying a vector in the body frame by Γ will result in the same vector expressed in the world frame. Using Γ, Newton's Second Law and the conservation of



(a) Roll (b) Pitch (c) Yaw

Figure 4.2: Roll, pitch and yaw rotation conventions.

momentum, we arrive at the equations of motion

$$\begin{aligned}
\ddot{x} &= u_1(\cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi) - \frac{k_1}{m}\dot{x},\\
\ddot{y} &= u_1(\sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi) - \frac{k_2}{m}\dot{y},\\
\ddot{z} &= u_1(\cos\theta\cos\phi) - \frac{k_3}{m}\dot{z},\\
\ddot{\theta} &= u_2 - \frac{k_4\ell}{I_1}\dot{\theta},\\
\ddot{\phi} &= u_3 - \frac{k_5\ell}{I_2}\dot{\phi},\\
\ddot{\psi} &= u_4 - \frac{k_6}{I_3}\dot{\psi},
\end{aligned} \tag{4.2}$$

where m is the mass of the quadcopter, g is the gravitational constant, $k_1, k_2, k_3, k_4, k_5, k_6$ are drag coefficients, I_1, I_2, I_3 are the principle moments of inertia of the quadcopter[1], and ℓ is half the distance between two adjacent rotors. The physical values that we use in our simulations are taken from [12] and are listed in appendix A.

The fictitious control inputs u_1, u_2, u_3, u_4 can be thought of as the thrust, pitch input, roll input, and yaw moment, respectfully, and are related to the F_i's by

$$\begin{aligned}
u_1 &= (F_1 + F_2 + F_3 + F_4)/m,\\
u_2 &= \ell(-F_1 - F_2 + F_3 + F_4)/I_1,\\
u_3 &= \ell(-F_1 + F_2 + F_3 - F_4)/I_2,\\
u_4 &= C(F_1 - F_2 + F_3 - F_4)/I_3,
\end{aligned} \tag{4.3}$$

[1]The quadcopter is in fact a symmetrical top (*cf.* [15]), so that $I_1 = I_2$

where C is the force to moment scaling factor. That is, $T_i = CF_i$.

To put the equations of motion into state space form, we introduce the state variables

$$\begin{aligned} x_1 &= x, \quad x_2 = \dot{x}, \\ y_1 &= y, \quad y_2 = \dot{y}, \\ z_1 &= z, \quad z_2 = \dot{z}, \\ \theta_1 &= \theta, \quad \theta_2 = \dot{\theta}, \\ \phi_1 &= \phi, \quad \phi_2 = \dot{\phi}, \\ \psi_1 &= \psi, \quad \psi_2 = \dot{\psi}, \end{aligned}$$

so that (4.2) can be written as the 12-dimensional first order system

$$\begin{aligned} \dot{x_1} &= x_2, \\ \dot{y_1} &= y_2, \\ \dot{z_1} &= z_2, \\ \dot{\theta_1} &= \theta_2, \\ \dot{\phi_1} &= \phi_2, \\ \dot{\psi_1} &= \psi_2, \\ \dot{x_2} &= u_1(\cos\psi_1 \sin\theta_1 \cos\phi_1 + \sin\psi_1 \sin\phi_1) - \frac{k_1}{m} x_2, \\ \dot{y_2} &= u_1(\sin\psi_1 \sin\theta_1 \cos\phi_1 - \cos\psi_1 \sin\phi_1) - \frac{k_2}{m} y_2, \\ \dot{z_2} &= u_1(\cos\theta_1 \cos\phi_1) - \frac{k_3}{m} z_2, \\ \dot{\theta_2} &= u_2 - \frac{k_4 \ell}{I_1} \theta_2, \\ \dot{\phi_2} &= u_3 - \frac{k_5 \ell}{I_2} \phi_2, \\ \dot{\psi_2} &= u_4 - \frac{k_6}{I_3} \psi_2. \end{aligned} \tag{4.4}$$

We can write this more concisely by defining the state vector to be

$$\mathbf{x} = (x_1, y_1, z_1, \theta_1, \phi_1, \psi_1, x_2, y_2, z_2, \theta_2, \phi_2, \psi_2)^T$$

and the control vector

$$\mathbf{u} = (u_1, u_2, u_3, u_4)^T$$

so that the nonlinear dynamics can be more compactly expressed as

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{4.5}$$

with $\mathbf{f}$ continuously differentiable and given by the right-hand side of (4.4).

4.2 Linearization of Quadcopter Dynamics

We now are interested in linearizing the nonlinear dynamics of the quadcopter given in (4.4) about its equilibrium points in order to develop our gain-scheduled controller. To do this, we must first find the equilibrium points of the quadcopter dynamics.

Equilibrium points of (4.4) satisfy

$$0 = \mathbf{f}(\mathbf{x}_{eq}, \mathbf{u}_{eq}), \tag{4.6}$$

where

$$\begin{aligned}\mathbf{x}_{eq} &= (x_{1eq}, y_{1eq}, z_{1eq}, \theta_{1eq}, \phi_{1eq}, \psi_{1eq}, x_{2eq}, y_{2eq}, z_{2eq}, \theta_{2eq}, \phi_{2eq}, \psi_{2eq})^T, \\ \mathbf{u}_{eq} &= (u_{1eq}, u_{2eq}, u_{3eq}, u_{4eq})^T.\end{aligned}$$

Solving (4.6) results in the stationary points

$$\begin{aligned}\mathbf{x}_{eq} \equiv \mathbf{x}_{ss} &= (x_{1ss}, y_{1ss}, z_{1ss}, 0, 0, \psi_{1ss}, 0, 0, 0, 0, 0, 0)^T, \\ \mathbf{u}_{eq} \equiv \mathbf{u}_{ss} &= (g, 0, 0, 0)^T,\end{aligned} \tag{4.7}$$

where $x_{1ss}, y_{1ss}, z_{1ss}, \psi_{1ss} \in \mathbb{R}$ and we introduce the subscript ss to mean *steady state*. This result should be intuitive as it corresponds to the hovering configuration of the quadcopter.

Linearizing the nonlinear dynamics (4.4) about the steady state values (4.7) results in

$$\dot{\mathbf{x}} = \mathbf{A}(\mathbf{x} - \mathbf{x}_{ss}) + \mathbf{B}(\mathbf{u} - \mathbf{u}_{ss}), \tag{4.8}$$

where the Jacobian matrices

$$\mathbf{A} = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{(\mathbf{x},\mathbf{u})=(\mathbf{x}_{ss},\mathbf{u}_{ss})} \equiv \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{ss}, \qquad \mathbf{B} = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{(\mathbf{x},\mathbf{u})=(\mathbf{x}_{ss},\mathbf{u}_{ss})} \equiv \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{ss}$$

are given by

$$\mathbf{A} = \begin{pmatrix} 0_{6\times 6} & \mathbf{I}_6 \\ \Psi(\psi_{1ss}) & \Delta \end{pmatrix}, \tag{4.9}$$

$$\mathbf{B} = \begin{pmatrix} 0_{8\times 4} \\ \mathbf{I}_4 \end{pmatrix}, \tag{4.10}$$

where $0_{m\times n}$ is a $m \times n$ zero matrix, $\mathbf{I}_n$ is the $n \times n$ identity matrix, and $\Psi(\psi_{1ss})$ and Δ are given by

$$\Psi(\psi_{1ss}) = \begin{pmatrix} 0 & 0 & 0 & g\cos\psi_{1ss} & g\sin\psi_{1ss} & 0 \\ 0 & 0 & 0 & g\sin\psi_{1ss} & -g\cos\psi_{1ss} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\Delta = \begin{pmatrix} -k_1/m & 0 & 0 & 0 & 0 & 0 \\ 0 & -k_2/m & 0 & 0 & 0 & 0 \\ 0 & 0 & -k_3/m & 0 & 0 & 0 \\ 0 & 0 & 0 & -k_4\ell/I_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -k_5\ell/I_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & -k_6/I_3 \end{pmatrix}.$$

Thus we see that the linearization depends on the parameter ψ_{1ss} — the steady state value of the yaw angle. Also, we can write (4.8) in terms of the shifted variables

$$\mathbf{x}_\delta \equiv \mathbf{x} - \mathbf{x}_{ss}, \quad \mathbf{u}_\delta \equiv \mathbf{u} - \mathbf{u}_{ss}$$

as

$$\dot{\mathbf{x}_\delta} = \mathbf{A}(\psi_{1ss})\mathbf{x}_\delta + \mathbf{B}\mathbf{u}_\delta \tag{4.11}$$

so that the origin $(\mathbf{x}_\delta, \mathbf{u}_\delta) = (0, 0)$ is an equilibrium point of (4.11).

Note that in the case of output feedback, we will be using the output

$$\mathbf{y} = (x_1, y_1, z_1, \psi_1)^T,$$

which we can express as

$$\mathbf{y} = \mathbf{C}\mathbf{x},$$

where

$$\mathbf{C} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{4.12}$$

so that with $\mathbf{y}_\delta \equiv \mathbf{y} - \mathbf{y}_{ss}$, where $\mathbf{y}_{ss} = (x_{1ss}, y_{1ss}, z_{1ss}, \psi_{1ss})^T$, we can express the system as

$$\begin{aligned} \dot{\mathbf{x}_\delta} &= \mathbf{A}(\psi_{1ss})\mathbf{x}_\delta + \mathbf{B}\mathbf{u}_\delta, \\ \mathbf{y}_\delta &= \mathbf{C}\mathbf{x}_\delta, \end{aligned} \tag{4.13}$$

which is a linear parameter-varying system as described in section 3.2.

4.3 Controllability and Observability of Linearized Quadcopter Dynamics

We now look at the controllability of the *linearized* dynamics of the quadcopter using the procedure outlined in section 2.3. We denote the controllability matrix of the linearized system (4.8) as

$$\mathcal{C}(\mathbf{A}(\psi_{1ss}), \mathbf{B}).$$

It can be verified (*e.g.* through row reduction) that

$$\text{rank } \mathcal{C}(\mathbf{A}(\psi_{1ss}), \mathbf{B}) = 12, \quad \forall\, \psi_{1ss} \in \mathbb{R}.$$

Since $\mathbf{A}$ is a 12×12 matrix, the system is thus controllable by theorem 2.3.1.

For the purpose of output feedback control, we will use $\mathbf{y} = (x_1, y_1, z_1, \psi_1)^T$ as the output variable. This is a common choice in various control laws for the quadcopter. With this choice, we denote the observability matrix of (4.13) as

$$\mathcal{O}(\mathbf{A}(\psi_{1ss}), \mathbf{C}),$$

where $\mathbf{C}$ is given in (4.12). It can be verified that

$$\text{rank } \mathcal{O}(\mathbf{A}(\psi_{1ss}), \mathbf{C}) = 12, \quad \forall\, \psi_{1ss} \in \mathbb{R}$$

and so we also have that the system is observable by theorem 2.3.2. With the guarantees of controllability and observability of the linearized dynamics, we are now prepared to develop the gain-scheduled control law for the quadcopter.

Chapter 5

Gain-Scheduled Control of the Quadcopter

As discussed, the technique of gain scheduling is well suited for linear parameter-varying systems where the varying parameter can be assigned to be the scheduling variable in the control law. This strategy can be applied to nonlinear systems when the linearized dynamics happen to depend on a parameter of interest to the control engineer. An example of such a system is the quadcopter. We have seen in section 4.2 that the linearization of the quadcopter dynamics about any of its equilibrium points results in a linear parameter-varying system depending on the yaw angle of the vehicle. We can take advantage of this model to develop a gain-scheduled controller that switches based on the value of the yaw angle in order to maintain stability.

We will use the linearized dynamics of the quadcopter to develop a series of gain-scheduled control laws that use the yaw angle as the scheduling variable. We will construct proportional (P) and proportional-integral (PI) control laws for use with both state and observer-based output feedback, where the output vector is chosen so that the linearized system is observable for all values of the scheduling parameter. Simulations will be conducted for each control law, tracking trajectories that exhibit the gain-scheduled nature of the controller. Finally, we conclude with an overall analysis of applying the gain scheduling technique to the quadcopter, and propositions for further development of this work.

5.1 Fixed-Gain Control

We will begin with developing the fixed-gain controller that will be used as a basis for the development of the full gain-scheduled controller. We will demonstrate when this controller is appropriate for use, and conditions where gain scheduling can be applied for when it fails.

As discussed in chapter 4, let

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$$

be the nonlinear dynamics of the quadcopter, and let

$$\dot{\mathbf{x}_\delta} = \mathbf{A}\mathbf{x}_\delta + \mathbf{B}\mathbf{u}_\delta$$

be the linearization about the equilibrium point $(\mathbf{x}_{ss}, \mathbf{u}_{ss})$, where

$$\begin{aligned}\mathbf{x}_{ss} &= (x_{1ss}, y_{1ss}, z_{1ss}, 0, 0, \psi_{1ss}, 0, 0, 0, 0, 0, 0)^T,\\ \mathbf{u}_{ss} &= (g, 0, 0, 0)^T\end{aligned}$$

and $\mathbf{x}_\delta = \mathbf{x} - \mathbf{x}_{ss}, \mathbf{u}_\delta = \mathbf{u} - \mathbf{u}_{ss}$. The matrices $\mathbf{A} = \mathbf{A}(\psi_{1ss})$ and $\mathbf{B}$ are given by (4.9) and (4.10), respectively. As done in section 2.5, we can use the controller

$$\mathbf{u} = -\mathbf{K}(\mathbf{x} - \mathbf{x}_{ss}) + \mathbf{u}_{ss}$$

to stabilize the quadcopter at $\mathbf{x}_{ss}$, where $\mathbf{K}$ is designed so that $\mathbf{A} - \mathbf{B}\mathbf{K}$ is Hurwitz. Note that the linearized dynamics do not depend on the spatial coordinates $(x_{1ss}, y_{1ss}, z_{1ss})$ of the steady state. As such, the gain matrix $\mathbf{K}$ only depends on the yaw angle configuration ψ_{1ss} of the quadcopter, and so we can use this controller to track a reference

$$\mathbf{r}(t) = (x_{1ref}(t), y_{1ref}(t), z_{1ref}(t), \psi_{1ss})^T,$$

where ψ_{1ss} is a constant yaw angle. For any frozen value of the reference signal $\mathbf{r}$ we want to stabilize the quadcopter at the steady state

$$\mathbf{x}_{ss} = \mathbf{C}^T\mathbf{r},$$

with $\mathbf{C}$ given by (4.12). Figure 5.1 gives a block diagram of the control scheme. This controller can be used to track trajectories in which there is little variation in the yaw angle of the quadcopter. Figure 5.2 illustrates the tracking of a circular trajectory in the x_1y_1-plane with a constant yaw angle $\psi_{1ss} = 0$.

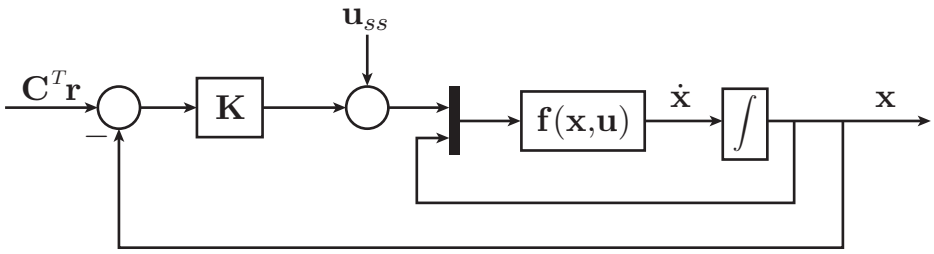


Figure 5.1: Block diagram of the proportional fixed-gain controller.

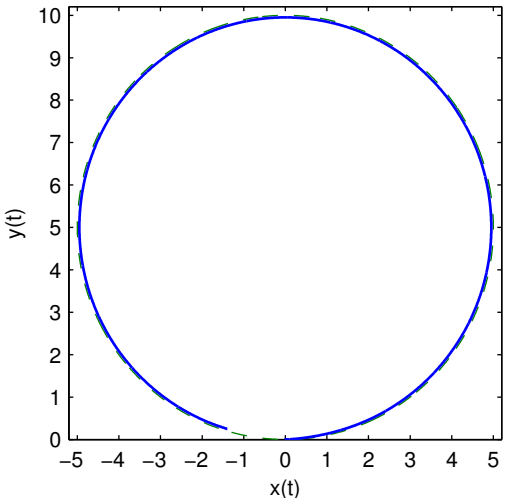
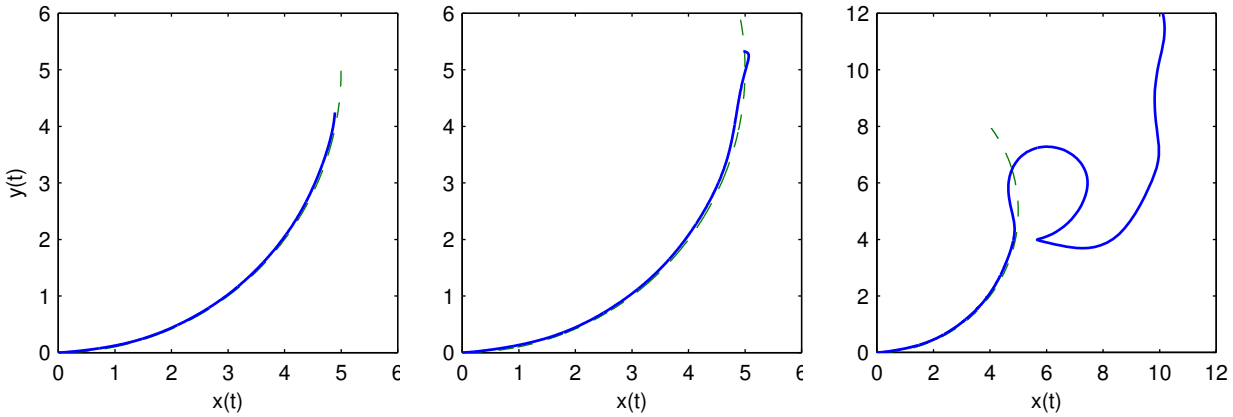


Figure 5.2: Tracking of circular trajectory with constant yaw under fixed-gain control.

We now consider what happens if we use this controller to track a reference in which the yaw angle deviates away from the value about which the linearization was taken. The controller is only valid in a neighbourhood of ψ_{1ss}, and so if the difference $|\psi_1(t) - \psi_{1ss}|$ becomes too large we will no longer be guaranteed stability. We illustrate this in figure 5.3 by repeating the tracking of the circular trajectory under the fixed-gain controller using the linearization about $\psi_{1ss} = 0$, but this time we prescribe the yaw angle $\psi_1(t)$ to remain tangential to the curve so that the quadcopter is always facing in the direction it is moving.



(a) 25% of circle (b) 30% of circle (c) 35% of circle

Figure 5.3: Attempt to track the circular trajectory with varying yaw angle under fixed-gain control. The heading of the quadcopter is set to be tangential to the reference curve for all time.

We can see that the quadcopter begins to lose stability after completing 30% of the trajectory, and becomes unstable after the yaw angle has deviated away from ψ_{1ss} by approximately $2\pi/3$ radians. Clearly the fixed-gain controller cannot be relied upon when undergoing trajectories in which $\psi_1(t)$ substantially varies. To correct this issue, we can switch to a new controller that has been developed for the current yaw angle of the quadcopter before the dynamics become unstable. This is the idea behind the gain-scheduled controller.

5.2 Gain-Scheduled Proportional Control with State Feedback

We are now ready to develop the gain-scheduled controller. We begin by assuming that the entire state $\mathbf{x}$ is available on-line for feedback.

To determine how frequently the controller will switch, we decide on a tolerance ψ_{tol} for the yaw angle. In most of the simulations we conduct, we will be using $\psi_{tol} = \pi/6$ simply to illustrate that a smaller tolerance is not required, although can certainly be implemented.

We next use the linearization of the quadcopter dynamics about the initial yaw angle $\psi_1(0) \equiv \psi_0^*$ to develop the control law

$$\mathbf{u}(\mathbf{x}; \psi_0^*) = -\mathbf{K}(\psi_0^*)(\mathbf{x} - \mathbf{C}^T\mathbf{r}(t)) + \mathbf{u}_{ss},$$

which will serve to track the reference signal $\mathbf{r}(t)$ near ψ_0^*. Once the yaw angle has deviated away from the specified tolerance, or

$$|\psi_1(t) - \psi_0^*| > \psi_{tol},$$

we switch to the control

$$\mathbf{u}(\mathbf{x}; \psi_1^*) = -\mathbf{K}(\psi_1^*)(\mathbf{x} - \mathbf{C}^T\mathbf{r}(t)) + \mathbf{u}_{ss},$$

where ψ_1^* is the value of the yaw angle at the time of the switch. We repeat this procedure as often as necessary and in this manner construct a sequence of control laws

$$\mathbf{u}(\mathbf{x}; \psi_i^*) = -\mathbf{K}(\psi_i^*)(\mathbf{x} - \mathbf{C}^T\mathbf{r}(t)) + \mathbf{u}_{ss},$$

which together comprise the gain-scheduled control law. The control scheme is illustrated in figure 5.4

At each step, the gain matrix $\mathbf{K}(\psi_i^*)$ is designed to render the matrix $\mathbf{A}(\psi_i^*) - \mathbf{BK}$ Hurwitz. This is done by solving the algebraic Ricatti equation (2.8) as discussed in section 2.4 in order to find the control with associated gain matrix $\mathbf{K}$ that minimizes the performance cost (2.6). The matrices $\mathbf{Q}$ and $\mathbf{R}$ which we will use for simulation purposes are diagonal and of the form

$$\begin{aligned} \mathbf{Q} &= \text{diag}(w_{x1}, w_{y1}, w_{z1}, 1, 1, w_{\psi 1}, 1, 1, 1, 1, 1, 1), \\ \mathbf{R} &= \text{diag}(w_{u1}, w_{u2}, w_{u3}, w_{u4}), \end{aligned} \tag{5.1}$$

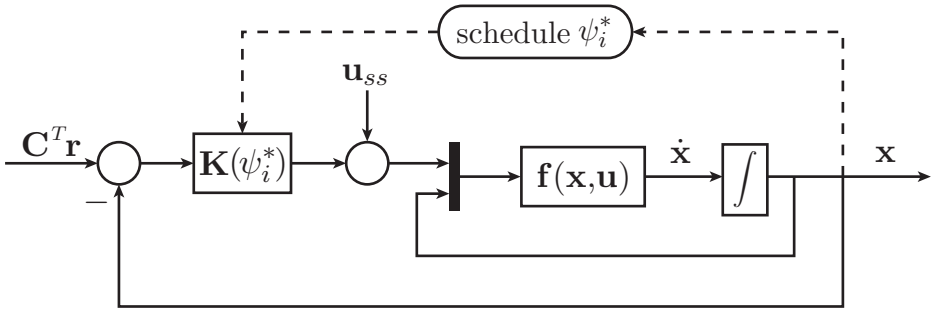


Figure 5.4: Block diagram of the proportional gain-scheduled controller.

where $w_{x1}, w_{y1}, w_{z1}, w_{\psi 1}$, represents the amount of effort to be applied to the states x_1, y_1, z_1, ψ_1, in the linear quadratic regulator (2.6). Similarly, $w_{u1}, w_{u2}, w_{u3}, w_{u4}$ represent the efforts to be applied to each of the control inputs u_1, u_2, u_3, u_4. In most of the simulations we will be using $w_{ui} = 0.001$ for each i, so that

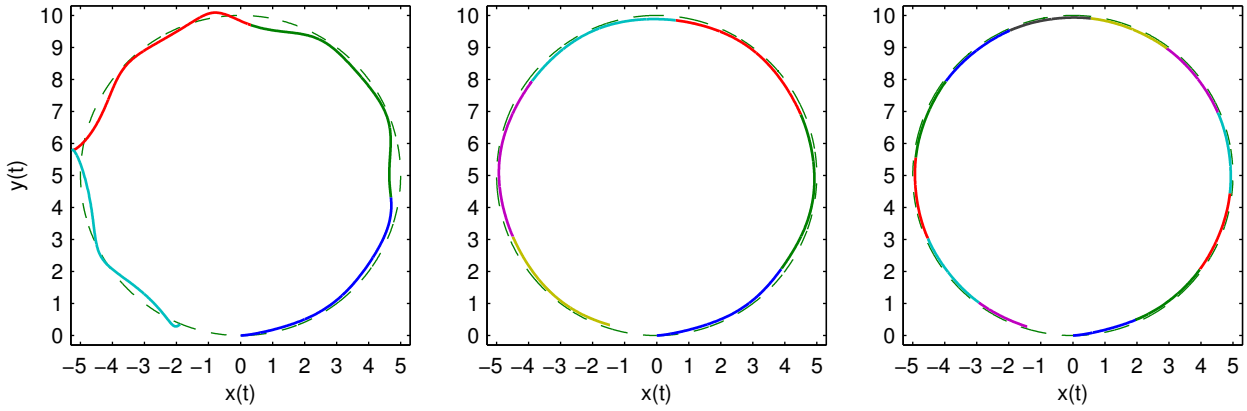
$$\mathbf{R} = \text{diag}(0.001, 0.001, 0.001, 0.001).$$

These small weightings correspond to little restriction on the actual control inputs. If we were interested in restricting the amount of effort that could be applied by the control inputs we could increase these values. The effect of doing so is considered in section 5.6.

To demonstrate how this control law is superior to the fixed-gain controller developed in section 5.1 we repeat the circular trajectory in which the yaw angle remains tangential to the curve at all times. Figure 5.5 shows the results of the quadcopter operating under a gain-scheduled controller with tolerances on the yaw angle set to $\psi_{tol} = \pi/2, \pi/3$, and $\pi/6$. The switching of the controller is depicted by the different colours along the trajectory.

We can see that the performance of the controller increases as ψ_{tol} decreases, or the switching becomes more frequent. The quadcopter does a much better job of tracking the trajectory when ψ_{tol} is decreased from $\pi/2$ to $\pi/3$, however the improvement from the latter is less noticeable when it is decreased to $\pi/6$.

One advantage that the quadcopter has over other UAVs is its ability to vary its yaw angle separately from the desired motion in x, y, z space. We have seen this already in figures 5.2 and 5.5, where a circular trajectory was followed while constantly facing one direction under the fixed-gain controller and facing its direction of travel under the gain-scheduled controller.



(a) $\psi_{tol} = \pi/2$ (b) $\psi_{tol} = \pi/3$ (c) $\psi_{tol} = \pi/6$

Figure 5.5: Tracking of the circular trajectory using gain scheduling with various switch tolerances. The colour of the trajectory changes when the controller switches.

We will further exhibit this feature using the gain-scheduled controller by following a helix trajectory, where the quadcopter will complete 3 counter-clockwise rotations in the x_1y_1-plane while traveling upwards in the z_1 direction with a constant velocity. However, we will prescribe the quadcopter itself to rotate 6 times about the z_b axis, so that $\psi_1(t)$ varies from 0 to 12π. Figure 5.6 illustrates the trajectory along with the switching of the controller, and figure 5.7 compares the performance of the states $x_1(t), y_1(t), z_1(t), \psi_1(t)$ against the reference signal.

As can be seen, the controller succeeds in tracking the reference. However, we can see from the plots that there does exist steady state error when the input is linear. This is to be expected as we have only applied a corrective effort that is proportional to the error.

Finally, to get a full appreciation for the effectiveness of the gain-scheduled controller over the fixed-gain controller, we will conduct a simulation in which the quadcopter is to track a Lissajous curve in the x_1y_1-plane. Such curves can be parameterized as

$$\begin{aligned} x_1(t) &= A\sin(at + \delta), \\ y_1(t) &= B\sin(bt), \end{aligned}$$

for $A, B, a, b, \delta \in \mathbb{R}$.[1] In our simulations, we will use $A = 10, B = 5, a = 2, b = 3, \delta = 0$.

[1]When a/b is rational, the resulting trajectory will be a closed curve. Otherwise, it will be a plane-filling curve.

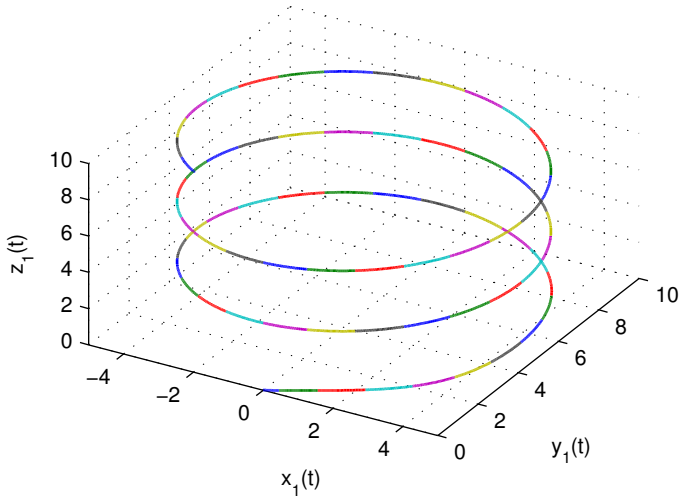


Figure 5.6: Tracking of a helix trajectory using gain scheduling. The quadcopter is prescribed to complete 6 rotations about its z_b axis during this time. Colours are used to indicate the switching of the controller.

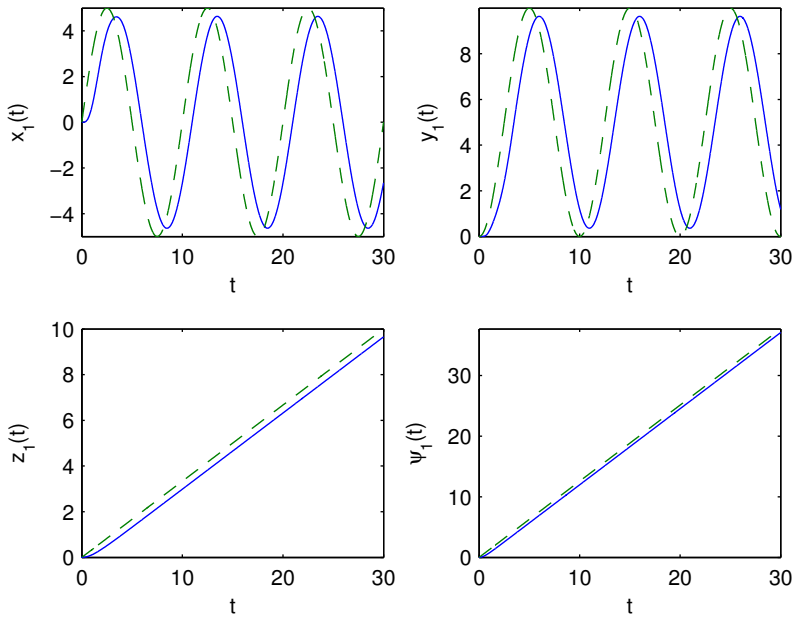


Figure 5.7: Comparing values of the state (solid blue) to the reference (dashed green) of the helix trajectory using gain scheduling.

As the quadcopter tracks the trajectory, we will require that the yaw angle remains tangent to the curve at all times. In such a trajectory, the rate at which the controller must switch is no longer constant. We further subject the quadcopter to multiple step-inputs in the z_1 direction to observe its response.

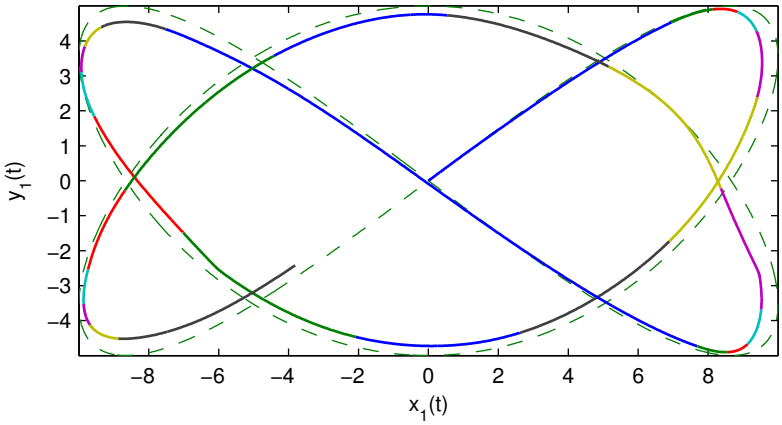


Figure 5.8: Tracking of the Lissajous trajectory using gain scheduling, while simultaneously being subjected to step inputs in the z_1 direction. The heading of the quadcopter is set to remain tangential to the curve for all time.

Figure 5.8 depicts the Lissajous curve tracked by the quadcopter. It is clear that along parts of the trajectory where there is little change in the yaw angle, the controller does not need to switch. An update of the gain matrix $\mathbf{K}$ only occurs when the yaw angle has deviated away from the previous value upon which the dynamics were linearized about by the specified value of ψ_{tol}, in this case $\psi_{tol} = \pi/6$. Similarly, switching is much more frequent when the yaw angle varies more significantly, such as around the corners of the Lissajous curve.

In figure 5.9 we can see how well the quadcopter tracks the individual reference inputs. In particular, we see that it responds well to the step inputs. In the simulation, the quadcopter undergoes step sizes of 4 meters and 8 meters and is able to maintain stability under state feedback.

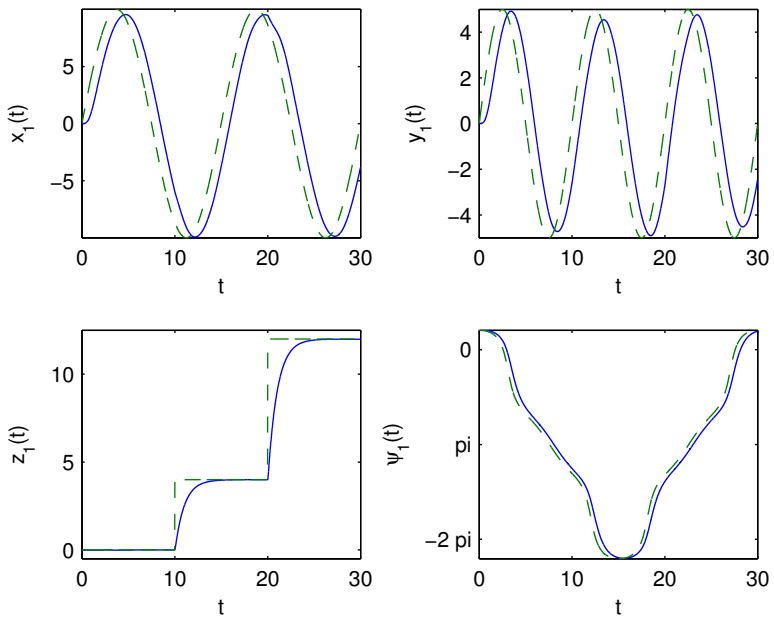


Figure 5.9: Tracking of Lissajous trajectory using gain scheduling.

5.3 Gain-Scheduled Proportional-Integral Control with State Feedback

We now extend the proportional gain-scheduled controller with state feedback developed in section 5.2 by implementing integral action as outlined in section 2.6. To do this, we introduce the integrator

$$\dot{\sigma} = \mathbf{y} - \mathbf{r}, \tag{5.2}$$

where $\mathbf{y} = (x_1, y_1, z_1, \psi_1)$ is a subset of the measurable state vector $\mathbf{x}$ and has the same dimension as the control vector $\mathbf{u} \in \mathbb{R}^4$. Note that we can write $\mathbf{y} = \mathbf{C}\mathbf{x}$, with the matrix $\mathbf{C}$ given by (4.12). We now augment (5.2) with the dynamics of the quadcopter and aim to develop a feedback control $\mathbf{u}$ to stabilize the augmented system

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}), \\ \dot{\sigma} &= \mathbf{C}\mathbf{x} - \mathbf{r} \end{aligned} \tag{5.3}$$

at the steady state $(\mathbf{x}_{ss}, \sigma_{ss})$, where σ_{ss} produces the desired $\mathbf{u}_{ss}$. Here $\mathbf{w} \in \mathbb{R}^l$ is a constant vector of unknown perturbations and disturbances, and it is assumed that for each pair

$(\mathbf{r}, \mathbf{w}) \in D_{\mathbf{r}} \times D_{\mathbf{w}}$ there is a corresponding pair $(\mathbf{x}_{ss}, \mathbf{u}_{ss})$ satisfying

$$0 = \mathbf{f}(\mathbf{x}_{ss}, \mathbf{u}_{ss}, \mathbf{w}),$$
$$0 = \mathbf{C}\mathbf{x}_{ss} - \mathbf{r}$$

for any frozen value of the reference signal $\mathbf{r}(t)$.

As shown in section 2.6, we can use the control

$$\mathbf{u} = -\mathbf{K}_1(\mathbf{x} - \mathbf{x}_{ss}) - \mathbf{K}_2\sigma$$

or since at any particular moment in time we have that $\mathbf{x}_{ss} = \mathbf{C}^T\mathbf{r}$,

$$\mathbf{u} = -\mathbf{K}_1(\mathbf{x} - \mathbf{C}^T\mathbf{r}) - \mathbf{K}_2\sigma$$

to stabilize (5.3) at $(\mathbf{x}_{ss}, \sigma_{ss})$ where $\mathbf{K}_1 \in \mathbb{R}^{4\times 12}, \mathbf{K}_2 \in \mathbb{R}^{4\times 4}$, and

$$\mathcal{K} = \begin{pmatrix} \mathbf{K}_1 & \mathbf{K}_2 \end{pmatrix}$$

is designed so that $\mathcal{A}(\psi_{1ss}) - \mathcal{B}\mathcal{K}$ is Hurwitz, with

$$\mathcal{A} = \begin{pmatrix} \mathbf{A}(\psi_{1ss}) & 0 \\ \mathbf{C} & 0 \end{pmatrix}, \qquad \mathcal{B} = \begin{pmatrix} \mathbf{B} \\ 0 \end{pmatrix}$$

and the matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$ given by (4.9),(4.10),(4.12), respectfully. The control scheme with integral action is represented in figure 5.10.

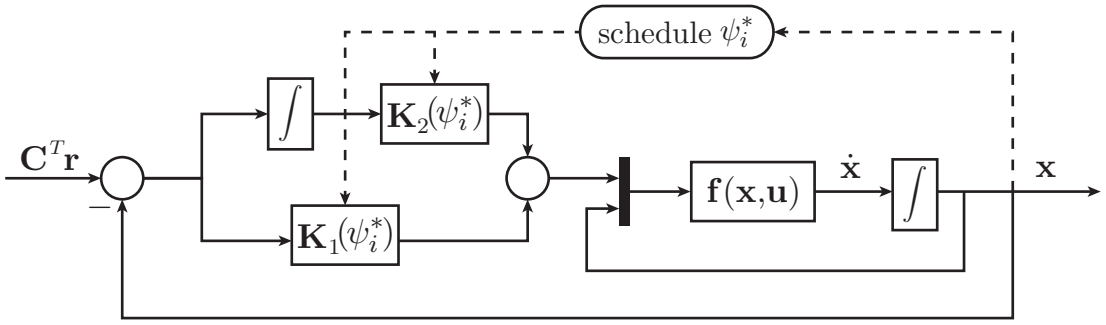


Figure 5.10: Block diagram of the gain-scheduled controller with integral action.

Note that in the case of integral action, we are now dealing with the augmented state vector

$$\xi = \begin{pmatrix} \mathbf{x} \\ \sigma \end{pmatrix}$$

so that when using the LQR approach discussed in section 2.4 to obtain a suitable gain matrix $\mathcal{K}$, we can write the cost (2.6) as

$$J = \int_0^\infty \left(\xi^T \mathbf{Q}_\xi \xi + \mathbf{u}^T \mathbf{R} \mathbf{u}\right)\, dt, \tag{5.4}$$

with the block matrix Q_ξ given by

$$\mathbf{Q}_\xi = \begin{pmatrix} \mathbf{Q}_\mathbf{x} & 0 \\ 0 & \mathbf{Q}_\sigma \end{pmatrix}.$$

As before, the matrix $\mathbf{Q}_\mathbf{x} \in \mathbb{R}^{12\times 12}$ can be thought as a weighting for the components of the state vector $\mathbf{x}$ taking the same form as (5.1), and the matrix $\mathbf{Q}_\sigma \in \mathbb{R}^{4\times 4}$ a weighting for the integral part of the controller. For simulation purposes, we will use

$$\mathbf{Q}_\sigma = w_\sigma \cdot \mathbf{I}_4$$

so that each component of the integrator receives equal weighting $w_\sigma \in \mathbb{R}$. The performance cost (5.4) can then be expressed

$$\begin{aligned} J &= \int_0^\infty \begin{pmatrix} \mathbf{x}^T & \sigma^T \end{pmatrix} \begin{pmatrix} \mathbf{Q}_\mathbf{x} & 0 \\ 0 & \mathbf{Q}_\sigma \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \sigma \end{pmatrix} + \mathbf{u}^T \mathbf{R} \mathbf{u}\, dt \\ &= \int_0^\infty \mathbf{x}^T \mathbf{Q}_\mathbf{x} \mathbf{x} + \sigma^T \mathbf{Q}_\sigma \sigma + \mathbf{u}^T \mathbf{R} \mathbf{u}\, dt, \end{aligned}$$

from which we can obtain the desired gain matrix $\mathcal{K}$ as discussed in section 2.4.

To illustrate the effect of the implementation of integral action, we repeat the tracking of the helix and Lissajous curves as done in section 5.2. Figure 5.11 shows the result of tracking the helix trajectory using the same parameters and yaw tolerance $\psi_{tol} = \pi/6$ as before but under the gain-scheduled PI controller. Comparing this to the results in figure 5.7, we can see that the integral component of the control successfully eliminates the steady state error in tracking the linear reference signals that we saw with the gain-scheduled P controller. However, even with a relatively small weight of $w_\sigma = 0.1$ as used in the simulation, it does result in overshoot as can be seen in the plots for $x_1(t)$ and $y_1(t)$.

Figure 5.12 shows the results of using the control to track the Lissajous trajectory. In this case we increase the weight of the integrator to $w_\sigma = 1$. Comparing this with figure 5.9, we can again see the controller's effort to eliminate the steady state error when the trajectory is roughly linear. We can also clearly observe overshoot in the tracking of the step inputs to $z_1(t)$ but settling with no steady state error.

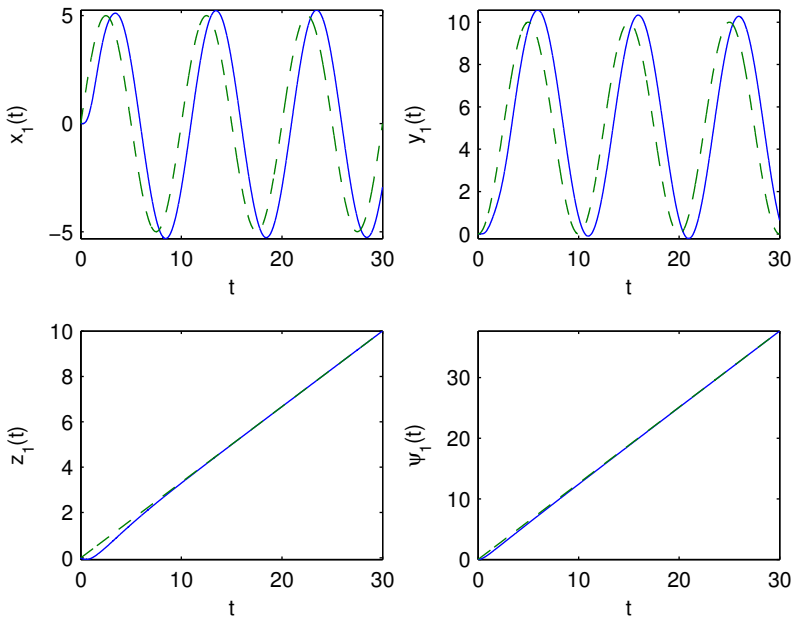


Figure 5.11: Tracking of helix trajectory using gain scheduling with integral action.

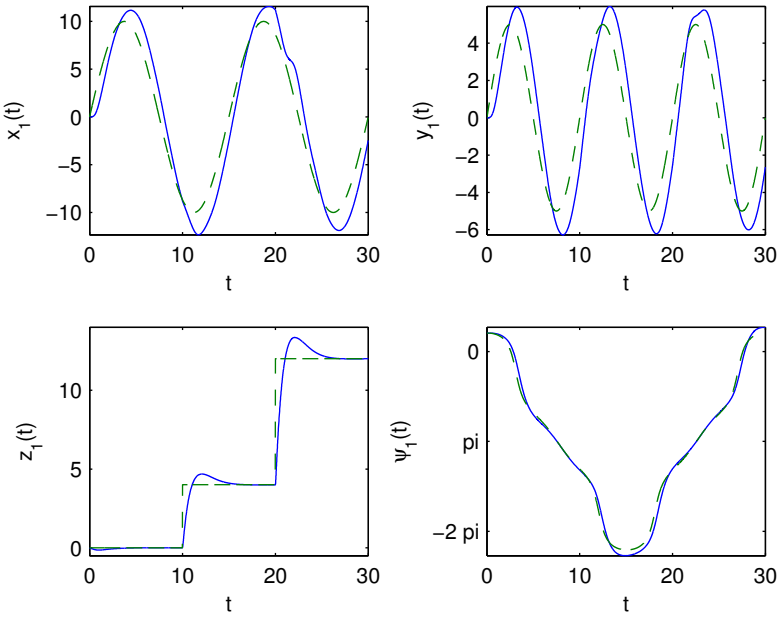


Figure 5.12: Tracking of Lissajous trajectory using gain scheduling with integral action.

5.4 Gain-Scheduled Proportional Control with Output Feedback

The controllers developed in sections 5.2 and 5.4 work well, however they rely on the entire state vector being available on-line. When using the control algorithm on an actual physical system, it may not be the case that we are capable of measuring all 12 states of the quadcopter at each time step. Additionally, even if we could measure each state, we would be faced with measurement noise which could affect the performance of the controller which was designed around a nominal model. As such, we may want to only use an output which is a subset of the state. Specifically, we would like to use an output from which we can both reliably obtain accurate measurements and deduce information about the other states.

A natural choice for the output $\mathbf{y}$ would be the position and the yaw angle of the quadcopter, so that $\mathbf{y} = (x_1, y_1, z_1, \psi_1)^T$. Using appropriate sensors, it is easy to obtain accurate measurements of these state components. Further, as shown in section 4.3, the linearized model of the quadcopter is observable under this output. That is, we can introduce an exogenous variable $\hat{\mathbf{x}}$ which will converge to the actual state $\mathbf{x}$ and use it in our control law. Further, using ψ_1 in our measured output allows us to maintain accuracy in our gain scheduling algorithm which relies on accurate measurements of the yaw angle.

Using this output, our system now takes the form

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}),\\ \mathbf{y} &= \mathbf{C}\mathbf{x},\end{aligned}$$

where $\mathbf{C}$ is given by (4.12). We can no longer use the entire state vector in our control algorithm, and so introduce an observer $\hat{\mathbf{x}}$ to estimate the state $\mathbf{x}$. In section 2.2 we used the linearized dynamics of the nonlinear system in our model for the observer. Here, we will extend the observer to include a copy of the full nonlinear dynamics of the quadcopter. Thus, we choose our observer to be

$$\dot{\hat{\mathbf{x}}} = \mathbf{f}(\hat{\mathbf{x}}, \mathbf{u}) + \mathbf{L}(\mathbf{y} - \mathbf{C}\hat{\mathbf{x}}),$$

where $\mathbf{f}(\hat{\mathbf{x}}, \mathbf{u})$ is the non-linear quadcopter dynamics evaluated at the estimate $\hat{\mathbf{x}}$. Augmenting this with the state equation gives

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}),\\ \dot{\hat{\mathbf{x}}} &= \mathbf{f}(\hat{\mathbf{x}}, \mathbf{u}) + \mathbf{L}\mathbf{C}(\mathbf{x} - \hat{\mathbf{x}}).\end{aligned}$$

Linearizing about the desired steady state values $(\mathbf{x}, \hat{\mathbf{x}}) = (\mathbf{x}_{ss}, \mathbf{x}_{ss})$ results in

$$\begin{aligned}\dot{\mathbf{x}}_\delta &= \mathbf{A}\mathbf{x}_\delta + \mathbf{B}\mathbf{u}_\delta,\\ \dot{\hat{\mathbf{x}}}_\delta &= \mathbf{A}\hat{\mathbf{x}}_\delta + \mathbf{B}\mathbf{u}_\delta + \mathbf{L}\mathbf{C}(\mathbf{x}_\delta - \hat{\mathbf{x}}_\delta).\end{aligned}$$

Notice that we recover the same structure as the linear observer in (2.4).

We can not choose $\mathbf{u}_\delta = -\mathbf{K}\mathbf{x}_\delta$ as we have done before since we do not have on-line access to full state feedback in the case of output feedback. As such, we instead use the estimate and choose

$$\mathbf{u}_\delta = -\mathbf{K}\hat{\mathbf{x}}_\delta.$$

The closed-loop linearized system then takes the form

$$\begin{aligned}\dot{\mathbf{x}}_\delta &= \mathbf{A}\mathbf{x}_\delta - \mathbf{B}\mathbf{K}\hat{\mathbf{x}}_\delta,\\ \dot{\hat{\mathbf{x}}}_\delta &= (\mathbf{A} - \mathbf{B}\mathbf{K})\hat{\mathbf{x}}_\delta + \mathbf{L}\mathbf{C}(\mathbf{x}_\delta - \hat{\mathbf{x}}_\delta).\end{aligned}$$

As done in section 2.3, we can introduce the error in our estimate $\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}}$, or equivalently $\mathbf{e} = \mathbf{x}_\delta - \hat{\mathbf{x}}_\delta$, transforming the closed-loop dynamics into the form

$$\begin{aligned}\dot{\mathbf{x}}_\delta &= (\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}_\delta + \mathbf{B}\mathbf{K}\mathbf{e},\\ \dot{\mathbf{e}} &= (\mathbf{A} - \mathbf{L}\mathbf{C})\mathbf{e},\end{aligned}$$

which we can write as

$$\begin{pmatrix}\dot{\mathbf{x}}_\delta\\ \dot{\mathbf{e}}\end{pmatrix} = \begin{pmatrix}\mathbf{A} - \mathbf{B}\mathbf{K} & \mathbf{B}\mathbf{K}\\ 0 & \mathbf{A} - \mathbf{L}\mathbf{C}\end{pmatrix}\begin{pmatrix}\mathbf{x}_\delta\\ \mathbf{e}\end{pmatrix}. \tag{5.5}$$

Since the matrix above is upper-triangular, we can design $\mathbf{K}$ and $\mathbf{L}$ so that the matrices $\mathbf{A} - \mathbf{B}\mathbf{K}$ and $\mathbf{A} - \mathbf{L}\mathbf{C}$ are Hurwitz. As such, the matrix in (5.5) will have eigenvalues with negative real part and so we will have that $\mathbf{e} \to 0$ and $\mathbf{x}_\delta \to 0$, or equivalently $\hat{\mathbf{x}} \to \mathbf{x}$ and $\mathbf{x} \to \mathbf{x}_{ss}$.

As such, our output feedback controller is

$$\mathbf{u} = -\mathbf{K}(\hat{\mathbf{x}} - \mathbf{x}_{ss}) + \mathbf{u}_{ss}$$

coupled with the observer. Then, with $\mathbf{x}_{ss} = \mathbf{C}^T\mathbf{r}$ for tracking of the reference $\mathbf{r}(t) = (x_{1ref}(t), y_{1ref}(t), z_{1ref}(t), \psi_{1ref}(t))^T$, and $\mathbf{y} = \mathbf{C}\mathbf{x}$ to emphasize the use of output feedback, the controller takes the form

$$\begin{aligned}\mathbf{u} &= -\mathbf{K}(\hat{\mathbf{x}} - \mathbf{C}^T\mathbf{r}(t)) + \mathbf{u}_{ss},\\ \dot{\hat{\mathbf{x}}} &= \mathbf{f}(\hat{\mathbf{x}}, \mathbf{u}) + \mathbf{L}(\mathbf{y} - \mathbf{C}\hat{\mathbf{x}}).\end{aligned}$$

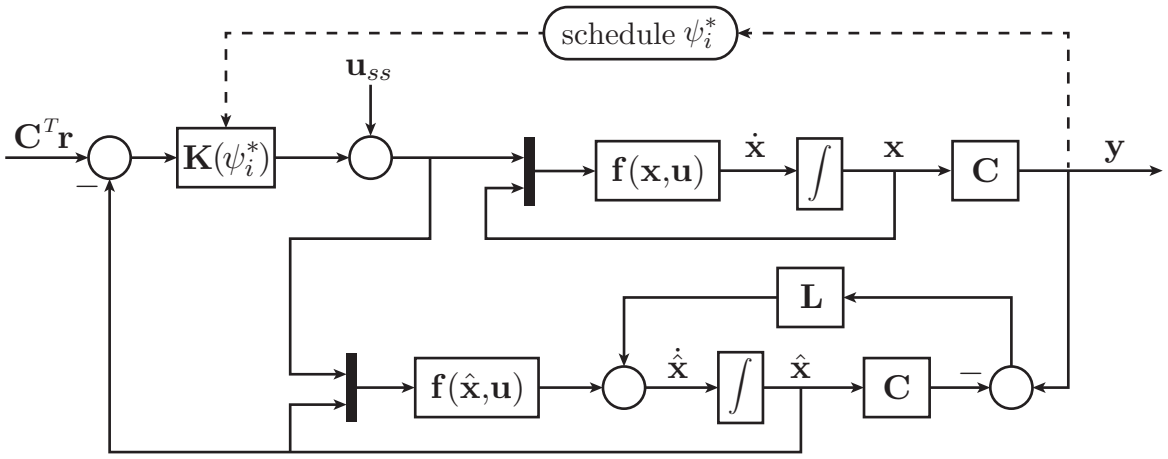


Figure 5.13: Block diagram of the gain-scheduled controller under output feedback.

We illustrate the control law in figure 5.13.

We now repeat the simulations conducted for the gain-scheduled controllers using state feedback and compare the results. We first have the quadcopter track the helix trajectory. To see the effect of the observer, we initialize the simulation with an initial condition for the estimate that is different from that of the state. Results for this simulation are shown in figure 5.14

Comparing with figure 5.6, we can see that the plots are almost identical. The only difference in the trajectories is near the beginning. In the case of output feedback, we are relying on values produced from the observer which will converge to the state as time increases. We can demonstrate this by plotting the value of the estimate as well as the value of the true state as shown in figure 5.15.

We can see that the observer does a good job at converging to the state. In theory, placing the closed-loop poles assigned to the observer farther left in the open left-hand plane will increase the rate of convergence of the observer to the state. However, placing them too far can result in the "peaking" phenomenon, which can be unsafe in physical implementation of the controller and lead to instability [16]. Further, too high of a gain $\mathbf{L}$ will amplify any unmodeled noise that the system experiences which can also lead to instability [6]. To account for this uncertainty resulting from noise, we may wish to implement another observer such as the Kalman filter [3], which takes noise into account in finding an optimal gain.

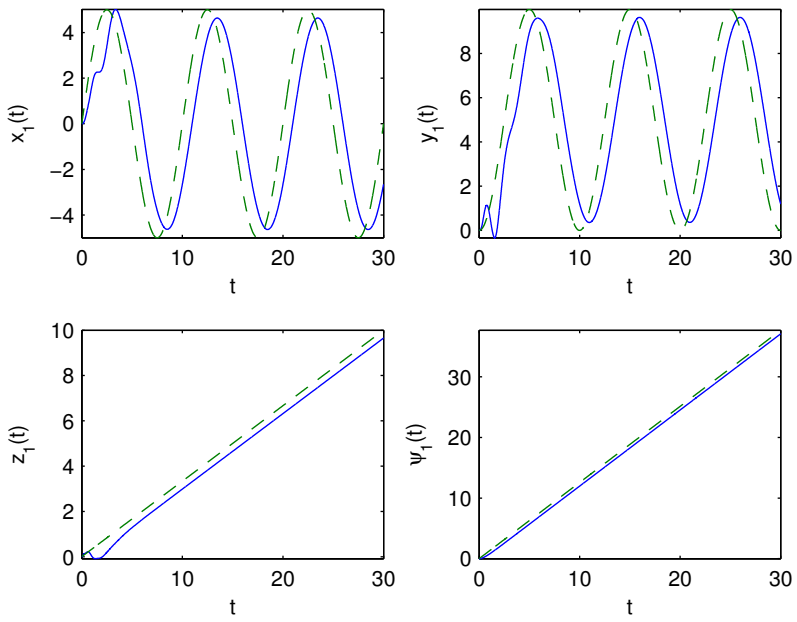


Figure 5.14: Tracking of helix trajectory using gain scheduling under output feedback.

Finally, we repeat the tracking of the Lissajous trajectory. The results of using the proportional gain-scheduled controller with output feedback are shown in figure 5.16. Again, we observe similar behaviour as seen in the previous simulation. That is, there is a small amount of deviation from the desired trajectory at the beginning of the simulation, but as time increases we recover the results in the full state feedback simulation in figure 5.9. We can see how the observer successfully tracks the state in figure 5.17.

5.5 Gain-Scheduled Proportional-Integral Control with Output Feedback

We now aim to improve the controller developed in the previous section by implementing integral action into the control law. We saw in section 5.3 that implementing integral action eliminated the steady state error when the quadcopter was subjected to linear reference signals. We expect to see the same result in the case of observer-based output feedback.

As before, we take the controlled and measured output to be $\mathbf{y} = (x_1, y_1, z_1, \psi_1)^T$ and

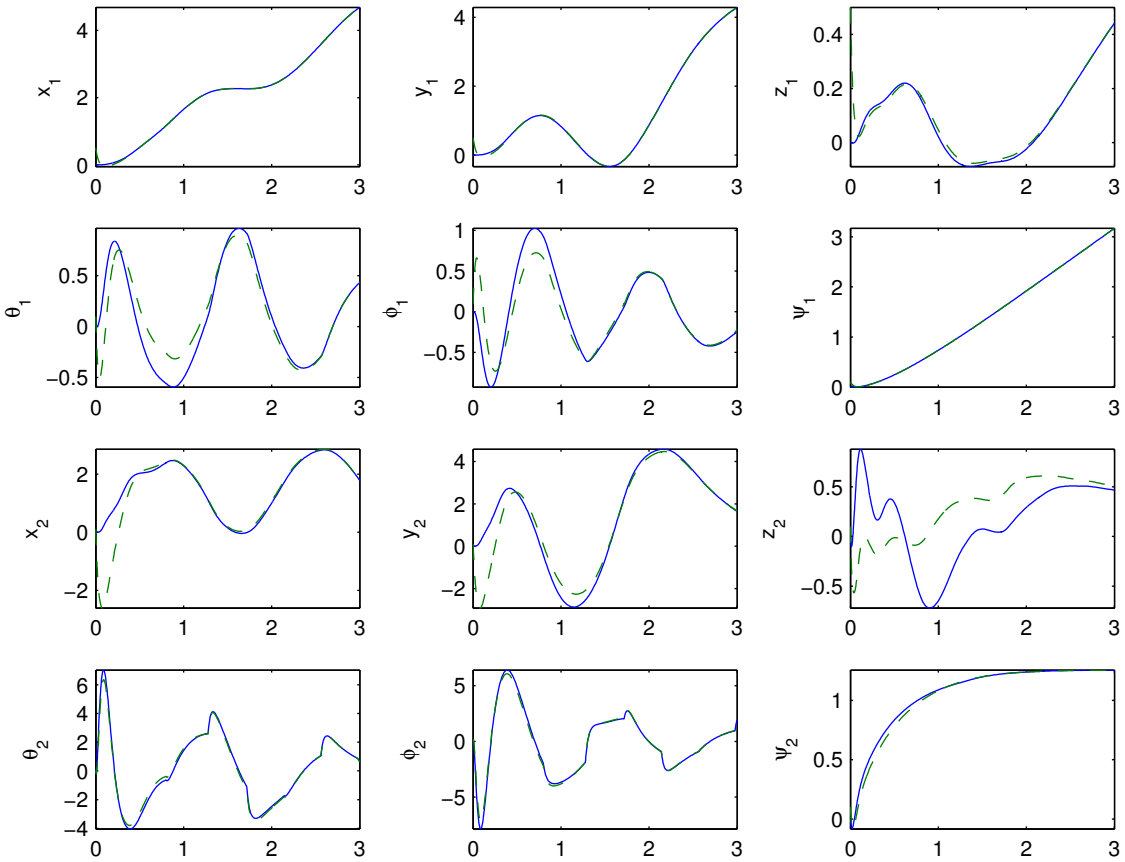


Figure 5.15: Comparing the observer $\hat{\mathbf{x}}$ (dashed green) with the actual state $\mathbf{x}$ (solid blue) from $t = 0$ to $t = 3$ for tracking of the helix trajectory.

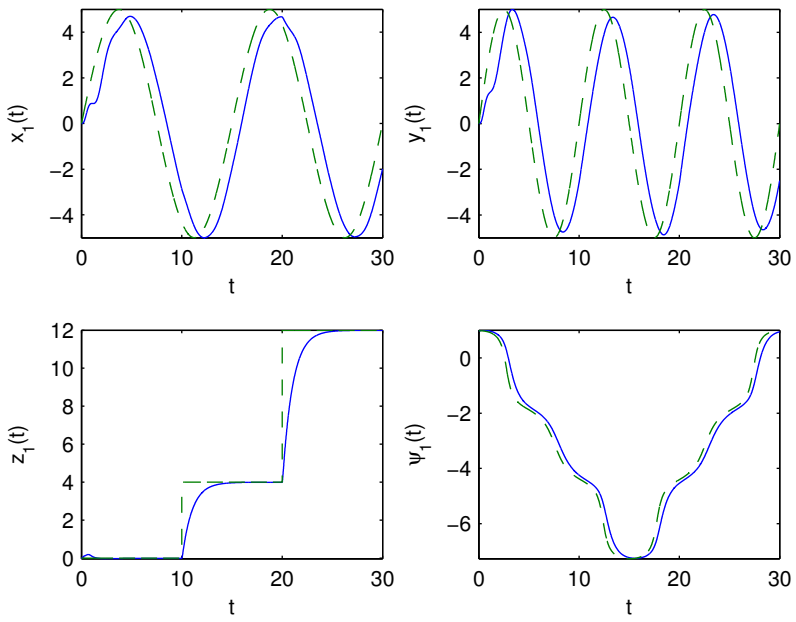


Figure 5.16: Tracking of Lissajous trajectory using gain scheduling under output feedback.

we augment the state model of the quadcopter

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}),$$
$$\mathbf{y} = \mathbf{C}\mathbf{x},$$

with the integrator

$$\dot{\sigma} = \mathbf{y} - \mathbf{r}$$

so that the augmented dynamics can be written

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}),$$
$$\dot{\sigma} = \mathbf{y} - \mathbf{r}.$$

Further, to compensate for the lack of having the full state available for feedback we introduce the observer

$$\dot{\hat{\mathbf{x}}} = \mathbf{f}(\hat{\mathbf{x}}, \mathbf{u}) + \mathbf{L}(\mathbf{y} - \mathbf{C}\hat{\mathbf{x}}).$$

Augmenting the observer with the state and integrator then gives

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}),$$
$$\dot{\sigma} = \mathbf{y} - \mathbf{r},$$
$$\dot{\hat{\mathbf{x}}} = \mathbf{f}(\hat{\mathbf{x}}, \mathbf{u}) + \mathbf{L}(\mathbf{y} - \mathbf{C}\hat{\mathbf{x}}).$$

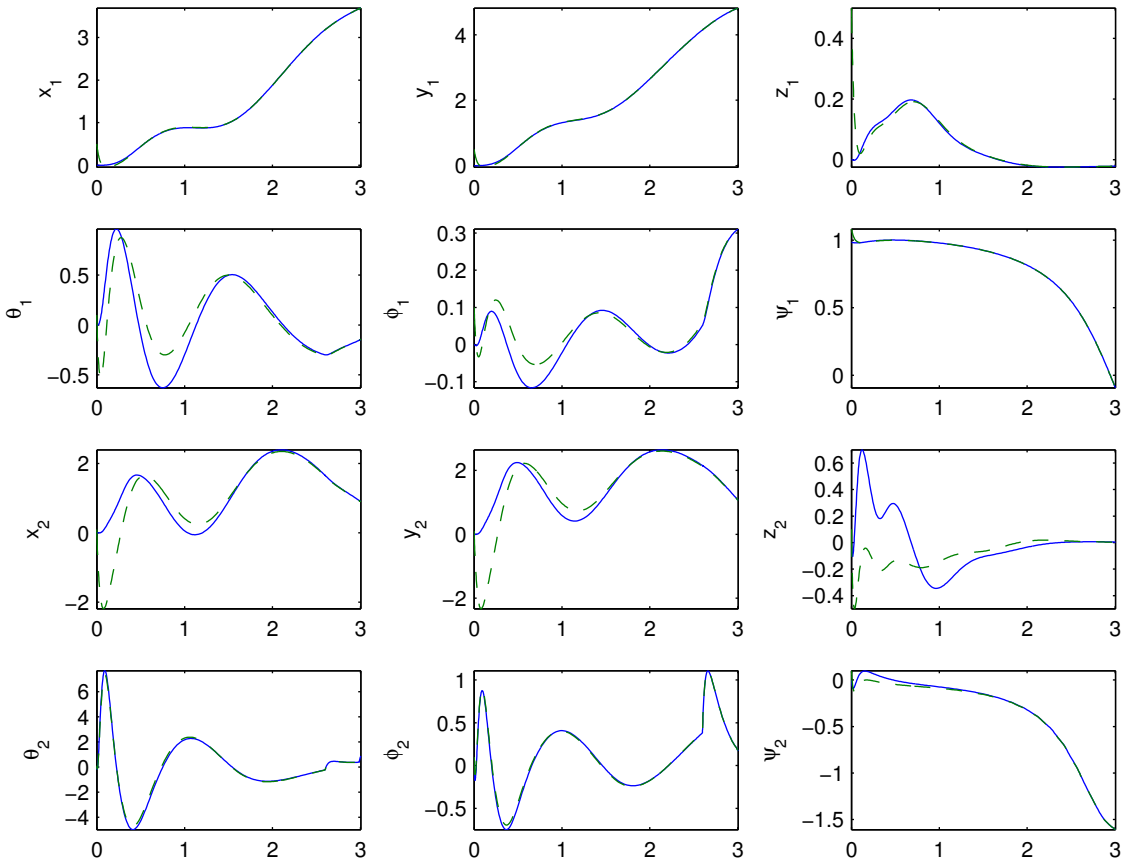


Figure 5.17: Comparing the observer $\hat{\mathbf{x}}$ (dashed green) with the actual state $\mathbf{x}$ (solid blue) from $t = 0$ to $t = 3$ for tracking of the Lissajous trajectory.

As before, for any frozen value of the reference signal $\mathbf{r}$ we wish to stabilize the above system at the operating point $(\mathbf{x}, \sigma, \hat{\mathbf{x}}) = (\mathbf{x}_{ss}, \sigma_{ss}, \mathbf{x}_{ss})$ where σ_{ss} produces the desired $\mathbf{u}_{ss}$. Linearizing the system about this steady state configuration gives

$$\begin{aligned} \dot{\mathbf{x}}_\delta &= \mathbf{A}\mathbf{x}_\delta + \mathbf{B}\mathbf{u}_\delta, \\ \dot{\sigma}_\delta &= \mathbf{C}\mathbf{x}_\delta, \\ \dot{\hat{\mathbf{x}}}_\delta &= \mathbf{A}\hat{\mathbf{x}}_\delta + \mathbf{B}\mathbf{u}_\delta + \mathbf{L}\mathbf{C}(\mathbf{x}_\delta - \hat{\mathbf{x}}_\delta). \end{aligned} \tag{5.6}$$

As before, we introduce

$$\xi = \begin{pmatrix} \mathbf{x} \\ \sigma \end{pmatrix} \tag{5.7}$$

so that (5.6) can be written

$$\begin{aligned} \dot{\xi}_\delta &= \mathcal{A}\xi_\delta + \mathcal{B}\mathbf{u}_\delta, \\ \dot{\hat{\mathbf{x}}}_\delta &= \mathbf{A}\hat{\mathbf{x}}_\delta + \mathbf{B}\mathbf{u}_\delta + \mathbf{L}\mathbf{C}(\mathbf{x}_\delta - \hat{\mathbf{x}}_\delta), \end{aligned}$$

where

$$\mathcal{A} = \begin{pmatrix} \mathbf{A} & 0 \\ \mathbf{C} & 0 \end{pmatrix}, \qquad \mathcal{B} = \begin{pmatrix} \mathbf{B} \\ 0 \end{pmatrix}.$$

In the case of state feedback, we would let $\mathbf{u}_\delta = -\mathbf{K}_1\mathbf{x}_\delta - \mathbf{K}_2\sigma_\delta$, or equivalently $\mathbf{u}_\delta = -\mathcal{K}\xi_\delta$ where $\mathcal{K} = (\mathbf{K}_1 \;\; \mathbf{K}_2)$. Since we are dealing with output feedback, we instead replace the appearance of the state $\mathbf{x}_\delta$ in the controller with our estimate $\hat{\mathbf{x}}_\delta$ and take

$$\mathbf{u}_\delta = -\mathbf{K}_1\hat{\mathbf{x}}_\delta - \mathbf{K}_2\sigma_\delta, \tag{5.8}$$

which results in the closed-loop system

$$\begin{aligned} \dot{\mathbf{x}}_\delta &= \mathbf{A}\mathbf{x}_\delta - \mathbf{B}\mathbf{K}_1\hat{\mathbf{x}}_\delta - \mathbf{B}\mathbf{K}_2\sigma_\delta, \\ \dot{\sigma}_\delta &= \mathbf{C}\mathbf{x}_\delta, \\ \dot{\hat{\mathbf{x}}}_\delta &= (\mathbf{A} - \mathbf{B}\mathbf{K}_1)\hat{\mathbf{x}}_\delta - \mathbf{B}\mathbf{K}_2\sigma_\delta + \mathbf{L}\mathbf{C}(\mathbf{x}_\delta - \hat{\mathbf{x}}_\delta). \end{aligned}$$

Introducing the error $\mathbf{e} = \mathbf{x}_\delta - \hat{\mathbf{x}}_\delta$ allows us to rewrite this as

$$\begin{pmatrix} \dot{\mathbf{x}}_\delta \\ \dot{\sigma}_\delta \\ \dot{\mathbf{e}} \end{pmatrix} = \begin{pmatrix} \mathbf{A} - \mathbf{B}\mathbf{K}_1 & -\mathbf{B}\mathbf{K}_2 & \mathbf{B}\mathbf{K}_1 \\ \mathbf{C} & 0 & 0 \\ 0 & 0 & \mathbf{A} - \mathbf{L}\mathbf{C} \end{pmatrix} \begin{pmatrix} \mathbf{x}_\delta \\ \sigma_\delta \\ \mathbf{e} \end{pmatrix},$$

or by using (5.7), we can write it as the upper triangular block matrix

$$\begin{pmatrix} \dot{\xi}_\delta \\ \dot{\mathbf{e}} \end{pmatrix} = \begin{pmatrix} \mathcal{A} - \mathcal{B}\mathcal{K} & \mathcal{B}\mathbf{K}_1 \\ 0 & \mathbf{A} - \mathbf{L}\mathbf{C} \end{pmatrix} \begin{pmatrix} \xi_\delta \\ \mathbf{e} \end{pmatrix}.$$

Now it is clear that by designing $\mathcal{K} = (\mathbf{K}_1 \;\; \mathbf{K}_2)$ and $\mathbf{L}$ such that the matrices $\mathcal{A} - \mathcal{B}\mathcal{K}$ and $\mathbf{A} - \mathbf{LC}$ are Hurwitz, we will achieve the desired result under the controller in (5.8). Note that, as done previously, we can take $\sigma_{ss} = -\mathbf{K}_2^{-1}\mathbf{u}_{ss}$ which results in

$$\mathbf{u} = -\mathbf{K}_1\hat{\mathbf{x}}_\delta - \mathbf{K}_2\sigma.$$

As such, with $\mathbf{x}_{ss} = \mathbf{C}^T\mathbf{r}$ in the case of reference tracking and $\mathbf{y} = \mathbf{Cx}$ for output feedback, we can represent the final expression for the observer-based output feedback gain-scheduled controller with integral action as

$$\begin{aligned}
\mathbf{u} &= -\mathbf{K}_1(\hat{\mathbf{x}} - \mathbf{C}^T\mathbf{r}) - \mathbf{K}_2\sigma, \\
\dot{\hat{\mathbf{x}}} &= \mathbf{f}(\hat{\mathbf{x}}, \mathbf{u}) + \mathbf{L}(\mathbf{y} - \mathbf{C}\hat{\mathbf{x}}), \\
\dot{\sigma} &= \mathbf{y} - \mathbf{r}.
\end{aligned}$$

The control law is represented in figure 5.18.

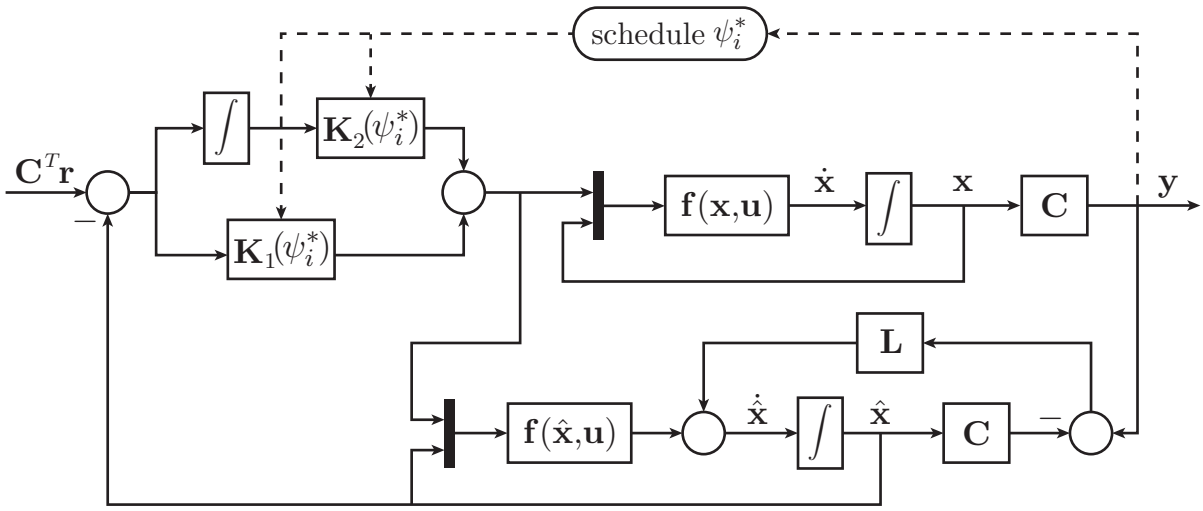


Figure 5.18: Block diagram of the gain-scheduled controller with integral action under output feedback.

We now implement this controller into the tracking of the helix and Lissajous trajectories. We expect to see the same behaviour observed in the previous instances when we implemented integral action and output feedback separately. Namely, the elimination of

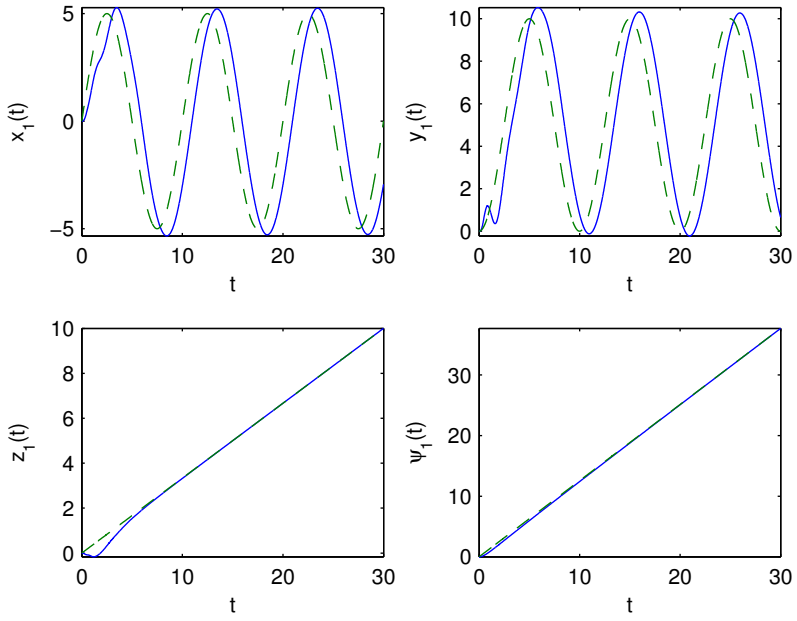


Figure 5.19: Tracking of helix trajectory using gain scheduling under output feedback with integral action.

steady state error when subjected to linear reference signals and improved performance as the observer converges to the state.

We first repeat the helix trajectory simulation. Figure 5.19 compares the output values to the reference signal. The results are very similar to the PI controller with state feedback, with the exception of the beginning of the trajectory. The integrator modeled in the controller successfully eliminates the steady state error that was seen in the reference signals for $z_1(t)$ and $\psi_1(t)$. The result of the observer converging to the actual state can be seen in figure 5.20.

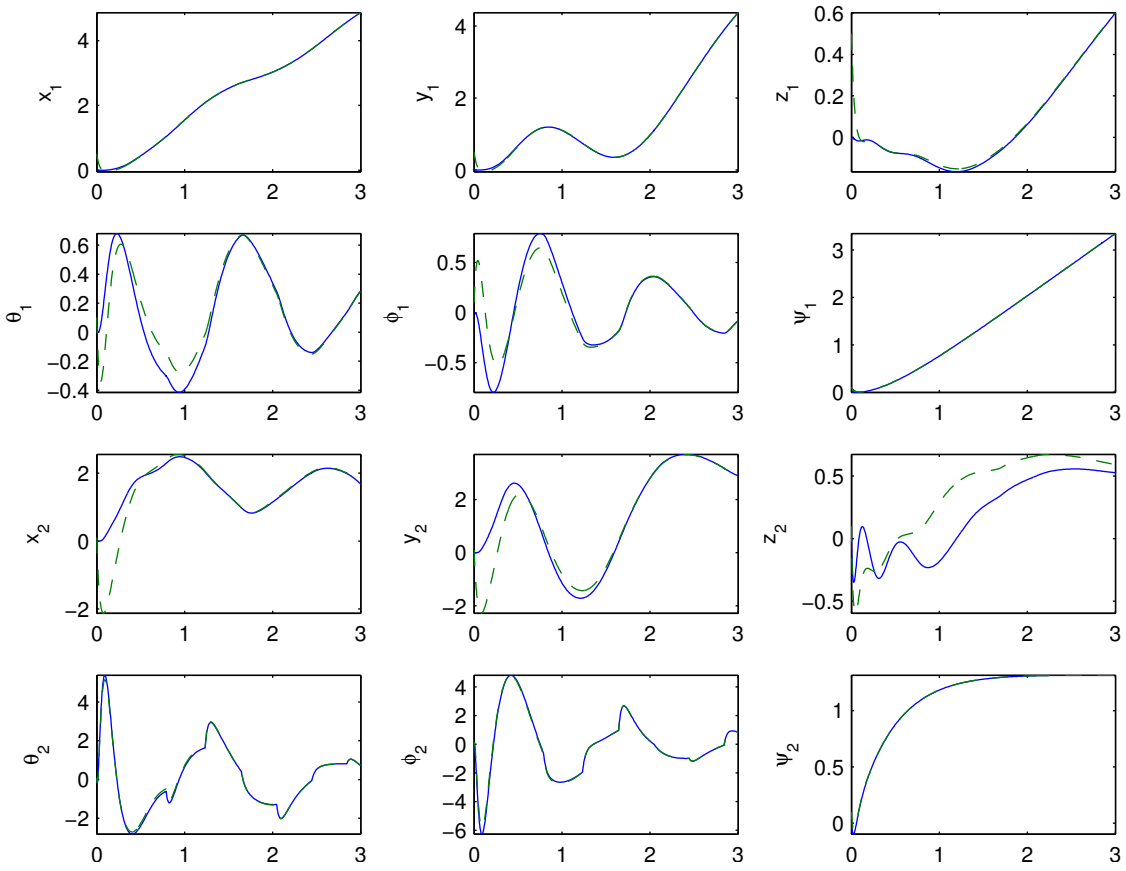


Figure 5.20: Comparing the observer $\hat{\mathbf{x}}$ (dashed green) with the actual state $\mathbf{x}$ (solid blue) from $t = 0$ to $t = 3$ for tracking of the helix trajectory with integral action.

Finally, we use the PI controller under output feedback to track the Lissajous trajectory, as can be seen in figure 5.21. Again, the behaviour we see here is a combination of the

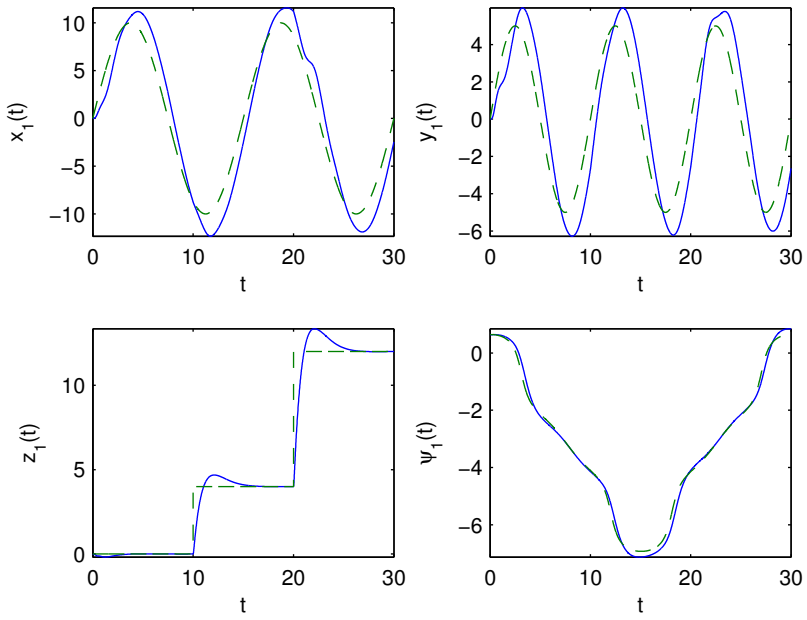


Figure 5.21: Tracking of Lissajous trajectory using gain scheduling under output feedback with integral action.

simulations done in sections 5.3 and 5.4 that employed integral action and output feedback separately. The increase in the weight for the integral component of the controller from $w_\sigma = 0.1$ to $w_\sigma = 1$ results in greater overshoot in the transient response but allows for quicker convergence to the steady state as can bee seen in the plot for $z(t)$. For sake of completeness, we compare the state with the observer at the beginning of the trajectory in figure 5.22.

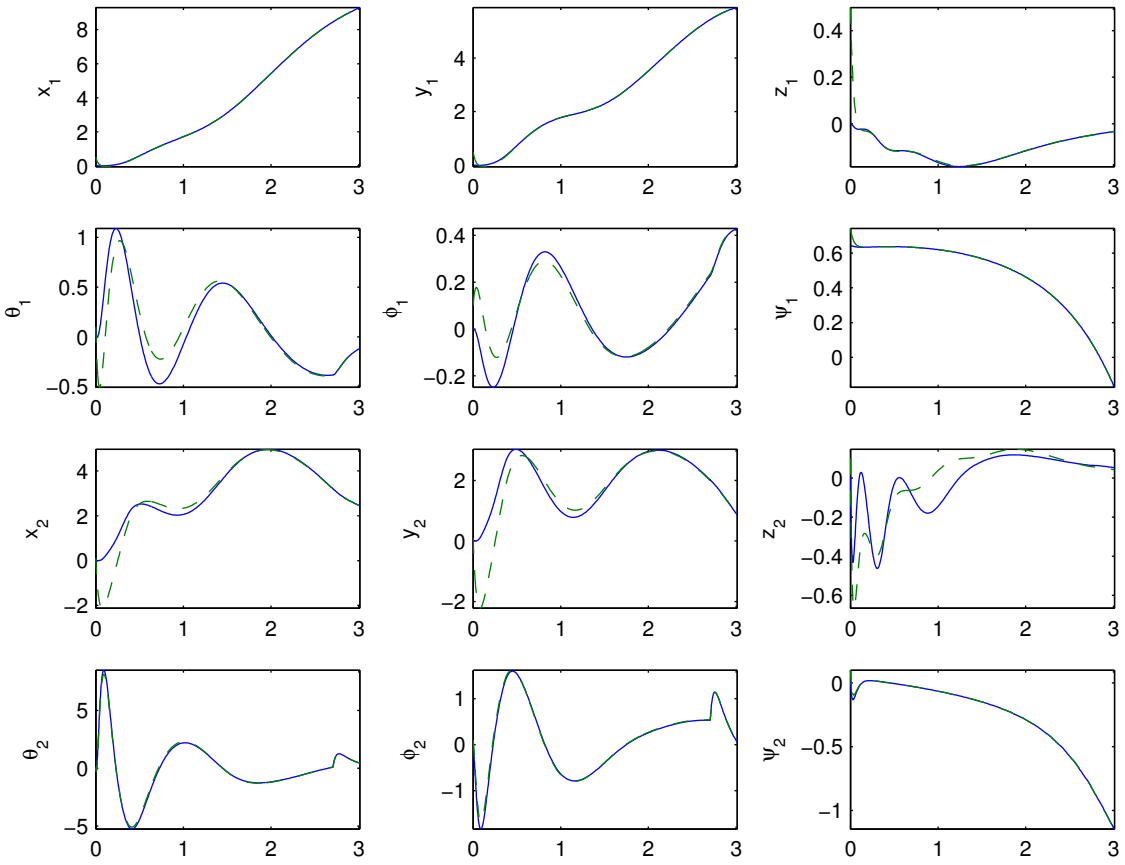


Figure 5.22: Comparing the observer $\hat{\mathbf{x}}$ (dashed green) with the actual state $\mathbf{x}$ (solid blue) from $t = 0$ to $t = 3$ for tracking of the Lissajous curve with integral action.

5.6 Considerations for Physical Implementation

We have seen that by switching discretely between different point designs, the quadcopter under the gain-scheduled controller is able to successfully track various reference signals. However, we have not yet considered whether or not such a controller would behave in physical applications. In the above simulations, we did not heavily restrict the effort that could be applied by the controller. Additionally, as a result of switching discretely between point designs, the controller is discontinuous. We will now take a closer look at these two factors and how they could affect physical implementations of the control law.

Let us restrict our attention to the use of state feedback without integral action. Recall that for each point design used in the control algorithm, we constructed the gain matrix $\mathbf{K}(\psi_i^*)$ such that the cost

$$J = \int_0^\infty \left(\mathbf{x}_\delta^T \mathbf{Q} \mathbf{x}_\delta + \mathbf{u}_\delta^T \mathbf{R} \mathbf{u}_\delta\right)\, dt \tag{5.9}$$

would be minimized under $\mathbf{u}_\delta = -\mathbf{K}(\psi_i^*)\mathbf{x}_\delta$. However, by using the weight matrix

$$\mathbf{R} = \text{diag}(w_{u1}, w_{u2}, w_{u3}, w_{u4})$$

with $w_{ui} = 0.001$ for each i, we put a very minimal restriction on the amount of effort that could be expended by the controller. By doing so, the controller is likely to produce very large signals that the actuator will not be able to realize. For example, figure 5.23 depicts the values of the control inputs u_1, u_2, u_3, u_4, corresponding to the thrust, pitch, roll and yaw moment inputs, respectively, as the gain-scheduled controller using state feedback without integral action tracks the Lissajous trajectory.

We can see that the amount of effort generated by the control inputs is extremely high, especially when subjected to the two step inputs at $t = 10$ and $t = 20$. Such high and rapid demand of effort from the controller could result in saturation of the actuator or even failure of the hardware.

Increasing the size of the matrix $\mathbf{R}$ in (5.9) allows us to reduce the amplitude of the signals generated by the controller. By using

$$\mathbf{R} = \text{diag}(1, 1, 1, 1),$$

we scale the previous instance of the weight matrix for the control by 1000. The control signals are reduced to those shown in figure 5.24. As we can see the amplitudes of the control signals are reduced considerably.

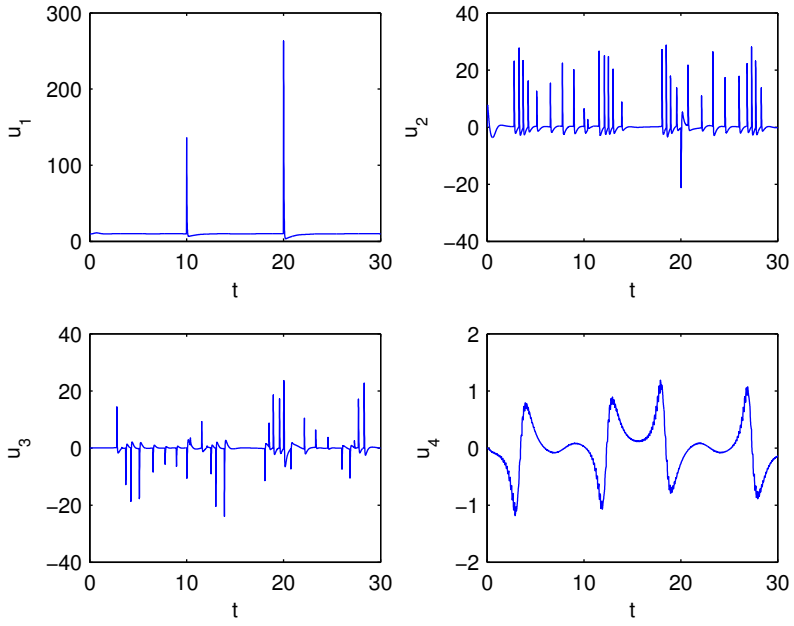


Figure 5.23: Signals generated from the controller during the tracking of the Lissajous trajectory, resulting from the use of $w_{ui} = 0.001$ in $\mathbf{R}$. Such large signals are unlikely to be realized in physical application.

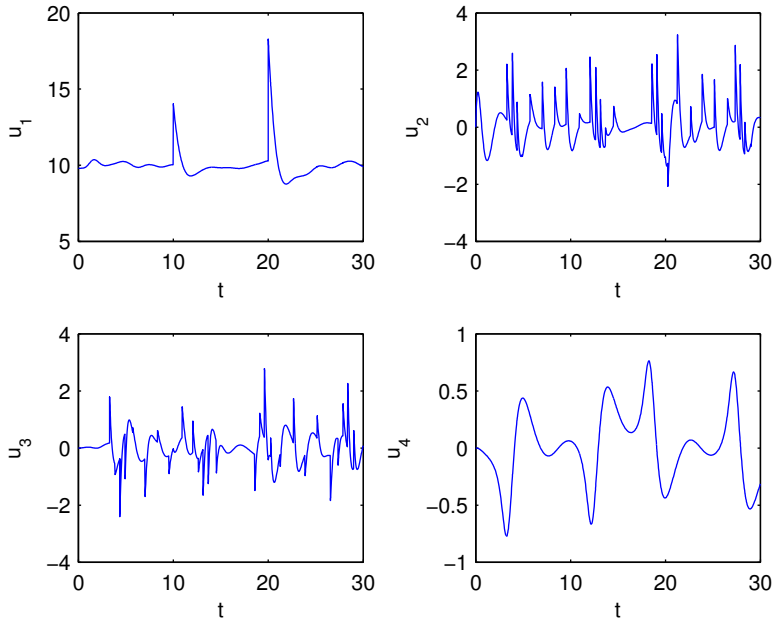


Figure 5.24: Signals generated from the controller during the tracking of the Lissajous trajectory, resulting from the use of $w_{ui} = 1$ in $\mathbf{R}$. These signals are much more likely to be realized in a physical actuator.

Of course, a reduction in the control effort does not come without a cost. By increasing $\mathbf{R}$ and leaving $\mathbf{Q}$ the same we are being less restrictive on how well the quadcopter tracks the reference. Figure 5.25 depicts the tracking of the state to the Lissajous reference as a result of the increase in $\mathbf{R}$. By comparing to figure 5.9, we can see that the quadcopter does not track the reference signals as well with the restriction applied to the control inputs.

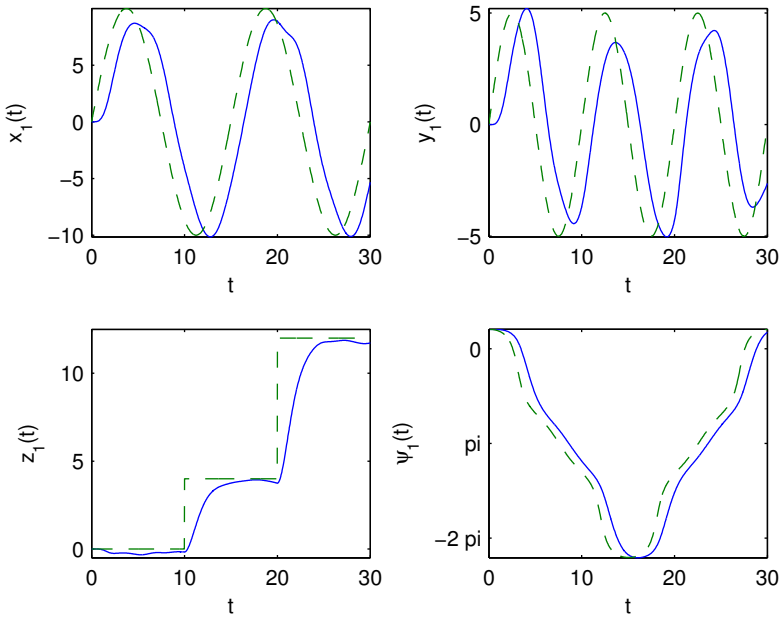


Figure 5.25: Tracking of the state to the reference signal for the Lissajous trajectory, using $w_{ui} = 1$. Comparing to figure 5.9, we see that the performance suffers as a result of restricting the effort expendable by the control inputs.

We may be able to improve the tracking while at the same time taking actuator saturation into account by increasing the matrix $\mathbf{R}$ so as to maintain satisfactory tracking of the reference signal, and further explicitly include restrictions on the control inputs,

$$|u_i(t)| \leq u_{i\max},$$

where each $u_{i\max}$ depends on the physical actuator used. As such, we can directly ensure that the actuator is able to handle the signals generated by $\mathbf{u}$. In this way, only reference trajectories that require inputs from the controller resulting in saturation of the actuator (such as the large step inputs seen in the Lissajous trajectory) will see a similar decrease in tracking performance as in figure 5.25.

We now address the effect of the discontinuity in the control law. By switching at intervals of $\pi/6$ in our simulations, we impose large fluctuations to the value of the u_i's at every switch. Essentially, the controller has to "reset" itself each time that the gain is rescheduled. To exploit this, we can repeat the plot of $u_3(t)$ shown in figure 5.24, but change the colour of the plot at every instance that the control switches, while at the same time leaving the discontinuities. The result of doing this is shown in figure 5.26.

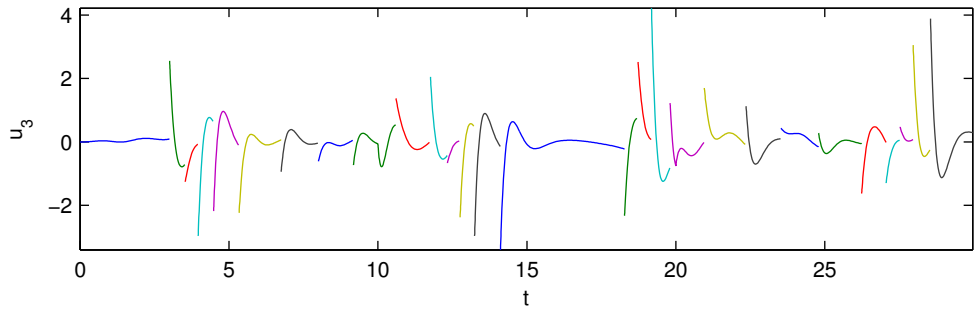


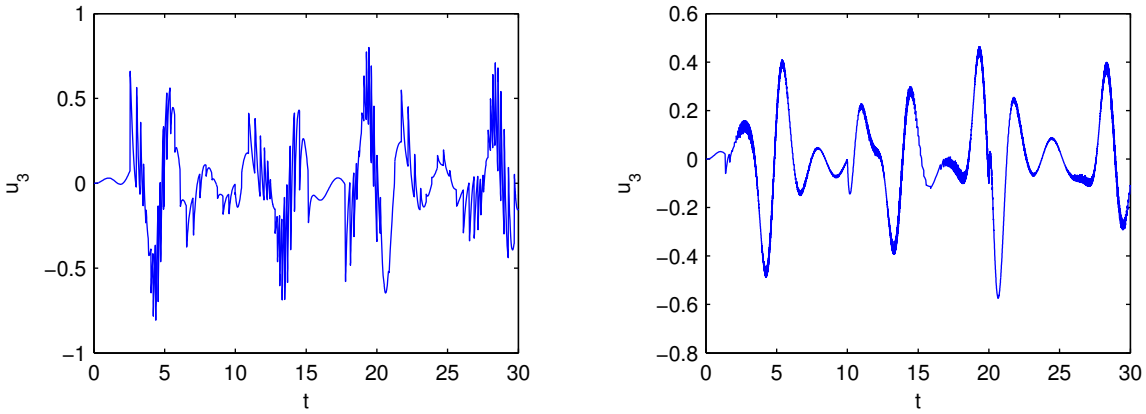
Figure 5.26: Plot of $u_3(t)$, using different colours to depict the scheduling of the gain. Large discontinuities appear in the controller as a result of using $\psi_{tol} = \pi/6$.

We can decrease the size of these discontinuities by making the switching more frequent. To do this, we decrease the size of ψ_{tol}. In figure 5.27 we see the output of $u_3(t)$ after repeating the Lissajous trajectory, with $\psi_{tol} = \pi/18$ and $\pi/180$, making the switching 3 times and 30 times, respectively, more frequent than before.

As the switching frequency increases, the size of the discontinuities decreases. In addition, this has the effect of reducing the amplitude of the control signal. We can visually see the signal "converge" as ψ_{tol} becomes closer to zero. In the limit as $\psi_{tol} \to 0$, the controller becomes continuous. However, in physical implementation we will always have a finite ψ_{tol}, which results in a phenomenon known as *chattering*. We depict the occurrence of chattering in figure 5.28.

In physical implementation, chattering is undesired. It can lead to high heat loss in electrical circuits as a result of the frequent switching of the controller, or can even cause the system to destabilize in the presence of unmodeled high frequency dynamics (if, for example, the frequency of the control switching matches that of the resonant frequencies of the unmodeled dynamics) [6].

Chattering often appears in control techniques that are inherently discontinuous, such as sliding mode control. In these controllers, the sgn$(\cdot)$ function is used to force the trajectory to converge to a surface or subset of the dynamics on which the system has desirable



(a) $\psi_{tol} = \pi/18$ (b) $\psi_{tol} = \pi/180$

Figure 5.27: Reducing the size of the discontinuities by increasing the frequency of the switching. The amplitude of the control signal also decreases.

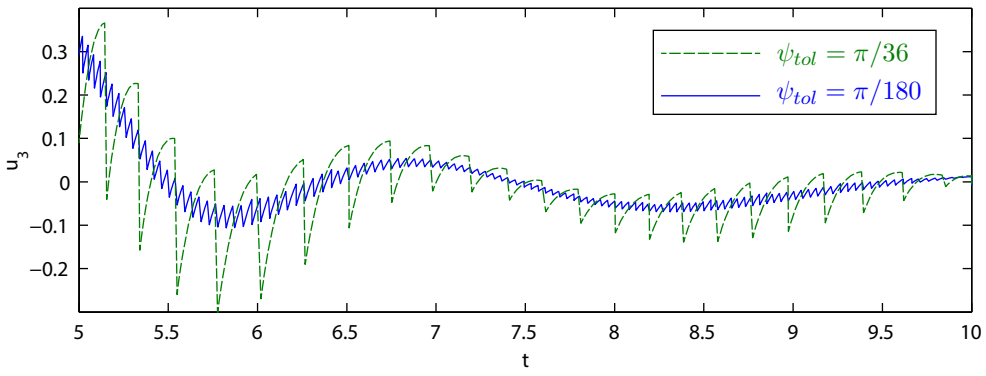


Figure 5.28: Presence of chattering as a result of increasing the switching frequency.

properties. The fact that $\mathrm{sgn}(\cdot)$ is discontinuous leads to chattering in these controllers. To prevent such undesirable behaviour, continuous approximations of $\mathrm{sgn}(\cdot)$ are often used (*e.g.*, [12]). In the case of gain scheduling, we can make the controller continuous by interpolating between the different point designs. Such methods are discussed in works such as [7, 10, 11].

Chapter 6

Conclusions

In this thesis, we used techniques from linear control theory to develop a gain-scheduled control law to apply to the quadcopter. By taking the yaw angle as the scheduling variable, we used the parameter-varying linearized dynamics of the quadcopter to develop the adaptive gain matrices in the control laws with the aid of a linear quadratic regular. We demonstrated that a conventional fixed gain control law was inadequate for tracking reference signals that involve significant variation in the yaw angle, and that the gain-scheduled controller successfully resolves this issue. To exemplify this, trajectory tracking simulations were conducted which involved both constant and non-constant variation in the yaw angle. Further, we introduced integral action into the control law which successfully eliminated the steady state error that we observed in the tracking of linear reference signals. Finally, we used an observer for state estimation where the output was taken to be the spatial position and yaw angle of the quadcopter. The observer was able to successfully converge to the state.

Although our control algorithm proved to be successful, there are many components in the design process that can be improved upon. Here we will discuss some improvements to the gain-scheduled controller that can be made in future work.

6.1 Establishing Region of Attraction

In the development of our control laws we have predominantly made use of the linearized model of the quadcopter. The downfall with this approach is that the linear model is only valid in a neighbourhood of the equilibrium point about which the linearization was taken.

Given a desired setpoint, we need to ensure that the initial condition of the state is within the region of attraction of the setpoint. From Lyapunov stability theory, we are guaranteed that such a region exists, however determining a suitable region is nontrivial. Nonetheless, techniques exist to estimate these regions of attraction. Some techniques are presented in [17, 18, 19, 20, 21]. As discussed, we may be able to use the fact that $V(\mathbf{x}) = \mathbf{x}^T\mathbf{P}\mathbf{x}$ is a Lyapunov function for the closed-loop system in a neighbourhood of the desired setpoint, where $\mathbf{P}$ is the unique positive definite solution to the Lyapunov equation

$$\mathbf{P}(\mathbf{A} - \mathbf{B}\mathbf{K}) + (\mathbf{A} - \mathbf{B}\mathbf{K})^T\mathbf{P} + \mathbf{Q} = 0,$$

with $\mathbf{Q}$ any positive definite symmetric matrix, to obtain a conservative estimate for the region of attraction.

6.2 Improvement of Gain Scheduling Method

Next we consider how the gains are scheduled in the control law. While the strategy used to schedule the controller on the yaw angle of the quadcopter was successful in our simulations, it can certainly be improved. Our strategy was to reschedule the controller on the current yaw angle of the quadcopter once it deviated a specified tolerance away from the value upon which it was previously scheduled. While this technique is sufficient for simulations, it is inefficient as a new controller must be derived on-line during each switch.

It would be much more efficient to first specify the tolerance ψ_{tol} and design the sequence of controllers before any simulations are done. For example, in most of our simulations we used $\psi_{tol} = \pi/6$. We can develop a controller as described above as

$$\mathbf{u}(\mathbf{x}) = \begin{cases} \mathbf{u}_1\left(\mathbf{x}; \dfrac{\pi}{12}\right), & 0 \leq \psi_1(t) < \dfrac{\pi}{6}; \\ \mathbf{u}_2\left(\mathbf{x}; \dfrac{\pi}{4}\right), & \dfrac{\pi}{6} \leq \psi_1(t) < \dfrac{\pi}{3}; \\ \quad \vdots & \\ \mathbf{u}_{12}\left(\mathbf{x}; \dfrac{23\pi}{12}\right), & \dfrac{11\pi}{6} \leq \psi_1(t) < 2\pi, \end{cases}$$

where $\mathbf{u}_i(\mathbf{x}; \psi_i^*)$ uses the linearization of the quadcopter dynamics about $\psi_i^* = \frac{\pi}{6}(i - \frac{1}{2})$, and $\psi_1(t)$ is taken mod 2π. In addition to not having to compute the control law on-line, this choice offers the additional improvement that the yaw angle of the quadcopter will

never be much more[1] than $\pi/12$ radians away from the point at which the dynamics were linearized about, as opposed to $\pi/6$ radians as was the case in the simulations. In general, for a suitable ψ_{tol} such that $2\pi/\psi_{tol} = n \in \mathbb{N}$ and $k \in \mathbb{N}, 1 \leq k \leq n$, we can use

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}_k \left(\mathbf{x}; \left(k - \frac{1}{2} \right) \psi_{tol} \right), \quad (k-1)\psi_{tol} \leq \psi_1(t) < k\,\psi_{tol} \tag{6.1}$$

so that we will maintain $|\psi_1(t) - \psi_k^*| \leq \psi_{tol}/2 + \epsilon$ for some small positive value ϵ[2]. Using such a controller, we can govern the switching by monitoring only the value of $\psi_1(t)$ and not its deviation away from the value of ψ_k^* upon which the linearization was taken.

Additionally, we can improve the controller by interpolating between each point design $\mathbf{u}_k$ given in (6.1). Numerous authors have developed different techniques for this interpolation process, many of which use linear interpolation. These include [22] where the poles, zeros and gains of the controller transfer functions are linearly interpolated, [23] where the solutions of Ricatti equations are linearly interpolated while employing the use of H_∞ controllers, and [24] where the state and observer gains are linearly interpolated. For additional techniques, we may wish to consult [7, 10] and references therein. This interpolation process will result in a continuous gain-scheduled controller and should eliminate the chattering phenomenon discussed in section 5.6.

6.3 Shaping of Dynamic Response

In the development of our control law, we saw that the eigenvalues in the closed-loop system could be, at least theoretically, arbitrarily placed. This feature enables us to place the eigenvalues in order to alter the performance in achieving different design objectives, whether it be to achieve a desired rate of convergence, track a reference signal, reject disturbances to the system or be robust to noise. In the designing of the controller we obtained the gain matrices by using a linear quadratic regular as discussed in section 2.4. However, not much consideration was given to the $\mathbf{Q}$ and $\mathbf{R}$ weight matrices that determine how the system will be affected by the controller.

In future work, we may want to look into this further and see how the weight matrices should be varied in order to best achieve various design objectives and shape the dynamic response. It would be beneficial to first compare the response to simpler reference inputs

[1] We say "much more" because by the time the yaw angle varies enough to trigger a switch in the control law, it will be in the next range of $\psi_1(t)$.

[2] See the above footnote.

such as step inputs and analyze various performance indicators such as rise time, settling time, overshoot, etc.

6.4 Actuator Saturation

We saw in section 4.3 that the linearized dynamics of the quadcopter were completely controllable and observable, implying that we could, in principle, arbitrarily place the poles of the closed-loop system to achieve the desired system response. Unfortunately, what we can accomplish in principle may not coincide with what we can accomplish in practice. The fact of the matter is that the physical actuators (or the power supply) may not be able to generate the signals output from the controller resulting from our choice of pole placement. It may be the case that the actuator saturates at the largest signal that it can deliver, which could in turn destabilize the system.

Various papers have considered the issue of actuator saturation, such as [25, 26] in which saturation is avoided, and [27] where controllers are developed that incorporate saturation. To prevent this problem in the controller development for the quadcopter model, we may wish to incorporate actuator saturation into the construction of the control law, by means of increasing the weight matrix $\mathbf{R}$ in the linear quadratic regulator as we have done in section 5.6, or through other techniques. Alternatively, we could avoid saturation by explicitly planning trajectories in which saturation will not occur. We have also seen that using a continuous gain-scheduled controller will reduce the amplitude of the control signal, aiding to prevent saturation from occurring.

6.5 Improved Observer Model

When we developed our gain-scheduled controllers in conjunction with output feedback, we made use of the observer

$$\dot{\hat{\mathbf{x}}} = \mathbf{f}(\hat{\mathbf{x}}, \mathbf{u}) + \mathbf{L}(\mathbf{y} - \mathbf{C}\hat{\mathbf{x}}),$$

in order to estimate the state. By varying the matrix $\mathbf{L}$, we had the freedom to arbitrarily place the poles of the error dynamics

$$\dot{\mathbf{e}} = (\mathbf{A} - \mathbf{LC})\mathbf{e}$$

to achieve the desired rate of convergence of the observer to the state.

Unfortunately, increasing the rate of convergence does not come without consequence in practical application. Increasing the rate of convergence by placing the closed-loop poles farther left in the open left-hand plane results in greater error in the transient response. As mentioned in section 5.4, this *peaking phenomenon* can be unsafe for the physical implementation of the system as it has the effect of amplifying noise in the system. As such, it would be unwise to make the observer gain matrix too large. This trade-off between noise attenuation and dynamic response can be done in an "optimal" way by using a Kalman filter, which incorporates a model for the noise in the system in the construction of the *Kalman gain* (or in our case the Kalman-Bucy filter for continuous-time systems [28]).

APPENDICES

Appendix A

Simulation Parameters

Symbol	Value	SI Units	Description
g	9.8	m/s^2	Gravitational constant
m	2	kg	Mass of quadcopter
ℓ	0.2	m	Half of adjacent rotor distance
I_1	1.25	$N\ s^2/rad$	Moment of inertia
I_2	1.25	$N\ s^2/rad$	Moment of inertia
I_3	2.50	$N\ s^2/rad$	Moment of inertia
k_1	0.010	N s/m	Drag coefficient
k_2	0.010	N s/m	Drag coefficient
k_3	0.010	N s/m	Drag coefficient
k_4	0.012	N s/m	Drag coefficient
k_5	0.012	N s/m	Drag coefficient
k_6	0.012	N s/m	Drag coefficient

Appendix B

Proofs

B.1 Controllability of $(\mathcal{A}, \mathcal{B})$

Let $\mathbf{A} \in \mathbb{R}^{n\times n}, \mathbf{B} \in \mathbb{R}^{n\times p}, \mathbf{C} \in \mathbb{R}^{p\times n}$, with $\mathcal{A}$ and $\mathcal{B}$ given by (2.16). Suppose that $(\mathbf{A}, \mathbf{B})$ is a controllable pair, and the matrix $\begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & 0 \end{pmatrix}$ has rank $n + p$. Then the pair $(\mathcal{A}, \mathcal{B})$ is controllable.

Proof. By way of contradiction, suppose that the pair $(\mathcal{A}, \mathcal{B})$ is not controllable. Then by the Hautus criterion for controllability, $\exists \lambda \in \mathbb{C}$ such that

$$\text{rank } \begin{pmatrix} \mathcal{A} - \lambda \mathbf{I}_{n+p} & \mathcal{B} \end{pmatrix} < n + p. \tag{B.1}$$

Since the matrix in (B.1) is not full rank, there is a $\mathbf{v} = (\mathbf{v}_1 \;\; \mathbf{v}_2)$ with $\mathbf{v}_1 \in \mathbb{C}^{1\times n}, \mathbf{v}_2 \in \mathbb{C}^{1\times p}$, and $\mathbf{v}_1, \mathbf{v}_2$ not both 0, such that

$$\begin{pmatrix} \mathbf{v}_1 & \mathbf{v}_2 \end{pmatrix} \begin{pmatrix} \mathbf{A} - \lambda \mathbf{I}_n & 0 & \mathbf{B} \\ \mathbf{C} & -\lambda \mathbf{I}_p & 0 \end{pmatrix} = 0,$$

giving the three equations

$$\mathbf{v}_1(\mathbf{A} - \lambda \mathbf{I}_n) + \mathbf{v}_2\mathbf{C} = 0, \tag{B.2}$$

$$\lambda \mathbf{v}_2 = 0, \tag{B.3}$$

$$\mathbf{v}_1\mathbf{B} = 0. \tag{B.4}$$

Case 1. $\lambda \neq 0$. Then by (B.3), $\mathbf{v}_2 = 0$. Taking this with $\mathbf{v} \neq 0$ we must now have that $\mathbf{v}_1 \neq 0$. Conditions (B.2) and (B.4) with $\mathbf{v}_2 = 0$ imply that

$$\mathbf{v}_1 \begin{pmatrix} \mathbf{A} - \lambda \mathbf{I}_n & \mathbf{B} \end{pmatrix} = 0.$$

Since $\mathbf{v}_1 \neq 0$ we then have that

$$\text{rank} \begin{pmatrix} \mathbf{A} - \lambda \mathbf{I}_n & \mathbf{B} \end{pmatrix} < n,$$

contradicting the first assumption that $(\mathbf{A}, \mathbf{B})$ is controllable.

Case 2. $\lambda = 0$. Then (B.1) implies

$$\text{rank} \begin{pmatrix} \mathbf{A} & 0 & \mathbf{B} \\ \mathbf{C} & 0 & 0 \end{pmatrix} < n + p,$$

or equivalently

$$\text{rank} \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & 0 \end{pmatrix} < n + p,$$

thereby contradicting the second assumption.

Hence we must have that $(\mathcal{A}, \mathcal{B})$ is controllable.

□

B.2 Invertibility of $\mathbf{K}_2$

Let $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathcal{A}, \mathcal{B}$ be as above, and $\mathcal{K} = (\mathbf{K}_1 \;\; \mathbf{K}_2)$ with $\mathbf{K}_1 \in \mathbb{R}^{p \times n}, \mathbf{K}_2 \in \mathbb{R}^{p \times p}$. Suppose that $\mathcal{K}$ is chosen so that the matrix $\mathcal{A} - \mathcal{B}\mathcal{K}$ is Hurwitz. Then $\mathbf{K}_2$ is nonsingular.

Proof. We can write $\mathcal{A} - \mathcal{B}\mathcal{K}$ as

$$\begin{pmatrix} \mathbf{A} - \mathbf{B}\mathbf{K}_1 & -\mathbf{B}\mathbf{K}_2 \\ \mathbf{C} & 0 \end{pmatrix}. \tag{B.5}$$

By way of contradiction, suppose that $\mathbf{K}_2$ is singular. We then have that the rank of $\mathbf{K}_2$ is less than p. In particular, the rank of $\mathbf{B}\mathbf{K}_2$ is less than p, which would result in the matrix in (B.5) having rank less than $n + p$, contradicting the fact that it is Hurwitz.

Hence we must have that $\mathbf{K}_2$ is nonsingular.

□

References

[1] D. Luenberger, "Observing the state of a linear system," *IEEE Transactions on Military Electronics*, vol. 8, pp. 74–80, April 1964.

[2] B. Friedland, *Control system design: an introduction to state-space methods*. Courier Corporation, 2012.

[3] R. Kalman, "On the general theory of control systems," *IRE Transactions on Automatic Control*, vol. 4, no. 3, pp. 110–110, 1959.

[4] D. N. Burghes and A. Graham, *Introduction to control theory, including optimal control*. Horwood, 1980.

[5] J. Kautsky, N. K. Nichols, and P. Van Dooren, "Robust pole assignment in linear state feedback," *International Journal of Control*, vol. 41, no. 5, pp. 1129–1155, 1985.

[6] H. K. Khalil, *Nonlinear systems*. Prentice Hall, 3rd ed., 2001.

[7] W. J. Rugh and J. S. Shamma, "Research on gain scheduling," *Automatica*, vol. 36, no. 10, pp. 1401–1425, 2000.

[8] W. J. Rugh, "Analytical framework for gain scheduling," in *American Control Conference*, pp. 1688–1694, 1990.

[9] D. A. Lawrence and W. J. Rugh, "Gain scheduling dynamic linear controllers for a nonlinear plant," *Automatica*, vol. 31, no. 3, pp. 381–390, 1995.

[10] D. J. Stilwell and W. J. Rugh, "Interpolation methods for gain scheduling," in *Proceedings of the 37th IEEE Conference on Decision and Control*, vol. 3, pp. 3003–3008, 1998.

[11] D. J. Stilwell and W. J. Rugh, "Interpolation of observer state feedback controllers for gain scheduling," *IEEE Transactions on Automatic Control*, vol. 44, no. 6, pp. 1225–1229, 1999.

[12] R. Xu and Ü. Özgüner, "Sliding mode control of a quadrotor helicopter," in *45th IEEE Conference on Decision and Control*, pp. 4957–4962, Dec 2006.

[13] A. Chamseddine, Y. Zhang, C. Rabbath, C. Join, and D. Theilliol, "Flatness-based trajectory planning/replanning for a quadrotor unmanned aerial vehicle," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, pp. 2832–2848, October 2012.

[14] E. Altug, J. Ostrowski, and R. Mahony, "Control of a quadrotor helicopter using visual feedback," in *IEEE International Conference on Robotics and Automation*, vol. 1, pp. 72–77, 2002.

[15] J. B. Marion, *Classical dynamics of particles and systems.* Academic Press, 2nd ed., 2013.

[16] P. D. Christofides and N. H. El-Farra, *Control of Nonlinear and Hybrid Process Systems.* Springer Berlin Heidelberg, 2005.

[17] A. Papachristodoulou and S. Prajna, "On the construction of Lyapunov functions using the sum of squares decomposition," in *Proceedings of the 41st IEEE Conference on Decision and Control*, vol. 3, pp. 3482–3487, Dec 2002.

[18] P. A. Parrilo and S. Lall, "Semidefinite programming relaxations and algebraic optimization in control," *European Journal of Control*, vol. 9, no. 2, pp. 307–321, 2003.

[19] S. Ratschan and Z. She, "Providing a basin of attraction to a target region of polynomial systems by computation of Lyapunov-like functions," *SIAM Journal on Control and Optimization*, vol. 48, no. 7, pp. 4377–4394, 2010.

[20] P. Giesl, "Construction of a local and global Lyapunov function using radial basis functions," *IMA Journal of Applied Mathematics*, 2008.

[21] S. F. Hafstein, "A constructive converse Lyapunov theorem on exponential stability," *Discrete and Continuous Dynamical Systems*, vol. 10, no. 3, pp. 657–678, 2004.

[22] R. Nichols, R. T. Reichert, and W. J. Rugh, "Gain scheduling for H_∞ controllers: A flight control example," *IEEE Transactions on Control Systems Technology*, vol. 1, no. 2, pp. 69–79, 1993.

[23] R. T. Reichert, "Dynamic scheduling of modern-robust-control autopilot designs for missiles," *IEEE Control Systems*, vol. 12, no. 5, pp. 35–42, 1992.

[24] R. Hyde and K. Glover, "The application of scheduled H_∞ controllers to a VSTOL aircraft," *IEEE Transactions on Automatic Control*, vol. 38, pp. 1021–1039, Jul 1993.

[25] D. Henrion, S. Tarbouriech, and V. Kučera, "Control of linear systems subject to input constraints: a polynomial approach," *Automatica*, vol. 37, no. 4, pp. 597–604, 2001.

[26] E. B. Castelan and J. E. Cury, "A reduced-order framework applied to linear systems with constrained controls," *IEEE Transactions on Automatic Control*, vol. 41, no. 2, pp. 249–255, 1996.

[27] S. Tarbouriech, G. Garcia, and J. G. da Silva, "Robust stability of uncertain polytopic linear time-delay systems with saturating inputs: an LMI approach," *Computers & Electrical Engineering*, vol. 28, no. 3, pp. 157–169, 2002.

[28] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," *Journal of Fluids Engineering*, vol. 83, no. 1, pp. 95–108, 1961.