Path Following for Mechanical Systems Applied to Robotic Manipulators

by

Andre Hladio

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2010

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

Many applications in robotics require faithfully following a prescribed path. Tracking controllers may not be appropriate for such a task, as there is no guarantee that the robot will stay on the path. The objective of this thesis is to develop a control design method which makes the "output" of a robot get to, and move along the prescribed path without leaving the path. We consider the class of mechanical systems, which encompasses robotics.

Various techniques exist for designing path following controllers. We base our approach on a technique called "transverse feedback linearization". Using this technique, if feasible, we decompose the dynamics of a mechanical system into a transversal subsystem and a tangential subsystem using a coordinate and feedback transformation. The transversal subsystem is linear, time-invariant and decoupled from the tangential subsystem. Stabilizing the origin of the transversal subsystem is equivalent to stabilizing a set corresponding to the output of the mechanical system being on the desired path, thereby partly achieving the control objective. Given a mechanical system and a path, we provide conditions under which this is possible.

The tangential subsystem describes all of the motions of the mechanical system, when the output is on the path. Some tangential dynamics may move the output along the path, and thereby meet the design objective. In order to move the output of the mechanical system along the path, we further decompose the tangential subsystem into a subsystem which moves the output along the path, and a subsystem which does not, if feasible, using partial feedback linearization. The subsystem which governs output motions along the path is linear, time-invariant and decoupled. The remaining tangential dynamics have no special structure. We provide conditions under which such a decomposition of the tangential dynamics is possible.

We show that a five-bar robotic manipulator has dynamics which may be transversely feedback linearized, and the tangential dynamics may be partially linearized. Given a circular path, we experimentally implement our path following design, and observe that our control objective is indeed met. Inherent advantages of path following over trajectory tracking are illustrated.

Standard feedback linearization of a five-bar robotic manipulator with a flexible link has been shown to fail. We show that this system is transversely feedback linearizable, and its tangential dynamics may be partially linearized, under mild restrictions. Simulations illustrate path following applied to this complex system.

Acknowledgements

My co-supervisors David Wang and Christopher Nielsen: thank you for giving me the opportunity work under your supervision, I could not have asked for a better arrangement. I am particularly grateful for your support through some unusual circumstances. I have learned much by your help and guidance, and my Master's work is something I will always be proud of.

I am grateful for being surrounded by people who have positively influenced me on a personal and intellectual level while writing this thesis. I am especially thankful to Kevin Walker, Mike Tribou, John Daly, John Tang, Roop Peterson, Ravi Ravichandran, and Ruth-Anne Vanderwater. I would also like to thank my readers and former teachers Daniel Davison and Kaan Erkorkmaz.

I appreciate the love, patience and support of my friends, family, and in particular, of my fiancée Manisha, in completing this thesis.

Dedication

To my grandparents: Angela, Mykola, William and Orysia.

Contents

List of Tables **ix**

List of Figures **xi**

1 Introduction **1**

1.1 Motivation 2

1.2 Literature Review 4

1.3 Organization and Contribution 8

2 Math Preliminaries **9**

2.1 Notation 9

2.2 Elementary Theory 10

2.2.1 Properties of Maps 10

2.2.2 Differential Geometry Basics 11

2.2.3 Matrix Properties 11

2.2.4 Vector Fields 12

2.2.5 Control Systems 13

2.2.6 Input-Output Feedback Linearization 14

2.2.7 Full-state Linearization 18

3 Path Following for Mechanical Control Systems **19**

3.1 Summary of Path Following via Transverse Feedback Linearization 20

3.2 Mechanical Control Systems 28

3.2.1 Modeling 28
3.3 Path Following for Mechanical Systems 30
3.3.1 Illustrative Examples 36
3.3.2 Partial Feedback Linearization of Tangential Dynamics 40
3.3.3 Path Parametrization and Tubular Neighbourhoods 43
3.3.4 Partial Feedback Linearization of Tangential Subsystem 44

4 Application to a Planar Five-bar Manipulator 54
4.1 Dynamic Model 54
4.2 Manipulator Path Following 57
4.2.1 Transversal and Tangential Control Design 62
4.2.2 Observer Design 62
4.3 Experimental Study 63
4.3.1 Experimental Setup 63
4.3.2 Parameter Identification 64
4.3.3 Gain Selection 65
4.3.4 Simulation Results 65
4.3.5 Experimental Results 66
4.3.6 Discussion 70

5 Application to a Five-bar Manipulator with a Flexible Link 72
5.1 Dynamic Model 73
5.1.1 Modeling Flexible Beam 73
5.1.2 Dynamic Model of Five-bar Manipulator with a Flexible Link . . . 75
5.2 Background and Motivation 84
5.3 Flexible Manipulator Path Following 86
5.4 Simulation Results 89
5.5 Discussion 103

6 Conclusions and Future Work 105

APPENDICES **107**

A Positive Definite Matrices are Invertible **108**

Bibliography **114**

List of Tables

4.1 Five-bar manipulator model parameters 65

5.1 Simulation model parameters 92
5.2 Flexible beam parameters 92
5.3 Controller gains for flexible manipulator simulation 96

List of Figures

1.1 A tracking control system 2
1.2 Path following 3
1.3 Organization and contribution 8

3.1 A simple mechanical control system 22
3.2 Relationship of compositions 34
3.3 A simple mechanical control system revisited 37
3.4 An illustrative underactuated mechanical system 38
3.5 Another illustrative underactuated mechanical system 40
3.6 Kinematically redundant and underactuated systems 42
3.7 Illustration of projection 45
3.8 Puck with position and rotation on a flat surface 51
3.9 A 1-dimensional non-closed curve 53

4.1 Two degree-of-freedom five-bar linkage robot 55
4.2 Manipulator with path 58
4.3 Illustration of projection for planar five-bar 60
4.4 Block diagram of control system 63
4.5 Photographs of five-bar robot 64
4.6 Simulation of path following for planar five-bar 66
4.7 End-point trajectory in output space 67
4.8 Transformed states and control effort for end-point position control 68
4.9 Transformed states and control effort for end-point velocity profile tracking 69

4.10 Path following experiment with disturbance 70

5.1 Illustration of flexible beam deflection 74
5.2 Illustration of the term $x_\ell(a, q)$ 77
5.3 A five-bar manipulator with horizontal flexibility in the last link 78
5.4 Model of link flexibility 84
5.5 Illustration of the reflected tip position 85
5.6 Flexible five-bar path 90
5.7 Unacceptable five-bar joint configurations 94
5.8 Unacceptable five-bar configurations in output space 95
5.9 Flexible tip in output space 97
5.10 Transversal and tangential states for simulation 1 97
5.11 Control effort and deflection weights for simulation 1 98
5.12 Flexible tip in output space - tracking speed profile 99
5.13 Transversal and tangential states for simulation 2 99
5.14 Control effort and deflection weights for simulation 2 100
5.15 Flexible tip in output space with disturbance 101
5.16 Transversal and tangential states for simulation 3 102
5.17 Control effort and deflection weights for simulation 3 102

Chapter 1

Introduction

Suppose that you're driving to the corner store because you ran out of milk, and all of a sudden a child runs out onto the street. Naturally, you slam on the breaks and wait for the child to get off the road (hopefully realizing their mistake) before proceeding. Now when you proceed, you drive at a reasonable pace, likely similar to your pace before the child ran onto the road. It would be quite odd to want to "catch up" to where you would have been had the child not momentarily stopped you, especially since getting milk at the corner store is not particularly time-sensitive. Although eventually getting to the corner store is important, your first priority is not to hit any kids, parked cars, etc.

This toy example loosely illustrates a concept called "path following". Path following, as the name implies, means we are primarily concerned with following the path, which corresponds to staying on the road in our example. In a sense, getting to our destination, the corner store, is a secondary concern. Since it is secondary (at least relative to the staying on the road), it is not appropriate to strictly enforce a "time schedule" for the trip. Distinguishing between the primary concern of staying on the path, and the secondary concern of motion along the path, the essence of path following is to follow the path "at all costs", while not demanding a strict "time schedule" along the path.

To further help define the idea of path following, we contrast it with a fundamentally different idea, which for now we loosely refer to as "trajectory tracking". To illustrate trajectory tracking, let us tweak our toy example. Suppose your destination is not the corner store, but rather an important meeting with your punctual boss. Once the child has cleared the road, you might speed up to make up lost time. In fact, you might even cut through a gas station to avoid a red light. In this new example, staying on your path is important, but it is equally important to stay on schedule to avoid a late arrival. As a result, cutting across a parking lot to save time is, in some sense, appropriate.

1.1 Motivation

We have illustrated the main idea of path following, and have linked this idea to intuitive behaviour of the driver of an automobile. We wish to apply this idea to control systems. Path following control design problems can be formulated for a large class of systems and can be solved using various techniques. In this thesis, we seek to achieve path following for mechanical control systems. The class of mechanical control systems is incredibly broad and diverse, largely due to the fact that humans have always used mechanical systems to perform tasks they were unwilling or unable to perform themselves.

Trajectory tracking

To provide a point of contrast with path following, we introduce the notion of trajectory tracking. Standard training in control theory includes the tracking problem (often introduced in the frequency domain framework). This is a quite natural place to start; it addresses the problem, "how do we get the output of this system to follow the input?". In trajectory tracking schemes, we have a reference signal, $r(t)$, parameterized by time, which the output, $y(t)$, ought to follow closely, subject to design trade-offs, performance limitations, etc. Applied to a robotic system, we have the block diagram of a standard tracking scheme in Figure 1.1.

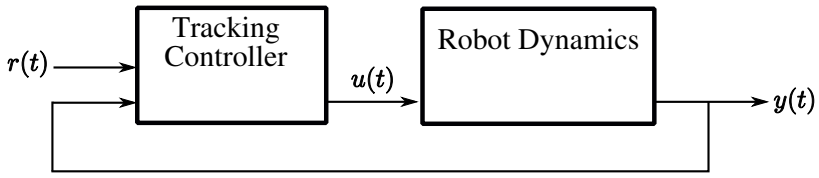


Figure 1.1: A block diagram showing a typical tracking control system

Tracking controllers can be designed using various control techniques, such as feedback linearization, sliding mode, adaptive, and robust control. Even considering the wealth of literature and experience using tracking controllers for mechanical systems, there are many situations in which tracking simply isn't appropriate.

Path following control

Suppose we have a path $\gamma \in \mathbb{R}^p$, as shown in Figure 1.2. In path following control, the objective is use feedback control to

1. make the output, y, of the system go to the path γ.

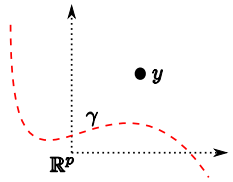


Figure 1.2: The output of a control system with a given path

2. ensure that if $y \in \gamma$ is properly initialized, it will remain on the path γ for all time, a property called "output invariance".

3. achieve desired motion along the path (if possible). Usual requirements of the motion along the path include speed/position and direction of traversal, along with boundedness of the internal dynamics.

Path following via set stabilization

There are various ways to implement a path following controller, many of which will be subsequently discussed in Section 1.2. We treat path following control as an instance of set stabilization. Using set stabilization, we may achieve invariance of the output with respect to the path. This means that once "properly" initialized on the path, the output, y, will remain on the path, γ, for all time. The precise definition of output invariance is found in Definition 3.1.2. In other words, we are stabilizing a family of trajectories, whose outputs correspond to being "on the path" (where a trajectory tracking scheme would only stabilize one particular trajectory). This collection of trajectories is a manifold in the state-space, which we call the *path following manifold*, denoted by $\Gamma^\star$. The set stabilization problem is presented in [38].

Applications

There are a plethora of applications for which path following via set stabilization is the appropriate approach for controlling mechanical systems. Consider, for example, a robotic manipulator which has the task of painting shapes on a canvas. Applying a trajectory tracking approach may cause the robot to deviate from the intended shape at difficult locations. However, a path following approach will guarantee the shape is faithfully painted.

There are many more examples where a path following controller, as opposed to a trajectory tracking controller, is appropriate for mechanical or robotic systems. Some examples found in the literature include

- robotic deburring [32]
- walking robots [41]
- exercise and rehabilitation machines [32], [61]
- teleoperation [31]
- obstacle avoidance [17]
- human robot interaction [4], [53]

1.2 Literature Review

We are able to group the pertinent literature to this thesis by the underlying themes of each contribution. An attempt has been made to present the literature in a chronological order, so that each idea we present provides a context for the subsequent idea, with a few exceptions. In some ways, the converse is also true; some earlier and seemingly unrelated contributions find their relevance to this thesis through later contributions[1]. Finally, the goal of this literature review is to frame this thesis in the context of the existing body of literature. Therefore, we acknowledge trading off detailed technical accuracy for brevity and clarity.

Path-constrained trajectory planning

Path-constrained trajectory planning represents the largest body of work aimed at achieving path following for robotic manipulators, and was heavily researched in the 1980's. The main idea of this literature is to take the system dynamics and actuator constraints into account when generating trajectories where following a path is important. For example, one can imagine that a bulky robot would have to slow down when taking a sharp turn, otherwise the motors would not be able to exert enough control effort to stay on the path. This approach relies on parameterizing the path, and lends itself naturally to satisfying actuator constraints. For an overview of this idea, see [30, Section 14.3.6].

One of the seminal contributions to this area was by Hollerbach [23]. Here, the author takes an offline trajectory, and suitably re-scales the time parameterization such that actuator constraints are satisfied based on the inverse dynamics. Extensions of this idea are found in a paper by Antonelli *et al.*[2] (and corresponding experiments in [3]), where the

[1]For example, ideas found in Bobrow *et al.* [9] are used in a more relevant contribution by Mettin *et al.* [34], published 24 years later

time scaling occurs online. Related seminal contributors were by Bobrow, Dubowsky and Gibson [8], [9], as well as by Shin and Mckay [43], who independently proposed similar ideas. These works involve generating time-optimal trajectories such that the endpoint of a manipulator is able to traverse a path in output space without violating actuator constraints. A computational improvement on this theme was published by Slotine and Yang [48].

Path-constrained trajectory planning has affinity to the contributions of this thesis primarily in that the goal of this technique is having the output of a robot follow a path faithfully. This idea differs from our approach in that

- only fully actuated manipulators are considered
- they are planning procedures; these techniques are mute on control
- ultimately time-parameterized trajectories must be implemented for control, so although a given path is feasible according to the manipulator dynamics and actuator constraints, disturbances or improper initialization may cause a departure from the path

Integrated Planning and Control

Having recognized deficiencies in path-constrained trajectory planning, namely that time-parameterized trajectory tracking must be implemented, researchers investigated *integrated planning and control*. In the context of performing a task (i.e. following a path), this idea is motivated by the practical consideration of having a manipulator modify its base plan in the presence of unexpected events.

Tarn, Xi, and Bejczy [54] propose a path-based integrated planning and control scheme for robots based on an inner control loop and an outer loop planner, running at the same rate as the controller. The planner generates motion commands based on a path parameterization (the orthogonal projection of the robot output onto the path). Therefore, the parameterization along the path may be viewed as an input to the equivalent system. Similar work is published by Song, Tarn, and Xi [49].

This hierarchical approach to integrating planning and control has a strong affinity to the contribution of this thesis, in that motion along the path is *not* parameterized by time at all. However, this idea differs from our approach in that

- two control loops are required: a low-level inner loop control and an outer loop planner
- this approach, and others like it, do not render the path invariant

- responding to unexpected events is done by the planner based on sensor information, and therefore undetected events cannot be handled

Contour Following

The term *contour following* is ambiguous in the literature. In one instance, contour following is essentially equivalent to path following, as found in work by Li and Horowitz [32]. They address, among other things, the hierarchical, sensor-dependent formulation of path-based integrated planning and control. Their main idea is called "Passive Velocity Field Control", and solves the path following problem for manipulators by encoding a desired velocity field, while preserving a passive relationship between the manipulator output and the environment.

In an earlier work by Huang and McClamroch [24], contour following is presented in the context of manipulators in contact with a surface. Hence, contact forces and changes in dynamics are important features of this work. As such, this particular instance of contour following has little relevance to our contribution.

Contour following perhaps is most prevalent in the control of machine tools. One of the seminal works in this area was by Koren [29], who introduced the idea of a "cross coupling" controller. Extending this idea, Chiu and Tomizuka [12] published interesting work, quite related to the contribution of this thesis. They showed that the tracking errors of a machine tool control system can be decomposed to "normal" and "tangential" errors with respect to a desired contour (path), and they accordingly designed tracking controllers (for example, they suggest more control effort in the normal direction, to make sure the contour is closely followed). Recently Sencer and Altintas [42] applied a sliding-mode based contour following controller to a 5-axis CNC machine. Though this approach is similar to ours in that it decomposes the system dynamics into tangential and normal subsystems, it possesses all of the drawbacks of trajectory tracking. Another difference is that in our approach, we can measure the "contour error" by virtue of *a priori* knowledge of the path, whereas the contour error must be estimated for use in contour following, see for instance Chen and Lee [11] or Erkorkmaz and Altintas [16].

Path parameterization as a reference signal

Parameterizing the path of interest is a common and well-established idea. One approach to designing path following controllers is to use the parameterization as a reference signal just as one would in trajectory tracking. Unlike trajectory tracking however, the velocity of the reference point is treated as an extra control input thus allowing the parameterization of the desired motion along the path be altered. Work by Hauser and Hindman [22] was influential in establishing this method.

Skjetne, Fossen and Kokotović [47] used this idea to solve the path following problem (which they refer to as the *output maneuvering problem*), in which a backstepping approach is used to design a robust controller. The authors provide an example using the motion of a cutting tool, whose model falls into the class of mechanical systems studied in this thesis. Also relevant to this thesis is work done by Aguiar, Hespanha and Kokotović [1], where the authors show that a path following formulation removes performance limitations inherent with trajectory tracking of non-minimum phase systems. In Chapter 5, we study the application of our path following controller to a flexible link robot, which is a non-minimum phase system.

This approach differs from our contribution mainly in that

- it does not guarantee invariance of the closed loop system.
- it is not well-suited to some systems where the dynamic task of controlling the motion along the path may be impossible.

Virtual Holonomic Constraints

A *virtual holonomic constraint* is a relation imposed on the generalized coordinates of a control system. It is virtual because it is preserved by some control action. The virtual holonomic constraints define the path, meaning that satisfying the constraints implies that the system output lies on the path. This has a strong affinity to this thesis, as the objective of satisfying the constraints entails stabilizing a set in the state-space.

In [44], Shiriaev, Perram and Canudas-de-Wit consider systems with degree of underactuation equal to one (such as the inverted pendulum on a cart). They show that the set corresponding to satisfying the virtual holonomic constraints may be stabilized through feedback control. The dynamics along the path, called the *virtual limit system*, are uncontrolled, and the authors provide a procedure to stabilize every feasible periodic motion of this reduced-order system. These ideas were extended by Shiriaev, Freidovich and Gusev [45] to consider mechanical systems with several degrees of underactuation. Interestingly, while for general nonlinear control systems the key difficulty is explicitly finding transverse coordinates, the authors remark that this may be done for mechanical systems. Though this thesis takes a fundamentally different approach, the main results of Chapter 3 also reveal a systematic way of choosing transverse coordinates for mechanical systems.

Mettin, La Hera, Morales, Shiriaev, Freidovich and Westerburg [34] tied together various ideas, including that of virtual holonomic constraints and path-constrained trajectory planning, and applied them to the control of a forestry crane. This literature has a very strong affinity to this thesis. Simulated and experimental results of the manipulator following a circular path are provided. However, this contribution is restricted to a particular manipulator which is kinematically redundant, whereas this thesis considers a much

broader class of systems. Furthermore, virtual holonomic constraints are defined in the configuration space, but the path is defined in the output space of a manipulator. Therefore the path must be composed with the inverse kinematics to get the virtual holonomic constraints. Since the inverse kinematic map does not uniquely map the manipulator's output to its configuration, some additional restrictions must be imposed. Conversely, in this thesis the path following methodology has the advantage of being able to express the path in terms of constraints in the output space of a mechanical system.

1.3 Organization and Contribution

The organization of this thesis, highlighting the contributions, is presented in Figure 1.3.

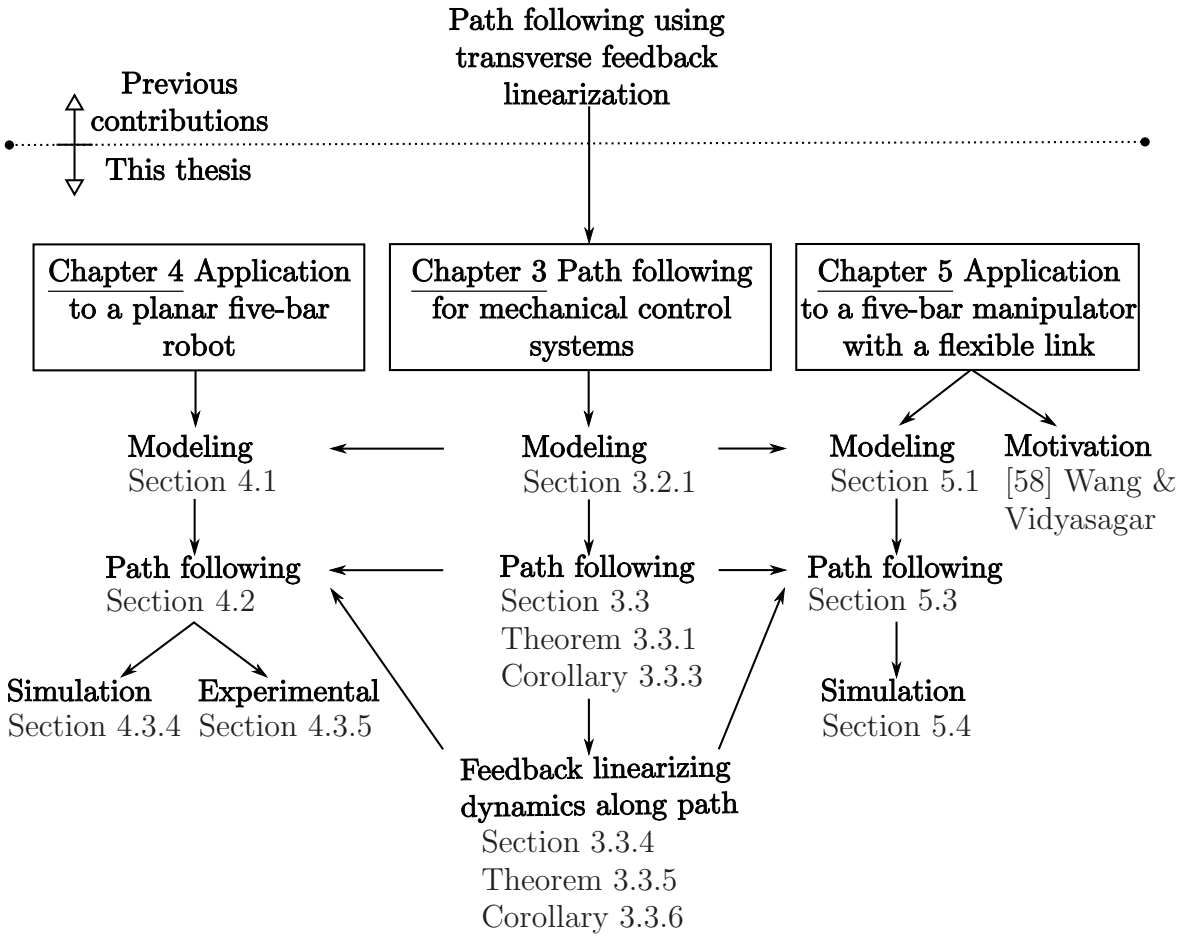


Figure 1.3: A flow chart describing the organization and contributions of this thesis

Chapter 2

Math Preliminaries

This chapter reviews the fundamental mathematical concepts which are heavily relied upon throughout this thesis, drawn from [60], [37], [19], [21], [25] and [27]. First, notation used in this thesis is presented. Next, relevant theory is presented, in particular relating to properties of maps, basics of differential geometry, relevant matrix properties, and derivatives along vector fields. We then introduce some fundamentals of control systems necessary to discuss feedback linearization. Input-output feedback linearization is presented, along with an introduction to the zero dynamics. This is followed by a qualitative discussion on full-state linearization.

2.1 Notation

In this thesis, $\mathbb{N}$ denotes the set of natural numbers, $\mathbb{Z}$ denotes the set of integers, $\mathbb{R}$ denotes the set of real numbers. If k is a positive integer, $\mathbf{k}$ denotes the set of integers $\{1, \ldots, k\}$. The symbol $:=$ means equal by definition. The k-dimensional Euclidean space is denoted $\mathbb{R}^k$. If $x \in \mathbb{R}^k$, we denote by x_i the i^{th} component of x. We let $\text{col}(x_1, \ldots, x_k) := [x_1 \ \cdots \ x_k]^\top$. Let a and b be two column vectors, define $\text{col}(a, b) := [a^\top \ b^\top]^\top$. Unless otherwise indicated, the norm $\|x\|$ of a k-dimensional vector denotes the Euclidean norm, that is,

$$\|x\| := \left(\sum_{i=1}^{k} x_i^2 \right)^{\frac{1}{2}}.$$

We use $\langle x, y \rangle$ to denote the Euclidean inner product of the vectors $x, y \in \mathbb{R}^n$, that is, $\langle x, y \rangle := x^T y$. Given a non-empty set $A \subset \mathbb{R}^n$, and a point $x \in \mathbb{R}^n$, the point-to-set distance is defined as

$$\|x\|_A := \inf_{a \in A} \|x - a\|.$$

Let f be a map from domain $U \subseteq \mathbb{R}^n$ to co-domain $V \subseteq \mathbb{R}^m$, i.e. $f : U \subseteq \mathbb{R}^n \to V \subseteq \mathbb{R}^m$. The notation $f(\cdot)$ indicates the element of the co-domain corresponding to its argument, which belongs to the domain. A scalar-valued function $f : U \subseteq \mathbb{R}^n \to \mathbb{R}$ is k times continuously differentiable at $u \in U$ if it possesses continuous partial derivatives of all orders less than or equal to k. Such a function is of differentiability class C^k on U. If f is C^k for all k, then f is C^∞, or smooth. A map $f : U \to V$, $x \mapsto f(x)$ is C^k if each of its component scalar functions is C^k. Given the map f and a point $p \in \mathbb{R}^n$, we denote $df_p := (\partial f/\partial x)(p)$. The matrix representation of df_p is the Jacobian of f evaluated at p. We denote $I_{m\times m}$ as the $m \times m$ identity matrix, and $0_{n\times m}$ as the $n \times m$ matrix of zeros.

2.2 Elementary Theory

2.2.1 Properties of Maps

A map, or function, is an operator taking elements of its *domain*, and generating elements in its *co-domain*. Surjectivity and injectivity are both fundamental properties of maps, on which we rely heavily in the main results of this thesis.

Definition 2.2.1. A map $f : U \subseteq \mathbb{R}^n \to V \subseteq \mathbb{R}^m$ is *surjective* or *onto* if for each $y \in V$ there exists at least one $x \in U$ such that $f(x) = y$.

Definition 2.2.2. A map $f : U \subseteq \mathbb{R}^n \to V \subseteq \mathbb{R}^m$ is *injective* or *one-to-one* if, given $x_1, x_2 \in U$, $f(x_1) = f(x_2)$ implies $x_1 = x_2$.

Definition 2.2.3. A map $f : U \subseteq \mathbb{R}^n \to V \subseteq \mathbb{R}^m$ is *bijective* (or is a *bijection*) if it is both surjective and injective.

Bijective maps are very powerful, and are the key ingredient in discussing coordinate transformations used in various forms of feedback linearization. However, loosely speaking, when transforming coordinates, we also are interested in somehow preserving smoothness. This leads to a important operator called a *diffeomorphism*.

Definition 2.2.4. A map $f : U \subseteq \mathbb{R}^n \to V \subseteq \mathbb{R}^m$ is a *diffeomorphism* if it is a smooth bijection, and if the inverse map $f^{-1} : V \to U$ is also smooth. If a diffeomorphism exists between U and V, they are said to be diffeomorphic, written $U \simeq V$.

Notice that if $U \simeq V$, this implies that U and V have the same dimension. This notation is used in the main results of this thesis. The concept of a diffeomorphism is central to feedback linearization. According to the previous definition, it is somewhat cumbersome to determine whether a mapping is locally a diffeomorphism. The Inverse Function Theorem enables a remarkable, and less cumbersome method of determining whether a map is locally a diffeomorphism.

Theorem 2.2.1. *(Inverse Function Theorem [25]) Let X be an open set of $\mathbb{R}^n$ and $f : X \to \mathbb{R}^n$, a C^∞ mapping. If df_{x_0} is nonsingular at some x_0 in X, then there exists an open neighbourhood U of x_0 in X such that $V = f(U)$ is open in $\mathbb{R}^n$ and $f|_U$ is a diffeomorphism onto V.*

2.2.2 Differential Geometry Basics

In many textbooks, manifolds are drawn as smooth "blobs" living in a 3-dimensional Euclidean space. In a sense, a manifold is indeed a generalization of smooth surface in $\mathbb{R}^3$, with the key element that around each point on the manifold the space "looks" Euclidean. For example, the earth is (more or less) a sphere, however to us it looks like a flat surface, which can be parameterized by two orthogonal coordinates. The surface of the earth is like a 2-dimensional manifold, however, manifolds may be of higher dimensions. The following two definitions are adapted from [21].

Definition 2.2.5. The set $U \subset \mathbb{R}^n$ is a *k-dimensional manifold* if it is locally diffeomorphic to $\mathbb{R}^k$.

Definition 2.2.6. The manifold $V \subset \mathbb{R}^n$ is a *submanifold* of the manifold $U \subset \mathbb{R}^n$ if $V \subset U$.

The idea of a submanifold is straightforward; it is essentially a manifold "inside" a larger manifold. This leads to the definition of an *embedding*, critical to the main results of this thesis.

Definition 2.2.7. An *embedding* $f : U \to V$ maps U diffeomorphically onto a submanifold of V.

Notice that in the previous definition, U and V are not diffeomorphic, since the dimension of V is higher than the dimension of U. This thesis focuses on path following, and we consider only *embedded* paths.

2.2.3 Matrix Properties

Though it is likely that the reader of this thesis is well-acquainted with properties of matrices, we review some of the commonly used ideas. Matrices are linear operators which take vectors in their domain, and generate vectors in their co-domain.

Definition 2.2.8. Given a linear operator $A : \mathbb{R}^n \to \mathbb{R}^m$, or equivalently a matrix $A \in \mathbb{R}^{m \times n}$, the *image* of A is given by

$$\mathrm{Im}(A) := \{y \in \mathbb{R}^m : Ax = y, \quad \text{for some } x \in \mathbb{R}^n\}.$$

Definition 2.2.9. Given a linear operator $A : \mathbb{R}^n \to \mathbb{R}^m$ the *rank* of A is the dimension of $\mathrm{Im}(A)$.

Theorem 2.2.2. *(Adapted from [19]) The rank of a real-valued matrix A is equal to*

- *the number of linearly independent columns of A.*
- *the largest r for which at least one $r \times r$ submatrix of A has a non-zero determinant.*

The rank of a matrix is one way to tell whether a matrix is surjective or not, which relates to our main results. Another important property of matrix operators is which vectors in the domain map to zero in the co-domain. This notion is important in discussing the feedback transformation in Section 3.1.

Definition 2.2.10. The *kernel* or *null space* of a linear operator $A : \mathbb{R}^n \to \mathbb{R}^m$, is the set $\{x \in \mathbb{R}^n \in U : Ax = 0\}$. In this thesis, the kernel of A is denoted $\ker(A)$.

With scalars, the concept of being "positive" (i.e. > 0) is straightforward. With matrices, the equivalent concept is that of being "positive definite". In the context of this thesis, we appeal to the positive definiteness of generalized mass matrices in Section 3.2.

Definition 2.2.11. A matrix $A \in \mathbb{R}^{n \times n}$ is *positive definite* if, given a vector $x \in \mathbb{R}^n$, $x \neq 0$, $x^T Ax > 0$. It is *positive semi-definite* if $x^T Ax \geq 0$.

Positive definite matrices are invertible. The proof of this fact is in Appendix A.

2.2.4 Vector Fields

In the study of nonlinear systems, we often encounter models of the form

$$\dot{x} = f(x), \quad x \in U \subseteq \mathbb{R}^n, \tag{2.1}$$

where f is a smooth mapping assigning a vector to each point in the state-space (i.e. each $x \in U$). For this reason, f is commonly referred to as a *vector field*.

In this thesis we often need the derivative of a function along a particular vector field. Consider the vector field of (2.1), and a real-valued function $\lambda : U \subseteq \mathbb{R}^n \to \mathbb{R}$. The *derivative of λ along f* is a function $L_f\lambda : U \to \mathbb{R}$, and is equal to the inner product

$$L_f\lambda(x) := \langle d\lambda(x), f(x) \rangle = \frac{\partial \lambda}{\partial x} f(x)$$

which is called the *Lie* or directional derivative of λ along f.

It is possible to group such operations together, for example, taking the Lie derivative of λ first with respect to the vector field f, then with respect to the vector field g would yield

$$L_g L_f \lambda(x) = \frac{\partial L_f \lambda}{\partial x} g(x).$$

Furthermore, this operation can be recursively defined, such that taking k derivatives of λ along f would be denoted $L_f^k \lambda(x)$ where

$$L_f^k \lambda(x) = \frac{\partial L_f^{k-1} \lambda}{\partial x} f(x), \quad \text{with} \quad L_f^0 \lambda(x) = \lambda(x).$$

2.2.5 Control Systems

In this thesis, we study control systems of the form

$$\dot{x} = f(x) + \sum_{i=1}^{m} g_i(x) u_i =: f(x) + g(x)u, \tag{2.2}$$

where $x \in \mathbb{R}^n$ is the state, $u := \text{col}(u_1, \ldots, u_m) \in \mathbb{R}^m$ is the control input and the vector fields $f, g_1, \ldots g_m$ are smooth. Furthermore, consider a function

$$y = h(x) = \begin{bmatrix} h_1(x) \\ \vdots \\ h_p(x) \end{bmatrix}, \quad y \in \mathbb{R}^p, \tag{2.3}$$

which is the output of the system. The idea of relative degree is key in allowing us to change the coordinates of (2.2) into a particularly convenient form.

Definition 2.2.12. System (2.2) with $u \in \mathbb{R}$, and with output $y = h(x)$, $y \in \mathbb{R}$ has *relative degree* r at a point x_0 if

1. $L_g L_f^k h(x) = 0$ for all x in a neighbourhood of x_0 and all $k < r - 1$
2. $L_g L_f^{r-1} h(x_0) \neq 0$

In other words, the relative degree of a single-input single-output (SISO) control system is the number of times the output must be differentiated before the input explicitly appears. In the multi-input multi-output (MIMO) setting, the equivalent concept is the vector relative degree for square[1] systems.

[1] "Square" systems indicate that the number of inputs is equal to the number of outputs, i.e. $p = m$

Definition 2.2.13. System (2.2) with $u \in \mathbb{R}^m$, and with output (2.3) $y \in \mathbb{R}^m$ (i.e. a square MIMO system) has *vector relative degree* $\{r_1, \ldots, r_m\}$ at a point x_0 if

1. $L_{g_j} L_f^k h_i(x) = 0$ for all $1 \leq j \leq m$, for all $k < r_i - 1$, for all $1 \leq i \leq m$ and all x in a neighbourhood of x_0.
2. The $m \times m$ matrix

$$\begin{bmatrix} L_{g_1} L_f^{r_1-1} h_1(x) & \ldots & L_{g_m} L_f^{r_1-1} h_1(x) \\ L_{g_1} L_f^{r_2-1} h_2(x) & \ldots & L_{g_m} L_f^{r_2-1} h_2(x) \\ \vdots & & \vdots \\ L_{g_1} L_f^{r_m-1} h_m(x) & \ldots & L_{g_m} L_f^{r_m-1} h_m(x) \end{bmatrix}$$

   is nonsingular at $x = x_0$.

2.2.6 Input-Output Feedback Linearization

We continue to refer to the square system (2.2) with output (2.3) , $y \in \mathbb{R}^m$, in this section. We remind the reader that, in the interest of brevity, we emphasize main ideas, rather than mathematical rigor. For a rigorous treatment, the interested reader is referred to [25].

SISO input-output feedback linearization

A system whose output yields a well-defined (vector) relative degree can be transformed into a system with a linear input-output map via a coordinate and feedback transformation. Consider the SISO case (i.e. when $m = 1$), and apply the local change of coordinates

$$\begin{aligned} T : & U \to T(U) \\ & x \mapsto (\eta, \xi) \end{aligned}$$

such that

$$T := \begin{bmatrix} \varphi(x) \\ h(x) \\ L_f h(x) \\ \vdots \\ L_f^{r-1} h(x) \end{bmatrix}$$

where $\eta := \varphi(x) = \text{col}(\varphi_1(x), \ldots, \varphi_{n-r}(x))$ and $\xi := \text{col}(h(x), \ldots, L_f^{r-1} h(x))$. The functions $\varphi_1(x), \ldots, \varphi_{n-r}(x)$ are chosen such that locally, T is a diffeomorphism [25]. The state-space

description in new coordinates is

$$\begin{aligned}
\dot{\eta} &= \frac{d\varphi(x(t))}{dt} \\
&= \frac{\partial \varphi(x)}{\partial x}\dot{x} \\
&= \frac{\partial \varphi(x)}{\partial x}\left(f(x)+g(x)u\right)\Big|_{x=T^{-1}(\eta,\xi)} \\
&= \frac{\partial \varphi(x)}{\partial x}f(x)\Big|_{x=T^{-1}(\eta,\xi)} + \frac{\partial \varphi(x)}{\partial x}g(x)u\Big|_{x=T^{-1}(\eta,\xi)} \\
&= p(\eta,\xi)+\sum_{i=1}^{m} q_i(\eta,\xi)u_i \\
&=: p(\eta,\xi)+q(\eta,\xi)u,
\end{aligned}$$

for the η dynamics, whereas for the ξ dynamics we have

$$\begin{aligned}
\dot{\xi}_1 &= \xi_2 \\
&\cdots\cdots \\
\dot{\xi}_{r-1} &= \xi_r \\
\dot{\xi}_r &= L_f^r h(x)\big|_{x=T^{-1}(\eta,\xi)} + L_g L_f^{r-1} h(x) u\big|_{x=T^{-1}(\eta,\xi)} \\
&:= b(\eta,\xi)+a(\eta,\xi)u.
\end{aligned}$$

It can be shown that T^{-1} exists, and moreover that, in the SISO case, $\varphi_1(x),\ldots,\varphi_{n-r}(x)$ can be chosen such that T is a diffeomorphism and that $q(\eta,\xi)\equiv 0$ [25, Proposition 4.1.3]. In this representation, it is clear that with a well-defined relative degree, the regular static feedback

$$u = \frac{1}{a(\eta,\xi)}\left(-b(\eta,\xi)+v\right)$$

will yield the system with linear input-output map

$$\begin{aligned}
\dot{\eta} &= p(\eta,\xi) \\
\dot{\xi}_1 &= \xi_2 \\
&\cdots\cdots \\
\dot{\xi}_{r-1} &= \xi_r \\
\dot{\xi}_r &= v.
\end{aligned}$$

The advantage of a system in this form is that standard linear control techniques may be used to design controllers, assuming that the uncontrollable η dynamics are stable.

MIMO input-output feedback linearization

In the MIMO setting, the concept of *vector relative degree* is analogous to the SISO case and can be used to establish a linear input-output map, through a coordinate transformation and feedback transformation. Considering the MIMO case, we may apply the local change of local coordinates

$$T := \begin{bmatrix} \varphi(x) \\ h_1(x) \\ L_f h_1(x) \\ \cdots\cdots \\ L_f^{r_1-1} h_1(x) \\ h_2(x) \\ L_f h_2(x) \\ \cdots\cdots \\ L_f^{r_2-1} h_2(x) \\ \cdots\cdots \\ L_f^{r_m-1} h_m(x) \end{bmatrix}$$

where $\{r_1, \dots r_m\}$ is the vector relative degree of the system, and $r_1 + \cdots + r_m \leq n$. We define

$$\eta := \varphi(x) = \text{col}\left(\varphi_1(x), \dots, \varphi_{n-\sum_{i=1}^{m} r_i}(x)\right)$$

and

$$\xi := \text{col}(h_1(x), \dots, L_f^{r_1-1} h_1(x), \dots, h_m(x), \dots, L_f^{r_m-1} h_m(x)).$$

In our notation for the ξ-coordinates, ξ_j^i denotes the j^{th} derivative of the i^{th} output. The development of the η dynamics for the MIMO case is similar to the SISO case. Therefore we may express the η dynamics as

$$\dot{\eta} =: p(\eta, \xi) + q(\eta, \xi)u,$$

where in general, it may not be possible to choose T such that $q(\eta, \xi) \equiv 0$ [25]. The ξ dynamics in the MIMO case have a similar form to their SISO counterpart, with the intuitive extension that multiple outputs must be considered. The ξ dynamics are

$$\begin{aligned} \dot{\xi}_1^i &= \xi_2^i \\ \dot{\xi}_2^i &= \xi_3^i \\ &\cdots\cdots \\ \dot{\xi}_{r_i-1}^i &= \xi_{r_i}^i \\ \dot{\xi}_{r_i}^i &= b_i(\eta, \xi) + \sum_{k=1}^{m} a_{ik}(\eta, \xi) u_k \end{aligned}$$

with $i \in \{1, \ldots, m\}$ and where

$$a_{ik}(\eta, \xi) := L_{g_k} L_f^{r_i - 1} h_i(x)\big|_{x = T^{-1}(\eta, \xi)} \quad k \in \{1, \ldots, m\},$$
$$b_i(\eta, \xi) := L_f^{r_i} h_i(x)\big|_{x = T^{-1}(\eta, \xi),}$$

where g_k denotes the k^{th} column of g. The function $\varphi(x)$ can be chosen such that T is a diffeomorphism onto its image [25, Proposition 5.1.2, Lemma 5.1.1]. We define $\bar{\beta}(\eta, \xi) := \operatorname{col}(-b_1(\eta, \xi), \ldots, -b_m(\eta, \xi))$ and

$$\bar{\alpha}(\eta, \xi) := \begin{bmatrix} a_{11}(\eta, \xi) & a_{12}(\eta, \xi) & \cdots & a_{1m}(\eta, \xi) \\ a_{21}(\eta, \xi) & a_{22}(\eta, \xi) & \cdots & a_{2m}(\eta, \xi) \\ \vdots & \vdots & \ddots & \\ a_{m1}(\eta, \xi) & a_{m2}(\eta, \xi) & \cdots & a_{mm}(\eta, \xi) \end{bmatrix}^{-1},$$

in order to arrive at the feedback transformation

$$u = \bar{\alpha}(\eta, \xi)\left(\bar{\beta}(\eta, \xi) + v\right)$$

where $v = \operatorname{col}(v_1, \ldots, v_m)$. This leads to the system with linear input-output map

$$\begin{aligned} \dot{\eta} &= p(\eta, \xi) + q(\eta, \xi) v \\ \dot{\xi}_1^i &= \xi_2^i \\ \dot{\xi}_2^i &= \xi_3^i \\ &\cdots\cdots \\ \dot{\xi}_{r_i - 1}^i &= \xi_{r_i}^i \\ \dot{\xi}_{r_i}^i &= v_i \end{aligned}$$

for $i \in \{1, \ldots, m\}$. Once again, the advantage of a system in this form is that standard linear control techniques may be used design controllers, assuming that the uncontrollable η dynamics are stable.

Zero dynamics

The *zero dynamics* of a system are the dynamics of the system under the constraint $y(t) = 0$ for all t. Recall our output (2.3), and that $\xi_1^i := h_i(x)$. It is straightforward to show that

$$(\forall\, t \geq 0) \quad h(x(t)) = 0 \quad \Leftrightarrow \quad \xi(t) = 0.$$

Since $\xi = 0$, the remaining dynamics are governed by the differential equation

$$\dot{\eta} = p(\eta, 0) + q(\eta, 0) v$$

for an arbitrary initial condition $\eta(0) = \eta^0$. Of course, in order to satisfy the constraint $y(t) = 0$, as with any linear system, the control input, $v(\xi)$, must be chosen such that

$$v(0) = 0.$$

If the zero dynamics are asymptotically stable with respect to an equilibrium point in the domain of interest, the system is said to be *minimum phase*. Otherwise, the system is said to be *non-minimum phase*. This idea is relevant to Chapter 5 of this thesis, in which a model of a flexible link has oscillatory, and hence non-minimum phase, zero dynamics.

2.2.7 Full-state Linearization

So far we have seen that given system (2.2) with output (2.3), it may be locally possible to convert the dynamics (through a coordinate and corresponding feedback transformation) such that the input-output map is linear, with some remaining nonlinear internal dynamics. Suppose the remaining nonlinear η dynamics are undesirable. Is it somehow possible to eliminate the η coordinates altogether? From the discussion on input-output feedback linearization, it is evident that systems with $r = n$ (or $r_1 + \cdots + r_m = n$ in the MIMO case) do not have η coordinates. Therefore we would like to change the (vector) relative degree of the system, which is unfortunately is dependent on the given system output, (2.3). However, nothing is stopping us from using another function, $\lambda(x)$, as an "output", which does not correspond to y, but may change the relative degree such that η dynamics do not exist.

Informally stated, the *State-space Exact Linearization Problem* (or simply "full-state linearization") entails finding a suitable function $\lambda : \mathbb{R}^n \to \mathbb{R}^l$ such that $r = r_1 + r_2 + \cdots + r_l = n$, the dimension of the state-space. We refer the reader to [25, Theorem 5.2.3], which provides conditions under which the *State-space Exact Linearization Problem* is solvable. The relevance of full-state linearization to this thesis lies in Chapter 5, where we discuss the results of a related publication in which the authors attempted full-state linearization on a robotic manipulator with a flexible link.

Chapter 3

Path Following for Mechanical Control Systems

We seek to solve the path following control problem for mechanical control systems. The approach taken in this thesis is to formulate the path following control problem as an instance of set stabilization, where stabilizing an appropriate set in the state space of the mechanical system causes the output of the system to lie on the desired path. In order to stabilize the set of interest, transverse feedback linearization is used. Transverse feedback linearization is a coordinate and feedback transformation that puts a system into a normal form that is particularly convenient for stabilizing sets in the system's state space. Transverse feedback linearization involves finding, if possible, a suitable "virtual" output and then performing partial feedback linearization with the virtual output. In the context of path following, regulating the virtual output is equivalent to driving the real output of the system to the desired path. In this chapter we show that for mechanical control systems, under mild regularity assumptions, there always exists a suitable virtual output so that transverse feedback linearization can be used to design path following controllers.

The transverse feedback linearization normal form ensures that the dynamics *transverse* to the set to be stabilized are linear, time-invariant (LTI) and controllable. For path following, this ensures that the problem of making the system's output approach the desired path is equivalent to stabilizing the LTI transverse dynamics. In general however transverse feedback linearization does not impose any special structure on the dynamics that govern the motion on the set, i.e., the path. In this chapter we show that, for mechanical systems with sufficient actuation and under mild regularity assumptions, the dynamics along the set can be further transformed into a normal form that facilitates control design for achieving desired motions along the path.

3.1 Summary of Path Following via Transverse Feedback Linearization

Recently there has been significant research applying transverse feedback linearization to achieve path following control. We base our approach primarily on the work of Nielsen and Maggiore [36]. In this section, we present a summary of these results. Before proceeding, we emphasize that transverse feedback linearization, as a tool to stabilize sets, has a natural application to path following. One may discuss both path following and transverse feedback linearization separately; however, our approach in this chapter is to present transverse feedback linearization in the context of path following. For a separate treatment, the interested reader is referred to [36], [38].

Consider a control-affine system with m inputs and p outputs,

$$\begin{aligned} \dot{x} &= f(x) + \sum_{i=1}^{m} g_i(x)u_i =: f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{3.1}$$

where $x \in \mathbb{R}^n$ is the state, $u = (u_1, \ldots, u_m)$ is the input and $y \in \mathbb{R}^p$ ($p \geq 2$) is the output. We are given a smooth parameterized curve to follow $\sigma : \mathbb{D} \to \mathbb{R}^p$, where $\mathbb{D}$ is either $\mathbb{R}$, when the curve is not closed, or $\mathbb{D} = \mathbb{S}^1$, when the curve is closed[1]. We impose geometric restrictions on the class of curves, $\sigma(\cdot)$, considered.

Assumption 1. *The curve $\sigma : \mathbb{D} \to \mathbb{R}^p$ enjoys the following properties:*

(i) σ is continuously differentiable (C^r, $r \geq 1$).

(ii) σ is regular, i.e., $\|\sigma'\| \neq 0$.

(iii) $\sigma : \mathbb{D} \to \sigma(\mathbb{D})$ is injective. Note that when $\mathbb{D} = \mathbb{S}^1$ this assumption requires that $\sigma(\mathbb{D})$ be a Jordan curve.

(iv) σ is proper, i.e., for any compact set $K \subset \mathbb{R}^p$, $\sigma^{-1}(K)$ is compact. Note that this assumption is automatically satisfied when $\mathbb{D} = \mathbb{S}^1$.

Assumption 1 guarantees that $\sigma(\mathbb{D})$ is an embedded submanifold of $\mathbb{R}^p$ with dimension 1.

Assumption 2. *There exists a continuously differentiable map $s : \mathbb{R}^p \to \mathbb{R}^{p-1}$ such that 0 is a regular value of s and $\sigma(\mathbb{D}) = s^{-1}(0)$. Moreover the* lift *of $s^{-1}(0)$ to $\mathbb{R}^n$*

$$\Gamma := (s \circ h)^{-1}(0) = \{x \in \mathbb{R}^n : s(h(x)) = 0\}$$

is a submanifold of $\mathbb{R}^n$.

[1]The notation $\mathbb{S}$ means $\mathbb{R} \bmod 2\pi$, the real numbers modulo 2π. On the set $\mathbb{S}^1$ two different real numbers x and $x + 2\pi$ are considered to be the same point. Thus $\mathbb{S}^1$ has the geometric structure of a circle.

By Assumption 1, the path σ, given as a parameterized curve, is a submanifold of $\mathbb{R}^p$ and hence the first part of Assumption 2 is always satisfied locally. Assumption 2 requires that the *entire* path can be represented as the zero level set of the function s in the output space of system (3.1)

$$\{y \in \mathbb{R}^p : s(y) = 0\} = \sigma(\mathbb{D}\,).$$

If we let $\gamma := s^{-1}(0)$, then Assumption 2 is equivalent to requiring that one can choose $s : \mathbb{R}^p \to \mathbb{R}^{p-1}$ so that its Jacobian has full rank $p-1$ at each point of γ. With regards to the second part of Assumption 2, a sufficient condition for

$$\Gamma = \{x : s_1(h(x)) = \cdots = s_{p-1}(h(x)) = 0\}$$

to be a submanifold of $\mathbb{R}^n$ is that the map $h : \mathbb{R}^n \to \mathbb{R}^p$ be *transversal* to the submanifold $s^{-1}(0)$ [21]. By Assumption 2 the dimension of Γ is $n - p + 1$. Intuitively, we see that making $x \to \Gamma$ is equivalent to making $y \to \gamma$.

Our objective is to design a smooth feedback which makes the output of the system (3.1) approach the path γ, and traverse it with a desired speed and direction. Additionally, we seek to render γ *output invariant*. Intuitively, output invariance ensures that if the state x of the closed-loop system is properly initialized with $y(0) = h(x(0)) \in \gamma$, then output signal $y(t)$ will remain on the path γ for all future time.

A natural question to ask is whether the path $\sigma(\mathbb{D}\,)$ is feasible for (3.1). This is equivalent to asking if there is a subset of Γ that can be stabilized. In general one may only be able to stabilize a subset Γ by suitable choice of the control input. Accordingly, for path following we seek to stabilize the largest controlled invariant submanifold of Γ, denoted by $\Gamma^\star$. Intuitively $\Gamma^\star$ is the collection of all motions of system (3.1) whose outputs can be made to lie in γ for all time given a suitable control input.

Assumption 3. *The maximal controlled-invariant subset of* $\Gamma = \{x \in \mathbb{R}^n : s(h(x)) = 0\}$, $\Gamma^\star$*, is a non-empty, closed embedded submanifold of the state space. Let the dimension of* $\Gamma^\star$ *be denoted* $n^\star$

Assumption 3 is a basic feasibility requirement needed to solve the path following problem for system (3.1) using transverse feedback linearization. If this assumption does not hold then there does not exist a smooth feedback that makes the output of system (3.1) stay on the path γ for all time. With this assumption, $\Gamma^\star$ is precisely the zero dynamics manifold of the control system $\dot{x} = f(x) + g(x)u$ with output $\hat{y} = s(h(x))$. Since the set $\Gamma^\star$ plays such an important role in the subsequent development, we formally define it.

Definition 3.1.1. The *path following manifold*, $\Gamma^\star$, of γ with respect to (3.1) is the maximal controlled-invariant submanifold contained in $(s \circ h)^{-1}(0)$.

We now give a precise definition of what it means for a path to be output invariant for a control system.

Definition 3.1.2. Let $\bar{u}(x)$ be a smooth feedback and let $\Gamma^\star$ be the path following manifold of γ with respect to (3.1). The path γ is *output invariant* under the closed loop vector field $\bar{f} := f + g\bar{u}$ if $\Gamma^\star$ is invariant under $\bar{f}$.

The following example illustrates the above concepts.

Example 3.1.1. Consider a magnetized puck of mass M on a frictionless plane as shown in Figure 3.1. In this fictitious system, we assume that two electromagnets may attract and/or repel the magnetized puck by directly applying the orthogonal forces[2] τ_1 and τ_2.

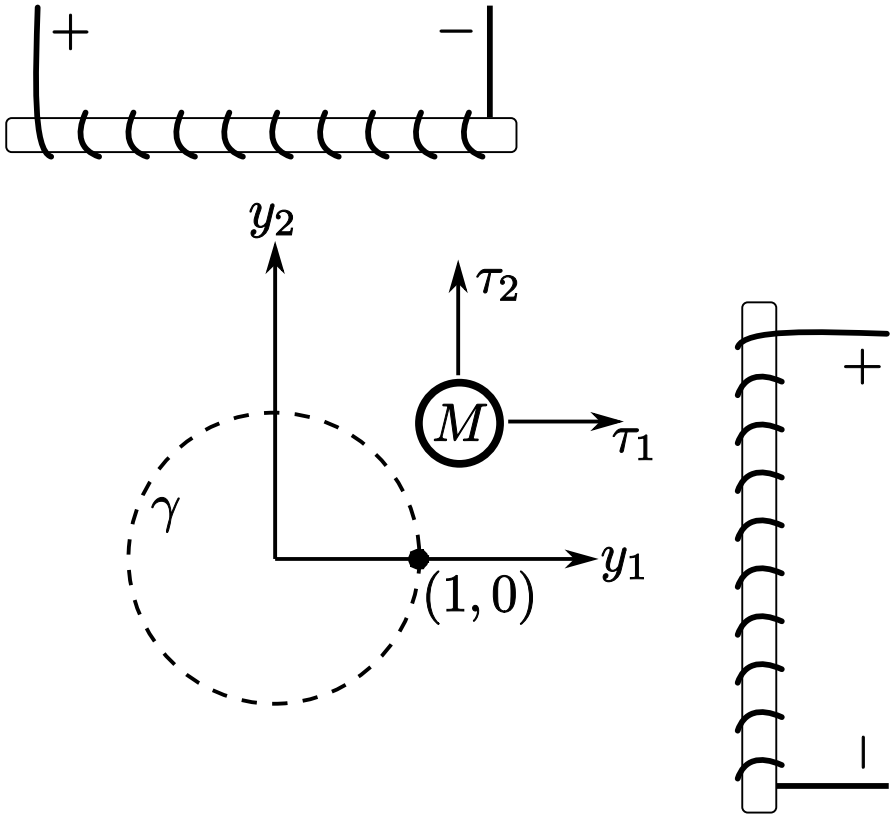


Figure 3.1: A simple mechanical control system

[2]The electrical dynamics are neglected for simplicity, and we treat the control inputs directly as forces

We take the state $x = \text{col}(x_1, x_2, x_3, x_4)$ of this system to be the positions and velocities of the puck mass in the y_1 and y_2 directions. The model of this linear system is

$$\begin{aligned} \dot{x}_1 &= x_3 \\ \dot{x}_2 &= x_4 \\ \dot{x}_3 &= \tfrac{1}{M}\tau_1 \\ \dot{x}_4 &= \tfrac{1}{M}\tau_2. \end{aligned}$$

We choose our output $y = \text{col}(y_1, y_2)$ as the position of the mass in the plane, such that

$$\begin{aligned} y_1 &= x_1 \\ y_2 &= x_2. \end{aligned}$$

Our path in output space is the unit circle, i.e. $\gamma = \{y \in \mathbb{R}^2 : s(y) = y_1^2 + y_2^2 - 1 = 0\}$.

In this example, $\Gamma = \{x \in \mathbb{R}^4 : x_1^2 + x_2^2 - 1 = 0\}$, which is a cylinder in 4-dimensional space. To illustrate the relationship between Γ and $\Gamma^\star$ for this system, in reference to Figure 3.1, consider an initial condition $x_0 = \text{col}(1, 0, \alpha, \beta) \in \Gamma, \quad \alpha, \beta \in \mathbb{R}$. We see that Γ is not an invariant set, as any velocity $\alpha \neq 0$, which is in the y_1 direction, will cause x to leave Γ. For this example, intuitively $\Gamma^\star$ must be the largest subset of Γ for which the puck velocities are tangent to γ.

$\triangle$

Now we state the path following problem (PFP), based on [38], which entails finding a smooth feedback such that the following objectives are met:

P1 For each initial condition in an open neighbourhood of $\Gamma^\star$, the corresponding solution $x(t)$ of (3.1) is defined for all $t \geq 0$ and $\|h(x(t))\|_\gamma \to 0$ as $t \to +\infty$

P2 The set γ is output invariant for the closed-loop system

P3 The motion on γ meets additional application specific requirements such as direction and speed of traversal of the path, and boundedness of the internal dynamics.

The approach we take in solving the PFP (see [38]) is summarized below.

Step 1 Find the path following manifold $\Gamma^\star$.

Step 2 Find, if possible, a coordinate transformation $T : x \mapsto (\eta, \xi)$, defined in a neighbourhood U of $\Gamma^\star$, and a regular feedback transformation $u = \alpha(x) + \beta(x)v$, ($\beta$ non-singular on U) such that $T(\Gamma^\star) = \{(\eta, \xi) : \xi = 0\}$ and, in new coordinates,

$$\begin{aligned} \dot{\eta} &= f^0(\eta, \xi) + g^\pitchfork(\eta, \xi)v^\pitchfork + g^\parallel(\eta, \xi)v^\parallel \\ \dot{\xi} &= A\xi + Bv^\pitchfork \end{aligned} \tag{3.2}$$

with $v := \mathrm{col}(v^{\pitchfork}, v^{\parallel}) \in \mathbb{R}^m$ and (A, B) a controllable pair. The ξ-subsystem is referred to as the *transversal subsystem*. On the other hand, the system $\dot{\eta} = f^0(\eta, 0) + g^{\parallel}(\eta, 0)v^{\parallel}$ is the *tangential subsystem*.

Step 3 Design a transversal feedback $v^{\pitchfork}(\xi)$ stabilizing the origin of the transversal subsystem. Note that stabilizing the origin of the transversal subsystem is equivalent to stabilizing $\Gamma^\star$, which corresponds to getting the output of (3.1) onto the path.[3]

Step 4 Design a tangential feedback $v^{\parallel}(\eta, \xi)$ such that, when $\xi = 0$, the tangential subsystem meets the applications-specific goals in **P3**, and moreover, the closed-loop system has no finite escape times [27].

In summary, our goal is to solve the PFP, and we set out to do so by stabilizing a set in the state space called the path following manifold denoted $\Gamma^\star$. On the path following manifold, the output of the system lies on the path. We use transverse feedback linearization to stabilize this set. There are other approaches to stabilizing $\Gamma^\star$, but transverse feedback linearization has particularly attractive features, among them

- As the name implies, and as indicated in (3.2), the dynamics of the transversal subsystem are linear, time-invariant and controllable which simplifies control design in order to accomplish goal **P2**.
- Objectives **P1** and **P2** are decoupled from objective **P3**, meaning controllers for the transversal and tangential subsystems may be independently designed.

Theorem 3.1.1 below states that transverse feedback linearization essentially amounts to partial feedback linearization with the additional requirement that the linear subsystem be representative of the dynamics of the system transversal to the target set $\Gamma^\star$. Furthermore, notice that the dynamics of the η-subsystem of (3.2) are left in a general form. Transverse feedback linearization imposes no structure on the η-subsystem, so that, in general, meeting **P3** may be difficult or impossible. According to **Step 4**, the objective is to somehow control the tangential dynamics of the transversely feedback linearized system. We will show that, for mechanical systems, the η dynamics can be given more structure that will help design controllers for **Step 4**. This is the subject of Section 3.3.2.

We now proceed in applying **Steps 1-4** to solve the PFP. In **Step 1** we must find $\Gamma^\star$, the path following manifold. This may be done using the zero dynamics algorithm described in [25].

[3]Set stabilization is not a coordinate invariant property. In particular, stabilizing x to $\Gamma^\star$ is equivalent to stabilizing (η, ξ) to $T(\Gamma^\star)$ if the trajectory $x(t)$ is bounded. If $x(t)$ is unbounded, say because the path itself is unbounded, then additional design constraints must be imposed.

Step 2 involves finding a coordinate transformation and a regular feedback transformation such that our system (3.1) is decomposed into a linear transversal subsystem, and a remaining tangential subsystem. We appeal to the following theorem, adapted from [36].

Theorem 3.1.1. *System* (3.1) *is locally transversely feedback linearizable at a point* $x^\star \in \Gamma^\star$ *if and only if there exist* l *smooth functions* $\lambda_1(x), \lambda_2(x), \ldots, \lambda_l(x)$, $1 \leq l \leq m$, *with the following properties:*

(a) $\Gamma^\star \subset \{x : \lambda_1(x) = \cdots = \lambda_l(x) = 0\}$.

(b) The "virtual output" $\hat{y} := \lambda(x) = \mathrm{col}(\lambda_1(x), \ldots, \lambda_l(x))$ *yields a uniform vector relative degree* $\{k_1, \ldots, k_l\}$ *at* $x^\star \in \Gamma^\star$, *and the indices* k_i *are such that* $k_1 + \cdots + k_l = n - n^\star$.

In general, finding the virtual output may be a difficult task. The existence of such an output can be determined using Theorem 3.2 in [36], where checkable necessary and sufficient conditions are presented. However, a good first guess is to try $\hat{y} = s(h(x))$, since it already satisfies property (a) above. This will be our approach in Section 3.3. We will provide necessary and sufficient conditions for this choice of virtual output with a mechanical system to satisfy Theorem 3.1.1.

Suppose that we have found a suitable virtual output $\lambda(x)$ that satisfies Theorem 3.1.1. We now show how we can use this virtual output to put system (3.1) into the normal form (3.2). By [25, Proposition 5.1.1], the $n - n^\star$ functions

$$\lambda_1(x), \ldots, L_f^{k_1-1}\lambda_1(x), \ldots, \lambda_l(x), \ldots, L_f^{k_l-1}\lambda_l(x)$$

have linearly independent differentials at $x^\star$. By [25, Proposition 5.1.2] it is possible to find an additional $n^\star$ real-valued functions $\varphi_i(x)$, $i \in \{1, \ldots, n^\star\}$ whose differentials are linearly independent at $x^\star$ so that, by the inverse function theorem, Theorem 2.2.1, there exists a neighbourhood U of $x^\star$ such that the mapping

$$\begin{aligned} T :& U \to T(U) \\ & x \mapsto T(x) = \\ & \qquad \mathrm{col}\left(\varphi_1(x), \ldots, \varphi_{n^\star}(x), \lambda_1(x), \ldots, L_f^{k_1-1}\lambda_1(x), \ldots, \lambda_l(x), \ldots, L_f^{k_l-1}\lambda_l(x)\right) \end{aligned} \tag{3.3}$$

is a local diffeomorphism at $x^\star$. Set

$$\eta = \varphi(x) = \mathrm{col}\,(\varphi_1(x), \ldots, \varphi_{n^\star}(x)),$$

$$\xi^i = \mathrm{col}\,(\xi_1^i, \ldots, \xi_{k_i}^i) = \mathrm{col}\,(\lambda_i(x), \ldots, L_f^{k_i-1}\lambda(x))$$

for $i \in \{1, \ldots, l\}$, and let $\xi = \operatorname{col}(\xi^1, \ldots, \xi^l)$. Then in (η, ξ) coordinates the η dynamics are given by

$$\begin{aligned}
\dot{\eta} &= \frac{d\varphi(x(t))}{dt} \\
&= \frac{\partial \varphi(x)}{\partial x} \dot{x} \\
&= \frac{\partial \varphi(x)}{\partial x} \left(f(x) + g(x)u \right) \Big|_{x = T^{-1}(\eta, \xi)} \\
&= \frac{\partial \varphi(x)}{\partial x} f(x) \Big|_{x = T^{-1}(\eta, \xi)} + \frac{\partial \varphi(x)}{\partial x} g(x) u \Big|_{x = T^{-1}(\eta, \xi)} \\
&= p(\eta, \xi) + \sum_{i=1}^{m} q_i(\eta, \xi) u_i \\
&=: p(\eta, \xi) + q(\eta, \xi) u.
\end{aligned}$$

On the other hand the ξ dynamics are given by

$$\begin{aligned}
\dot{\xi}_1^i &= \xi_2^i \\
\dot{\xi}_2^i &= \xi_3^i \\
&\cdots\cdots \\
\dot{\xi}_{k_i - 1}^i &= \xi_{k_i}^i \\
\dot{\xi}_{k_i}^i &= b_i(\eta, \xi) + \sum_{j=1}^{m} a_{ij}(\eta, \xi) u_j
\end{aligned}$$

with $i \in \{1, \ldots, l\}$ and where

$$\begin{aligned}
a_{ij}(\eta, \xi) &:= L_{g_j} L_f^{k_i - 1} \lambda_i(x) \big|_{x = T^{-1}(\eta, \xi)} \\
b_i(\eta, \xi) &:= L_f^{k_i} \lambda_i(x) \big|_{x = T^{-1}(\eta, \xi),} \qquad\qquad j \in \{1, \ldots, m\}.
\end{aligned}$$

Therefore the coefficient multiplying the control input u_j in the equation for $\dot{\xi}_{k_i}^i$ is the $(i, j)^{\text{th}}$ entry of the so-called decoupling matrix

$$D(x) := \begin{bmatrix} L_g L_f^{k_1 - 1} \lambda_1 \\ L_g L_f^{k_2 - 1} \lambda_2 \\ \vdots \\ L_g L_f^{k_l - 1} \lambda_l \end{bmatrix} = \begin{bmatrix} L_{g_1} L_f^{k_1 - 1} \lambda_1 & \cdots & L_{g_m} L_f^{k_1 - 1} \lambda_1 \\ L_{g_1} L_f^{k_2 - 1} \lambda_2 & \cdots & L_{g_m} L_f^{k_2 - 1} \lambda_2 \\ \vdots & & \\ L_{g_1} L_f^{k_l - 1} \lambda_l & \cdots & L_{g_m} L_f^{k_l - 1} \lambda_l \end{bmatrix}. \tag{3.4}$$

Let $(\eta^\star, \xi^\star) = T(x^\star)$ and recall that by Theorem 3.1.1, $D(x)\big|_{x = T^{-1}(\eta, \xi)}$ is full rank at $(\eta^\star, \xi^\star)$ which means that it is full rank in a neighbourhood of $(\eta^\star, \xi^\star)$. In general, the decoupling

matrix (3.4) is not square[4], but has dimensions $l \times m$. This suggests the following regular feedback transformation. Let $\beta(x) = [M(x)N(x)]$ where $M(x) := D(x)^T(D(x)D(x)^T)^{-1}$ is a right inverse of $D(x)$ and where $N(x)$ is an $m \times (m-l)$ smooth matrix-valued function whose columns span the kernel of $D(x)$ for all x near $x^\star$. Notice that $\beta(x)$ is just defined is non-singular near $x^\star$. Finally, let

$$\alpha(x) := -\beta(x)\,\mathrm{col}(L_f^{k_1}\lambda_1(x), \ldots, L_f^{k_l}\lambda_l(x), 0_{(m-l)\times 1}).$$

so that our regular feedback transformation is

$$u = \alpha(x) + \beta(x)v, \tag{3.5}$$

where $v = \mathrm{col}(v^{\pitchfork}, v^{\parallel})$ with $v^{\pitchfork} \in \mathbb{R}^l$ and $v^{\parallel} \in \mathbb{R}^{m-l}$. After applying this feedback transformation, the η dynamics are given by

$$\begin{aligned}
\dot{\eta} &= p(\eta,\xi) + q(\eta,\xi)\left(\alpha(x) + \beta(x)v\right)_{x=T^{-1}(\eta,\xi)} \\
&= p(\eta,\xi) + q(\eta,\xi)\alpha(x) + q(\eta,\xi)\beta(x)\left[\begin{array}{c} v^{\pitchfork} \\ v^{\parallel} \end{array}\right]\Bigg|_{x=T^{-1}(\eta,\xi)} \\
&=: f^0(\eta,\xi) + g^{\pitchfork}(\eta,\xi)v^{\pitchfork} + g^{\parallel}(\eta,\xi)v^{\parallel}.
\end{aligned}$$

The ξ dynamics are given by

$$\begin{aligned}
\dot{\xi}_1^i &= \xi_2^i \\
\dot{\xi}_2^i &= \xi_3^i \\
&\cdots\cdots \\
\dot{\xi}_{k_i-1}^i &= \xi_{k_i}^i \\
\dot{\xi}_{k_i}^i &= v_i^{\pitchfork}
\end{aligned}$$

for $i \in \{1, \ldots, l\}$. If we let $\xi = \mathrm{col}\,(\xi^1, \ldots, \xi^l)$ then the ξ dynamics can be written compactly as

$$\dot{\xi} = A\xi + Bv^{\pitchfork}$$

where the pair (A, B) are in Brunovský normal form. In conclusion, using the functions $\lambda(x)$ that satisfy the conditions of Theorem 3.1.1, we generate the coordinate transformation (3.3) and the feedback transformation (3.5) to obtain a system with the desired form.

Step 3 involves stabilizing the origin of the ξ-subsystem. One may appeal to standard linear feedback control design techniques to accomplish this. Finally, **Step 4** involves controlling the tangential dynamics, if possible. There are no guarantees on the structure of the η dynamics. However, in some systems it may be possible to impose further structure on these dynamics by appropriate choice of the functions $\varphi(x)$ and the input $v^{\parallel}$.

[4]In contrast, when performing input-output feedback linearization for systems where $p = m$, the corresponding decoupling matrix is square, and, given a well-defined relative degree at a point, invertible.

3.2 Mechanical Control Systems

3.2.1 Modeling

A standard method of deriving the equations of motion for mechanical control systems is via the Euler-Lagrange equations. For a simple mechanical control system with internal dissipation (i.e. friction) the Euler-Lagrange equations are

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}(q,\dot{q})\right) - \frac{\partial L}{\partial q}(q,\dot{q}) + \frac{\partial R}{\partial \dot{q}} = \tilde{\tau} \tag{3.6}$$

where $q = \mathrm{col}\,(q_1, \ldots, q_N)$ are generalized configuration coordinates and $\tilde{\tau} \in \mathbb{R}^N$ is the vector of generalized forces acting on the system [50]. We refer to $\tilde{\tau}$ as the inputs to mechanical control system.

Definition 3.2.1. A mechanical system (3.6) has N *degrees of freedom* (DOF) if N generalized configuration coordinates q are necessary to model the system [50].

Definition 3.2.2. The *configuration space* of a mechanical system is an N-dimensional manifold which is the collection of all possible system configurations.

The main results in this chapter are local and so we will assume, without loss of generality, that the configuration manifold is $\mathbb{R}^N$. The Lagrangian function

$$L(q,\dot{q}) = K(q,\dot{q}) - V(q) \tag{3.7}$$

equals the difference between kinetic energy K and potential energy V. In standard mechanical systems the kinetic energy K is of the form $K(q,\dot{q}) = \frac{1}{2}\dot{q}^\top M(q)\dot{q}$, where the $N \times N$ inertia (generalized mass) matrix $M(q)$ is symmetric and positive definite for all q. The function $R(\dot{q})$ is a Rayleigh dissipation function that satisfies [57]

$$\dot{q}^T \frac{\partial R}{\partial \dot{q}} \geq 0.$$

The Euler-Lagrange in coordinates are commonly expressed as

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tilde{\tau}, \tag{3.8}$$

where matrix $C(q,\dot{q})$ represents the velocity-dependent Coriolis, centripetal and dissipation terms. The function $G(q)$ represents the mapping to generalized forces from the system configuration, which typically includes terms describing the effect of gravity, and other configuration-dependent forces.

Since $M(q)$ is positive definite, $M^{-1}(q)$ exists for every q (see Appendix A for proof). Physically this is because mechanical bodies must have positive masses and rotational inertias. Thus we may multiply both sides of (3.8) by $M^{-1}(q)$ to get

$$\ddot{q} = -M^{-1}C(q,\dot{q})\dot{q} - M^{-1}G(q) + M^{-1}\tilde{\tau}. \tag{3.9}$$

To facilitate the subsequent discussion we will convert the equations of motion, (3.9), into state space form and distinguish between configuration and velocity states by defining

$$\begin{aligned} x_c &:= q \\ x_v &:= \dot{q}, \end{aligned}$$

and

$$\begin{aligned} x :=& (x_c, x_v) \\ =& \operatorname{col}\left(x_{c_1}, \ldots x_{c_N}, x_{v_1}, \ldots, x_{v_N}\right) \\ =& \operatorname{col}\left(q_1, \ldots, q_N, \dot{q}_1, \ldots \dot{q}_N\right). \end{aligned}$$

Setting $n = 2N$, the state x is an element of $\mathbb{R}^n$. Since $\tilde{\tau} \in \mathbb{R}^N$ has only $0 \leq m \leq N$ independent applied forces, we set $g_v(x_c)\tau := M^{-1}(x_c)\tilde{\tau}$, where $g_v(x_c) \in \mathbb{R}^{N\times m}$ and $\tau \in \mathbb{R}^m$. We also set $f_v(x) := -M^{-1}(x_c)C(x_c,x_v)x_v - M^{-1}(x_c)G(x_c)$, where $f_v(x) \in \mathbb{R}^{N\times 1}$. The equations of motion, (3.9), in state space form are now concisely expressed as

$$\begin{aligned} \dot{x} &= f(x) + g(x)\tau \\ &:= \begin{bmatrix} x_v \\ f_v(x) \end{bmatrix} + \begin{bmatrix} 0_{N\times m} \\ g_v(x_c) \end{bmatrix} \tau, \end{aligned} \tag{3.10}$$

where $f(x)$ has dimension $n \times 1$ and $g(x)$ has dimension $n \times m$. Mechanical control systems are often classified based on the number of independent control inputs they have relative to their degrees of freedom.

Definition 3.2.3. A mechanical system with m inputs and N degrees of freedom is said to be

(a) *fully actuated* if $m = N$ or

(b) *underactuated* if $m < N$.

The output equation of a control system is typically used to model the variable we are interested in controlling and/or the information available for feedback. In this chapter we take the former view and treat the output of a mechanical system as the variable we are interested in controlling. A typical example is the forward kinematics of a robot, i.e., the configuration of the end-effector given the relative configurations of each pair of adjacent

links of the robot. We restrict the class of output functions to be solely functions of the configuration variables $x_c = q$. Hence we take as the output equation of system (3.10) an equation of the form

$$y = h(x_c), \quad y \in \mathbb{R}^p \tag{3.11}$$

where h is smooth. We will call the function $h(x_c)$ the forward kinematic map, or simply the forward kinematics of the mechanical system, which maps from system configuration to the output space. We will assume throughout this chapter that $m \geq p - 1$.

Definition 3.2.4. A mechanical system is said to be *kinematically redundant* if $N > p$ [46].

For example, a 3-link planar elbow manipulator is kinematically redundant if the objective is specifying endpoint location ($N = 3, p = 2$). It is not kinematically redundant if the objective is specifying endpoint location and orientation ($N = 3, p = 3$).

We are interested in designing controllers that solve the path following problem for mechanical systems, using the approach in Section 3.1. Thus we restrict the dimension of the output be greater than one. For the problem considered in this thesis, a 1-dimensional output is not particularly useful because any curve that satisfies Assumption 1 with $p = 1$ will necessarily satisfy $\sigma(\mathbb{D}) = \mathbb{R}$. Thus the problem considered here reduces to the trivial problem of making y stay on the real line.

3.3 Path Following for Mechanical Systems

We now solve the path following control problem for the class of simple mechanical systems using transverse feedback linearization. To this end, we constrain the theory presented in Section 3.1 to the class of systems presented in Section 3.2.1, whose model we restate as

$$\dot{x} = f(x) + g(x)\tau. \tag{3.12}$$

Including the forward kinematics, and imposing the structure associated with simple mechanical systems, the structured model is given by

$$\begin{aligned} \dot{x}_c &= x_v \\ \dot{x}_v &= f_v(x) + g_v(x_c)\tau \\ y &= h(x_c), \end{aligned} \tag{3.13}$$

where $x_c \in \mathbb{R}^N, x_v \in \mathbb{R}^N, \tau \in \mathbb{R}^m$, $n = 2N$, $y \in \mathbb{R}^p$ and $m \geq p - 1$. We now present the main result.

Main result

Theorem 3.3.1. *Given a simple mechanical control system* (3.13) *and a smooth embedded path* γ *in output space satisfying Assumptions 1 and 2, let* $x^\star = \mathrm{col}(x_c^\star, x_v^\star) \in \mathbb{R}^n$ *satisfy* $h(x_c^\star) \in \gamma$*. The system*

$$\begin{aligned} \dot{x}_c &= x_v \\ \dot{x}_v &= f_v(x) + g_v(x_c)\tau \\ \hat{y} &= s \circ h(x_c) \end{aligned} \tag{3.14}$$

yields a well-defined vector relative degree of $\{2, \ldots, 2\}$ *at* $x^\star$ *if and only if*

$$\mathrm{Im}\left(\left.\frac{\partial h}{\partial x_c}\right|_{x_c = x_c^\star} g_v(x_c^\star) \right) + \ker\left(\left.\frac{\partial s}{\partial y}\right|_{y = h(x_c^\star)} \right) \simeq \mathbb{R}^p. \tag{3.15}$$

Proof. Let $x^\star = (x_c^\star, x_v^\star) \in \mathbb{R}^n$ be any point that satisfies $h(x_c^\star) \in \gamma$. Assume that (3.15) holds at $x^\star$. We first show that the virtual output

$$\hat{y} = \lambda(x_c) := s \circ h(x_c) \tag{3.16}$$

yields a well-defined relative degree at $x^\star$ by showing that the decoupling matrix is full rank at $x^\star$. Taking derivatives of $\hat{y}$, simple calculations reveal

$$\begin{aligned} \frac{d\hat{y}}{dt} &= \frac{\partial \lambda}{\partial x}\left(f(x) + g(x)\tau\right) \\ &= \begin{bmatrix} \frac{\partial \lambda}{\partial x_c} & \frac{\partial \lambda}{\partial x_v} \end{bmatrix} \left(\begin{bmatrix} x_v \\ f_v(x) \end{bmatrix} + \begin{bmatrix} 0_{N\times m} \\ g_v(x_c) \end{bmatrix} \tau, \right) \\ &= \begin{bmatrix} \frac{\partial \lambda}{\partial x_c} & 0 \end{bmatrix} \left(\begin{bmatrix} x_v \\ f_v(x) \end{bmatrix} + \begin{bmatrix} 0_{N\times m} \\ g_v(x_c) \end{bmatrix} \tau, \right) \\ &= \frac{\partial \lambda}{\partial x_c} x_v. \end{aligned}$$

Since the term multiplying the control input is identically zero, we take a second derivative.

$$\begin{aligned} \frac{d^2\hat{y}}{dt^2} &= \frac{\partial \frac{\partial \lambda}{\partial x_c} x_v}{\partial x}\left(f(x) + g(x)\tau\right) \\ &= \begin{bmatrix} \frac{\partial \frac{\partial \lambda}{\partial x_c} x_v}{\partial x_c} & \frac{\partial \lambda}{\partial x_c} \end{bmatrix} \left(\begin{bmatrix} x_v \\ f_v(x) \end{bmatrix} + \begin{bmatrix} 0_{N\times m} \\ g_v(x_c) \end{bmatrix} \tau, \right) \\ &= \frac{\partial \frac{\partial \lambda}{\partial x_c} x_v}{\partial x_c} x_v + \frac{\partial \lambda}{\partial x_c} f_v(x) + \frac{\partial \lambda}{\partial x_c} g_v(x_c)\tau \\ &= L_f^2 \lambda(x) + \frac{\partial \lambda}{\partial x_c} g_v(x_c)\tau. \end{aligned}$$

We now show that (3.15) implies that the $p-1 \times m$ matrix valued function multiplying the control input τ is full rank at $x^\star$ and hence that λ yields a well-defined vector relative degree of $\{2, \ldots, 2\}$. By the chain rule

$$\frac{\partial \lambda}{\partial x_c} g_v(x_c) = \left.\frac{\partial s}{\partial y}\right|_{y=h(x_c)} \frac{\partial h}{\partial x_c} g_v(x_c).$$

Since $m \geq p-1$, the best we can hope for is that this matrix is surjective, or onto (full row rank). By Assumptions 1 and 2, the $p-1 \times p$ matrix $\frac{\partial s}{\partial y}$ is full rank (onto) at each point on the path, in particular, at $y = h(x_c^\star)$. Condition (3.15) implies that any vector in the pre-image of the map $\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c^\star)}$ is in the image of the map $\frac{\partial h}{\partial x_c^\star} g_v(x_c^\star)$. Putting these facts together we get that

$$\operatorname{Im}\left(\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c^\star)} \left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)\right) = \operatorname{Im}\left(\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c^\star)}\right) = \mathbb{R}^{p-1},$$

which shows that the decoupling matrix is full rank at each $x^\star$ with $h(x_c^\star) \in \gamma$.

Now assume that the virtual output (3.16) yields a well-defined vector relative degree of $\{2, \ldots, 2\}$ at $x^\star \in \mathbb{R}^n$. By definition this means that the $p-1 \times m$ matrix

$$L_g L_f \lambda(x) = \frac{\partial s}{\partial y} \frac{\partial h}{\partial x_c} g_v(x_c)$$

is full rank at $x^\star$. Let $A := \left.\frac{\partial s}{\partial y}\right|_{y=h(x_c^\star)}$ and let $B := \left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)$. Let $\mathscr{R} := \ker A$ and let $\mathscr{S}$ be any independent subspace[5] so that $\mathbb{R}^p = \mathscr{R} \oplus \mathscr{S}$. Then any vector v in $\mathbb{R}^p$ can be written uniquely as $v = r + s$ with $r \in \mathscr{R}$ and $s \in \mathscr{S}$.

Since the decoupling matrix AB is full rank (onto) at $x^\star$, it follows that for any vector $w \in \mathbb{R}^{p-1}$, there exists a $u \in \mathbb{R}^m$ such that $w = ABu$. The vector $Bu \in \mathbb{R}^p$ can be written uniquely as $Bu = r + s$ so that $w = ABu = A(r+s) = Ar + As = As$. Note that the vector r is in the image of B. Since w is arbitrary, we have shown that any vector in $\mathbb{R}^p$ can be written as a linear combination of a vector in the image of B and a vector in the kernel of A. This is precisely condition (3.15) which is what we wanted to show. □

Corollary 3.3.2. *Under the hypothesis of Theorem 3.3.1, if condition* (3.15) *holds at some* $x^\star = \operatorname{col}(x_c^\star, x_v^\star) \in \mathbb{R}^n$ *with* $h(x_c^\star) \in \gamma$ *then, there exists a neighbourhood* U *of* $x^\star$ *such that in* U *the connected component of the path following manifold containing* $x^\star$ *is given by*

$$\Gamma^\star \cap U = \{x \in U : \lambda(x_c) = L_f \lambda(x) = 0\} \tag{3.17}$$

where $\lambda(x_c) = s \circ h$.

[5]Two vector subspaces $\mathscr{R}$ and $\mathscr{S}$ are independent if $\mathscr{R} \cap \mathscr{S} = \{0\}$.

Proof. We will find a set $\Gamma^\star$ that satisfies Definition 3.1.1. By Theorem 3.3.1, $\lambda = s \circ h$ yields a well-defined vector relative degree of $\{2, \ldots, 2\}$. In particular we have

$$\begin{aligned}\dot{\hat{y}} &= L_f\lambda(x)\\ \ddot{\hat{y}} &= L_f^2\lambda(x) + L_gL_f\lambda(x)\tau.\end{aligned}$$

where the $(p-1)\times m$ decoupling matrix $L_gL_f\lambda(x)$ is full rank at $x^\star$. This implies that $L_gL_f\lambda(x)$ has some $p-1\times p-1$ minor with a non-zero determinant at $x^\star$. Since the determinant is a continuous function of the entries of $L_gL_f\lambda(x)$, it follows that this determinant is non-zero in a neighbourhood U of $x^\star$.

This implies that the equation

$$0 = L_f^2\lambda(x) + L_gL_f\lambda(x)\tau$$

is solvable in U for some smooth state feedback $\tau^\star(x)$. Therefore the the set

$$\left\{x \in U : \hat{y} = \dot{\hat{y}} = 0\right\} = \{x \in U : \lambda(x_c) = L_f\lambda(x) = 0\}$$

can be made controlled invariant, i.e., if we chose an initial condition on this set and apply the control $\tau^\star$, the mechanical system will remain on this set. Thus we have a controlled-invariant set contained in the lift of the path. Furthermore, since there is no control input appearing in the $\dot{\hat{y}}$ equation, this set is maximal. □

Combining the above results we reach our main conclusion.

Corollary 3.3.3. *Given a mechanical system* (3.13) *and a path γ in the output space satisfying Assumptions 1 and 2. The system is locally transversely feedback linearizable with respect to the path following manifold* (3.17) *if there exists a point $x_c^\star \in \mathbb{R}^N$ such that*

$$\operatorname{Im}\left(\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)\right) + \ker\left(\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c^\star)}\right) \simeq \mathbb{R}^p.$$

Under the conditions of the above corollary, the functions $\lambda_1(x_c) = s_1(h(x_c)), \ldots, \lambda_{p-1}(x_c) = s_{p-1}(h(x_c))$ satisfy the hypothesis of Theorem 3.1.1.

The condition (3.15) may be intuitively explained as follows. The 1-dimensional path exists in $\mathbb{R}^p$. The matrix $\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)$ maps controls to $\mathbb{R}^p$. Motions tangent to the path lie in the kernel of $\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c^\star)}$. Such motions are not transversal to the path following manifold, $\Gamma^\star$. Hence, any control mapped into $\ker\left(\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c^\star)}\right)$ will cause a motion tangential to the path following manifold. Therefore, to ensure enough transverse control will

"appear" in the output space, it is necessary that $\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)$ maps controls to $\mathbb{R}^p$, not including $\ker\left(\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c^\star)}\right)$, which is equivalent to (3.15). As an aid to visualize the condition (3.15), Figure 3.2 illustrates the relationships of the various linear mappings.

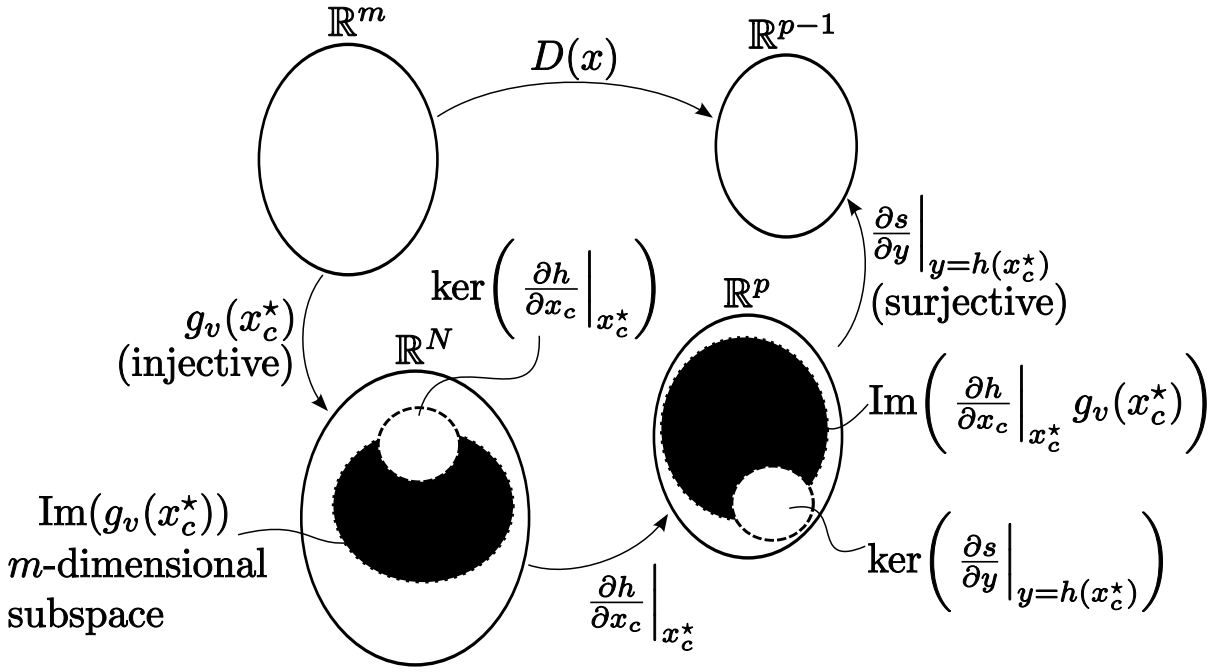


Figure 3.2: Relationship of compositions used to visualize proof of Theorem 3.3.1

The main result is notable as it suggests a systematic approach to applying transverse feedback linearization to solve the path following problem for mechanical systems. Using the constraint function, $\lambda(x_c) = s(h(x_c))$, by Corollary 3.3.3, checking whether transverse feedback linearization is possible boils down to checking the rank of a matrix, namely checking whether

$$\left[\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star) \quad \middle| \quad \kappa(x_c^\star)\right]$$

has rank p, where $\kappa(x_c^\star)$ spans $\ker\left(\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c^\star)}\right)$.

Corollary 3.3.3 provides easily checkable sufficient conditions given a particular choice $\lambda(x_c) = s \circ h(x_c)$ to satisfy Theorem 3.1.1. We emphasize that Theorem 3.3.1 is a local result; it is valid only in a neighbourhood of a point on the path following manifold.

Remark 3.3.1. Notice that the matrix

$$\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)$$

has dimensions $p \times m$, and recall that $m \geq p-1$. In the case where $m > p-1$, this matrix may have full rank p along the path, in which case the condition (3.15) is satisfied for *any* embedded path. Intuitively, this describes the situation where the control can induce a motion in any direction in the output space. Therefore a sufficient condition for (3.15) to hold, is for $\text{rank}\left(\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)\right) = p$. The remaining case is when $m = p-1$. In this case, it is impossible for the rank of $\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)$ to be p, which means that (3.15) depends on the shape of the embedded path. Intuitively speaking, for this condition to hold, all of the control must be able to drive the state transversally toward the path following manifold (which implies that there is no control that can drive the output tangentially along the path).

Remark 3.3.2. Theorem 3.3.1 provides necessary and sufficient conditions for a mechanical system with a virtual output to yield a well-defined vector relative degree of $\{2, \ldots, 2\}$. Relative degree is how many times one particular output must be differentiated before the control input, τ, appears non-singularly. It makes intuitive sense for systems governed by Newton's laws of motion to have a vector relative degree of $\{2, \ldots, 2\}$, since the virtual output is a function of generalized *position*, and τ is a generalized force, affecting *acceleration*. Naturally differentiating position twice yields acceleration.

Remark 3.3.3. One of our underlying assumptions has been that the path, $\gamma \in \mathbb{R}^p$, is a 1-dimensional set. The results of this section can be naturally extended to consider higher dimensional "paths" (though to avoid confusing terminology, we will call the higher dimensional paths "output sets"). Suppose we have an output set with dimension $\bar{c} > 1$. Generalizing our main results requires modification to the theorems and corollaries. For example

- Assumptions 1 and 2 must be modified such that the output set γ is an embedded submanifold of $\mathbb{R}^p$ and can be written as the zero level set of some function $s : \mathbb{R}^p \to \mathbb{R}^{p-\bar{c}}$ so that $\gamma = \{y \in \mathbb{R}^p : s(y) = 0\}$.
- We conjecture that condition (3.15) of Theorem 3.3.1 will provide necessary and sufficient conditions for the mechanical system with virtual output to yield a well-defined vector relative degree of $\{2, \ldots, 2\}$ for such higher dimensional output sets.
- We must generalize the notion of a path following manifold to the maximal controlled-invariant subset of Γ (this is more of an issue of terminology than anything else)

- The basic feasibility requirement generalizes to $m \geq p - \bar{c}$

We omit mathematical rigour in extending the dimension of the set in output space as it is outside the scope of the subsequent chapters. However, we comment that this natural extension, given Theorem 3.3.1 and Corollaries 3.3.2 and 3.3.3, is a good candidate for future research, as suggested in Chapter 6.

Remark 3.3.4. In Theorem 3.3.1 the only restriction on the degree of underactuation of the mechanical control system is that $m \geq p - 1$. Therefore we allow for underactuated systems. When $m > p - 1$ we will show in Section 3.3.2 that these "extra" controls *may* be effectively used to simplify path following control design.

Remark 3.3.5. In Section 3.1, **Steps 1-4** are used to solve the path following problem for more general systems. **Step 1** is finding the path following manifold, $\Gamma^\star$, so that in **Step 2** we may find l functions λ such that $\Gamma^\star \subset \{x : \lambda_1(x) = \cdots = \lambda_l(x) = 0\}$ (Condition (a) of Theorem 3.1.1), which may be difficult. In our approach, **Step 1** and condition (a) of Theorem 3.1.1 are obtained "for free", as our choice of virtual output satisfies $\Gamma^\star \subset \{x : s \circ h(x_c) = 0\}$ by definition. Therefore, Corollary 3.3.3 is analogous to condition (b) in Theorem 3.1.1. **Step 3** entails stabilizing an LTI system, and **Step 4** is the topic of Section 3.3.2.

3.3.1 Illustrative Examples

We return to the fictitious simple mechanical system of Example 3.1.1 to illustrate the main results of this thesis. Three successive examples are provided, with the objective of associating intuition with the theoretical results.

Example 3.3.1. Recall the magnetized puck of mass M on a frictionless plane, shown in Figure 3.3. Again assume the electromagnets may apply the orthogonal forces τ_1 and τ_2.

The state of this system consists of the positions and velocities of the puck in each orthogonal direction so that $x = \text{col}(x_{c_1}, x_{c_2}, x_{v_1}, x_{v_2})$. The system equations are

$$\begin{aligned} \dot{x}_c &= x_v \\ \dot{x}_v &= f_v(x) + g_v(x_c)\tau \end{aligned}$$

which belongs to the class of systems, (3.10), where $f_v(x) = 0_{2\times 1}$, $g_v(x_c) = \frac{1}{M}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, and $\tau = \text{col}(\tau_1, \tau_2)$. Our output, $y = \text{col}(y_1, y_2) = h(x_c)$, is the position of the mass in the plane, such that

$$\begin{aligned} y_1 &= x_1 \\ y_2 &= x_2. \end{aligned}$$

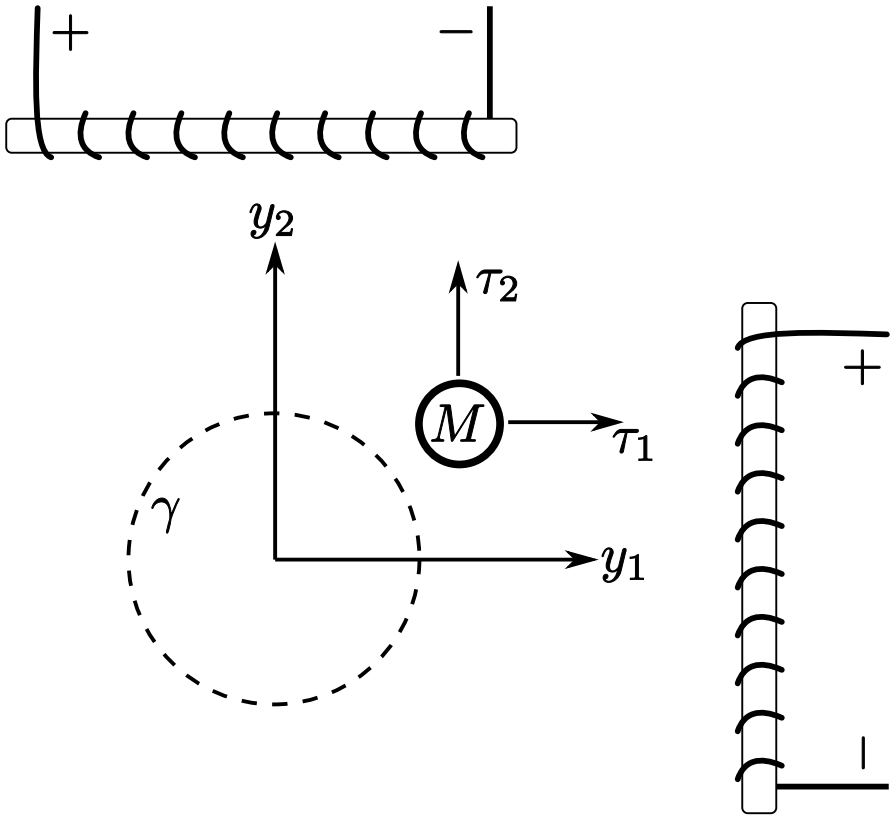


Figure 3.3: A simple mechanical control system revisited

Suppose the path in output space is the unit circle, i.e. $\gamma = \{y \in \mathbb{R}^2 : s(y) = y_1^2 + y_2^2 - 1 = 0\}$. Checking condition (3.15) is equivalent to checking if the rank of the matrix

$$\left[\frac{\partial h}{\partial x_c} g_v(x_c) \quad \middle| \quad \kappa(x_c) \right] = p$$

along the path, where $\kappa(x_c)$ spans the kernel of $\frac{\partial s}{\partial y}|_{y=h(x_c)}$. However, we find that

$$\frac{\partial h}{\partial x_c} g_v(x_c) = \frac{1}{M} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad M > 0,$$

which obviously has rank $p = 2$. Therefore, this system is transversely feedback linearizable *for any* path satisfying Assumptions 1 and 2, which intuitively results from the fact that the controls induce orthogonal motions in the output space.

△

We now consider a hypothetical situation: one of the electromagnets is broken. Our system still possesses the minimum number of controls, i.e. $m \geq p - 1$, where $p = 2$, and $m = 1$. We will stick with the circular path and see what happens...

Example 3.3.2. Reconsider the puck of mass M on a frictionless surface from Example 3.3.1. Suppose that the electromagnet associated with τ_2 in the previous example is no longer operational, as illustrated in Figure 3.4. The state of this underactuated system

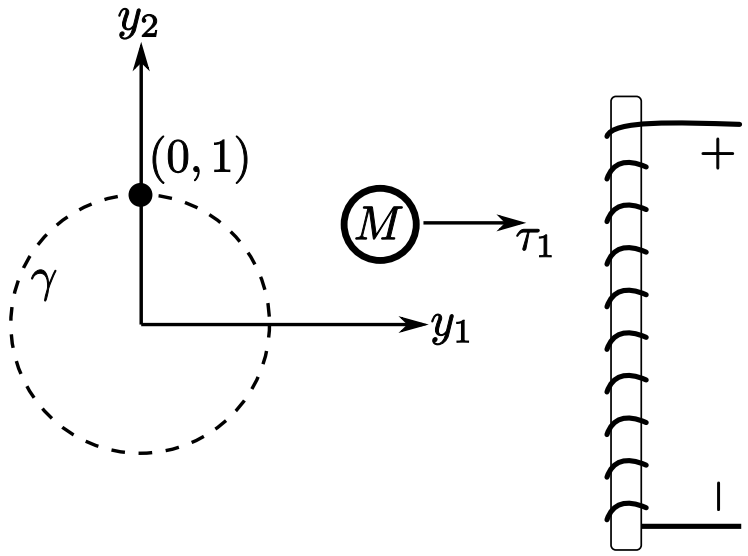


Figure 3.4: An illustrative underactuated mechanical system

remains $x = \text{col}(x_{c_1}, x_{c_2}, x_{v_1}, x_{v_2})$. The system equations are

$$\begin{aligned} \dot{x}_c &= x_v \\ \dot{x}_v &= f_v(x) + g_v(x_c)\tau_1, \end{aligned}$$

where $f_v(x) = 0_{2\times 1}$ and $g_v(x_c) = \frac{1}{M}\begin{bmatrix} 1 \\ 0 \end{bmatrix}$. The output remains the position $y = \text{col}(y_1, y_2) = h(x_c) = \text{col}(x_{c_1}, x_{c_2})$, with the circular path $\gamma = \{y \in \mathbb{R}^2 : s(y) = y_1^2 + y_2^2 - 1 = 0\}$, whose Jacobian is $\frac{\partial s}{\partial y} = [2y_1 \quad 2y_2]$. The kernel of $\frac{\partial s}{\partial y}$ is spanned by $[y_2 \quad -y_1]^T$.

Checking condition (3.15) is equivalent to checking if the rank of the matrix

$$\left[\frac{\partial h}{\partial x_c} g_v(x_c) \quad \middle| \quad \kappa(x_c)\right] = p$$

along the path, where $\kappa(x_c)$ spans the kernel of $\frac{\partial s}{\partial y}|_{y=h(x_c)}$. In this case,

$$\frac{\partial h}{\partial x_c} g_v(x_c) = \frac{1}{M} \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad M > 0$$

cannot have rank p. This means, intuitively speaking, that the shape of the path will determine whether this underactuated system is transversely feedback linearizable. This is true in general of systems where $m = p - 1$. The rank of

$$\left[\frac{\partial h}{\partial x_c} g_v(x_c) \quad \middle| \quad \kappa(x_c) \right] = \begin{bmatrix} \frac{1}{M} & x_{c_2} \\ 0 & -x_{c_1} \end{bmatrix}$$

along the circular path is equal to 2 *except* where $x_{c_1} = 0$. Therefore, at the point on the unit circle, $(0, 1)$, as shown in Figure 3.4, our control becomes *tangential*, rather than *transversal*, and cannot drive y onto γ (which is also true for the point $(0, -1)$).

Therefore we can *locally* transversely feedback linearize the mechanical system given the circular output at any point along the path except the "poles".

△

In the previous example, Example 3.3.2, we have shown that when $m = p-1$, the shape of the embedded path necessarily determines whether or not a simple mechanical system is transversely feedback linearizable. We have also linked the failure of condition (3.15) to the intuitive scenario where locally no control is able to drive the state transversally toward the path following manifold. In the next example, we keep the same underactuated system, but change the path, and see what happens...

Example 3.3.3. Reconsider the system and output of Example 3.3.2. Suppose that the path is no longer circular, but linear, as illustrated in Figure 3.4. The path is the line $\gamma = \{y \in \mathbb{R}^2 : s(y) = y_1 - y_2 = 0\}$, whose Jacobian is $\frac{\partial s}{\partial y} = [1 \quad -1]$. The kernel of $\frac{\partial s}{\partial y}$ is spanned by $[1 \quad 1]^T$.

Checking condition (3.15) is equivalent to checking if the rank of the matrix

$$\left[\frac{\partial h}{\partial x_c} g_v(x_c) \quad \middle| \quad \kappa(x_c) \right] = p$$

along the path, where $\kappa(x_c)$ spans the kernel of $\frac{\partial s}{\partial y}|_{y=h(x_c)}$. The rank of

$$\left[\frac{\partial h}{\partial x_c} g_v(x_c) \quad \middle| \quad \kappa(x_c) \right] = \begin{bmatrix} \frac{1}{M} & 1 \\ 0 & 1 \end{bmatrix}$$

along the linear path is equal to 2. Therefore, this underactuated system is transversely feedback linearizable given the particular path.

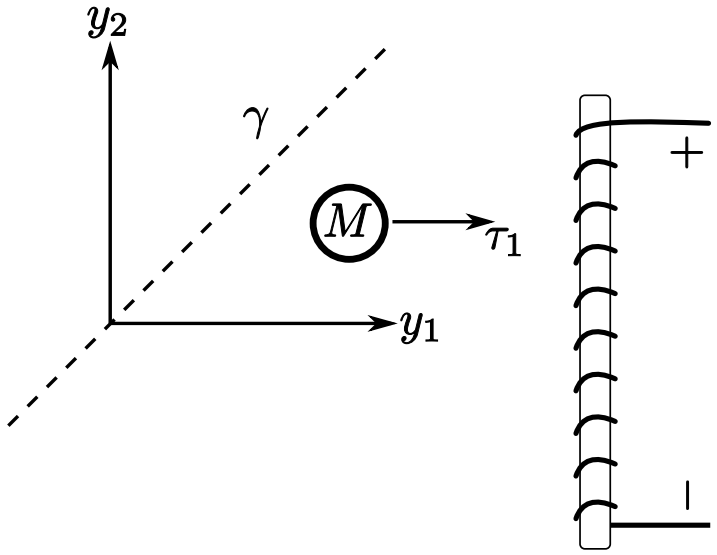


Figure 3.5: Another illustrative underactuated mechanical system

△

In Example 3.3.1 we considered a system with $m = p$, and showed that that particular system was transversely feedback linearizable for *any* path[6] satisfying Assumptions 1 and 2. In Example 3.3.2, losing an actuator so that $m = p-1$, we showed that condition (3.15) fails at some points on the circular path, since the only control becomes "tangential" at those points. By changing the path to a line, Example 3.3.3 reveals that condition (3.15) does indeed hold everywhere along the path.

3.3.2 Partial Feedback Linearization of Tangential Dynamics

We have presented conditions for partially feedback linearizing the dynamics of a mechanical system that are transverse to the path following manifold. Stabilizing the transverse dynamics will cause the output of the system to approach the desired path. Suppose that in doing this, we have not "used up" all of the available control. That is, suppose that m is strictly greater than $p - 1$. We will show that a tangential control, $v^{\parallel}$, may be used to achieve desired motions of the output along the path.

The objective of this section is to leverage the structure of simple mechanical systems in order to find a coordinate and feedback transformation such that the η-subsystem is

[6]For practical purposes, many mechanical systems satisfy $m = p$. It is the author's conjecture that *most often* when $m = p$, the practical mechanical system is transversely feedback linearizable independent of the embedded path.

partially linear, time-invariant, and controllable (in addition to transverse feedback linearization). Recall that the η, or tangential[7], dynamics govern motions of the output which are not transverse to the path. We seek to further decompose the tangential subsystem by identifying that part of the tangential dynamics that governs motion along the path in the output space. In this section we show that it is possible to feedback linearize that portion of the tangential dynamics that corresponds to motion in the output space.

Recall the normal form of the tangential dynamics from Section 3.1,

$$\dot{\eta} = f^0(\eta, \xi) + g^{\pitchfork}(\eta, \xi)v^{\pitchfork} + g^{\parallel}(\eta, \xi)v^{\parallel}.$$

We seek to further decompose these dynamics so that they appear as

$$\begin{aligned} \dot{\eta}_1 &= f^0(\eta_1, \eta_2, \xi) + g^{\pitchfork}(\eta_1, \eta_2, \xi)v^{\pitchfork} + g_1^{\parallel}(\eta_1, \eta_2, \xi)v_1^{\parallel} + g_2^{\parallel}(\eta_1, \eta_2, \xi)v_2^{\parallel} \\ \dot{\eta}_2 &= A^{\parallel}\eta_2 + B^{\parallel}v_2^{\parallel}, \end{aligned} \tag{3.18}$$

where $\dim(\eta_1) = n^\star - 2$, $\dim(\eta_2) = 2$, the pair $(A^{\parallel}, B^{\parallel})$ is controllable and $v^{\parallel} = (v_1^{\parallel}, v_2^{\parallel}) \in \mathbb{R}^{m-p} \times \mathbb{R}$. Therefore, our objective is to obtain the overall normal form

$$\begin{aligned} \dot{\eta}_1 &= f^0(\eta_1, \eta_2, \xi) + g^{\pitchfork}(\eta_1, \eta_2, \xi)v^{\pitchfork} + g_1^{\parallel}(\eta_1, \eta_2, \xi)v_1^{\parallel} + g_2^{\parallel}(\eta_1, \eta_2, \xi)v_2^{\parallel} \\ \dot{\eta}_2 &= A^{\parallel}\eta_2 + B^{\parallel}v_2^{\parallel} \\ \dot{\xi} &= A\xi + Bv^{\pitchfork}, \end{aligned}$$

A basic feasibility requirement to obtain (3.18) is that $m > p - 1$. We will show that physically, the η_1-subsystem represents the dynamics on the path following manifold that are redundant in the following sense. While changes to the η_1 states cannot cause the system to leave the path following manifold, and thereby leave the desired path, they also do not help to propel the system along the path in the output space. Uncontrollable tangential dynamics will also appear in the η_1 dynamics[8]. On the other hand η_2 represents the output position on the path. In other words, changes in the state η_2 will cause observable motion along the path in the output space.

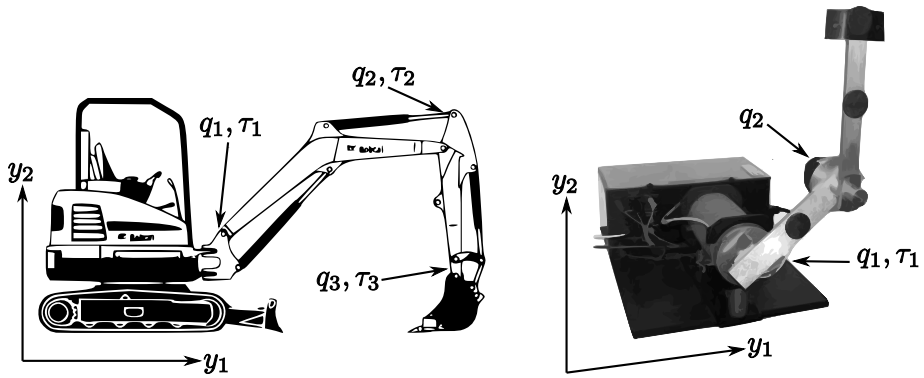
To help understand the relationship between the number of controls, m, the dimension of the output space, p, and the tangential normal form (3.18), consider Figure 3.6. Suppose we have a fully actuated 3-link planar manipulator (i.e. $p = 2, m = 3$), such as the excavator in Figure (3.6a), and we want the tip of the excavator bucket to trace a given path. In this case $p - 1 = 1$ and if we are able to put the system into the normal form (3.2) we will have $v = \mathrm{col}(v^{\pitchfork}, v^{\parallel}) \in \mathbb{R}^1 \times \mathbb{R}^2$. There are 2 tangential controls, such that $v^{\parallel} := \mathrm{col}(v_1^{\parallel}, v_2^{\parallel})$. However, since the path is one dimensional, there is only one degree

[7]We refer to η dynamics as "tangential"; however, this is not to be confused with the *tangential subsystem*, which is defined in Section 3.1 as the η dynamics constrained to the path following manifold.

[8]The reader may skip ahead to Example 3.3.5 where these notions are illustrated.

of freedom required to move the tip of the bucket along the path. Thus, in reference to (3.18), $v_2^{\parallel}$ will create observable motions of the tip of the excavator along the path, and $v_1^{\parallel}$ will create unobservable tangential motions associated with the joint configurations of this kinematically redundant system.

Conversely, suppose we have a 2-link planar pendulum with only the base joint actuated ($p = 2, m = 1$), as shown in Figure 3.6b so that $m = p - 1$. In this case, according to Corollary 3.3.3, all of the control must go toward stabilizing the transversal dynamics, leaving no control inputs left to affect any of the tangential dynamics. This demonstrates a violation of the basic feasibility of $m > p - 1$.



(a) Excavator

(b) Underactuated double pendulum

Figure 3.6: Interesting examples of kinematically redundant and underactuated mechanical systems

Remark 3.3.6. In this thesis we focus on paths, i.e., 1-dimensional sets in the output space. However, it is a natural extension to consider higher dimensional sets in the output space. Suppose we have a $\bar{c}$ dimensional set in the output space (for example, $\bar{c} = 2$ for a sphere). With respect to obtaining the normal form (3.18), the two basic feasibility requirements are

- $m > p - 1$, if the goal is to partially linearize *all* of the tangential dynamics corresponding to observable motions in the output space.
- $m > p - \bar{c}$, if the goal is to partially linearize *any* of the tangential dynamics corresponding to observable motions in the output space (i.e., $\dim(\eta_2) > 0$).

Assume that we are interested in partially linearizing as many of the tangential dynamics corresponding to observable output motions as possible, though not necessarily all of them.

The normal form (3.18) would change as follows. The ξ-subsystem would be $2(p-\bar{c})$ dimensional, with $\dim\left(v^{\pitchfork}\right) = p - \bar{c}$. We must introduce the notion of "c", the number of tangential controls which may cause observable tangential output motions. The η_2-subsystem would be $2c$ dimensional, with $\dim\left(v_2^{\parallel}\right) = c$, where

$$c \leq \min(m-p, 0) + \bar{c}$$

represents the number of controls which may cause observable tangential output motions. The number $\min(m-p,0)+\bar{c}$ corresponds to the number of tangential controls which *may* cause observable output motions. The number of tangential controls which *do* cause observable output motions must be less than or equal to $\min(m-p,0)+\bar{c}$, as not all tangential controls cause observable output motions (see Example 3.3.5). Finally the η_1-subsystem would be $n^\star - 2c$ dimensional.

3.3.3 Path Parametrization and Tubular Neighbourhoods

Our ultimate goal is to show that if we use the arc length of the path as one of our tangential states, we can partially feedback linearize the tangential dynamics. Let $\sigma : \mathbb{D} \to \mathbb{R}^p$ be a curve that satisfies Assumptions 1 and 2. Since σ is regular by assumption, without loss of generality we can assume that it has a unit speed parametrization, i.e.,

$$\|\sigma'(\cdot)\| = 1.$$

Under this assumption, the curve σ is parameterized by its arc length [39]. For closed-curves with finite length L, this means that σ is L-periodic, i.e., for any $\theta \in \mathbb{D}$, $\sigma(\theta + L) = \sigma(\theta)$. Thus in the case of closed curves we will treat σ as a map

$$\begin{aligned} \sigma : [0, L] &\to \mathbb{R}^p \\ \theta &\mapsto \sigma(\theta) \end{aligned}$$

where $\sigma([0, L]) = \gamma$. Note that $\theta \in [0, L)$ *uniquely* determines a position on the path.

We need a similar representation in the case of non-closed curves $\sigma : \mathbb{R} \to \mathbb{R}^p$. The problem is that non-closed curves that satisfy Assumption 1 do not have finite length so we will need to work with a piece of the curve. Once again we assume that σ is parameterized by its arc length.

Let $x_c^\star \in \mathbb{R}^N$ be such that $h(x_c^\star) \in \gamma$. Let V be a neighbourhood of $h(x_c^\star)$ such that $\gamma \cap V$ contains a single connected component of the path. Since γ is an embedded submanifold, such a V is guaranteed to exist. Let $\bar{V}$ be the closure of V. Then the portion of the path

$\gamma \cap \bar{V}$ has finite length L and if we denote the boundary of $\bar{V}$ by $\partial\bar{V}$, there exist two real numbers $\theta_1 < \theta_2$ such that $\theta_2 = \theta_1 + L$ and

$$\partial\bar{V} \cap \gamma = \{\sigma(\theta_1), \sigma(\theta_2)\}.$$

Using this fact we define a new map $\tilde{\sigma} : [0, L] \to \bar{V}$ as

$$\tilde{\sigma}(\theta) = \sigma(\theta + \theta_1).$$

To ease notation, we will drop the $\tilde{\cdot}$ notation with the understanding that the above construction has taken place. In conclusion, whether γ is closed or not we can find a map

$$\begin{aligned} \sigma : \mathcal{L} &\to \mathbb{R}^p \\ \theta &\mapsto \sigma(\theta) \end{aligned} \tag{3.19}$$

with $\mathcal{L} = [0, L]$ and with $\sigma(\mathcal{L}) = \gamma$ in the case of closed-curves and $\sigma(\mathcal{L}) = \bar{V} \cap \gamma$ in the non-closed case.

Before stating the main result of this section, we first define a map that associates to each point y in the output space of system (3.13) sufficiently close to the path γ (or $\bar{V} \cap \gamma$ if the curve is unbounded) a number between $[0, L]$. To make this idea precise we introduce the notion of a tubular neighbourhood of the curve γ. We denote the tubular neighbourhood of γ as $\gamma_\epsilon \subset \mathbb{R}^p$. Intuitively, the tubular neighbourhood of γ is an open subset of $\mathbb{R}^p$ that contains γ. It has the property that if $y \in \gamma_\epsilon$ then there exists a unique $y^\star \in \gamma$ that is closest to y. By the tubular neighbourhood theorem [21, Chapter 2], we know that such a neighbourhood exists.

3.3.4 Partial Feedback Linearization of Tangential Subsystem

Using the idea of tubular neighbourhoods we introduce the following projection operator which maps $y \in \gamma_\epsilon$ to a unique $\theta \in [0, L]$ such that the point $\sigma(\theta) \in \gamma$ is closest to y:

$$\begin{aligned} \varpi :&\gamma_\epsilon \to \mathcal{L} \\ &\varpi(y) = \arg\min_{\theta \in \mathcal{L}} \|y - \sigma(\theta)\|. \end{aligned} \tag{3.20}$$

Figure 3.7 illustrates the proposed projection and parametrization in the case of a closed curve in the plane. The next lemma, based on [14, Lemma 3.2], gives some geometric insight into the nature of the map ϖ.

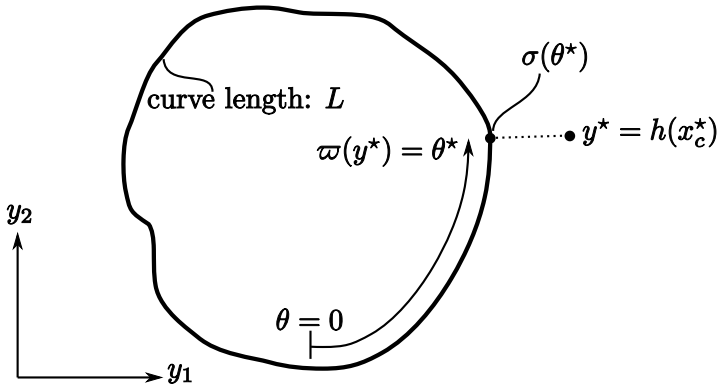


Figure 3.7: Illustration of projection and parameterization of a point $y^\star$ in the output space onto a closed 1-dimensional path

Lemma 3.3.4. *For all $y \in \gamma$ the vectors $d\varpi_y$ and ds_y are orthogonal.*

Proof. By the definition of ϖ, for any $\theta \in \mathbb{D}$ and any $y \in \varpi^{-1}(\theta)$, we have $\langle \sigma'(\theta),\ y - \sigma(\theta) \rangle = 0$. Since $\sigma'(\theta)$ is the tangent vector to γ at $\sigma(\theta)$, this shows that the set $\{y \in \gamma_\epsilon : \varpi(y) = \theta\}$ is a straight line segment perpendicular to γ. Hence the normal vector to this segment at $\sigma(\theta)$, $d\varpi_{\sigma(\theta)}$, is tangent to γ. Hence for any $\theta \in \mathbb{D}$, $(d\varpi_{\sigma(\theta)})^T = k\sigma'(\theta)$ for some $k \in \mathbb{R}$. In other words, the gradient of the function ϖ is tangent to the path when evaluated on the path. We now show that $k \neq 0$. Consider the identity $\varpi(\sigma(\theta)) = \theta$. Differentiating this identity respect to θ we get $d\varpi_{\sigma(\theta)}\sigma'(\theta) = 1$ which shows that $k \neq 0$.

Next we recall that on the set we have the identity $s \circ \sigma(\theta) \equiv 0$. Differentiating this identity, we conclude that $ds_{\sigma(\theta)}\sigma'(\theta) = 0$. We can combine these facts to determine that $ds_{\sigma(\theta)} \left(d\varpi_{\sigma(\theta)}\right)^T = 0$. □

Lemma 3.3.4 shows that the matrix $\left[\begin{array}{cc} ds_y & d\varpi_y \end{array}\right]^T_{y=\sigma(\theta)}$ is orthogonal and therefore full rank. We now present the main result of this section.

Theorem 3.3.5. *Given a mechanical control system (3.13) with $m > p - 1$ and a path γ in output space satisfying Assumptions 1 and 2, let $x^\star = (x_c^\star, x_v^\star) \in \mathbb{R}^n$ satisfy $h(x_c^\star) \in \gamma$. The system*

$$\begin{aligned} \dot{x}_c &= x_v \\ \dot{x}_v &= f_v(x) + g_v(x_c)\tau \\ \hat{y} &= \left[\begin{array}{c} \lambda(x_c) \\ \pi(x_c) \end{array}\right] = \left[\begin{array}{c} s \circ h(x_c) \\ \varpi \circ h(x_c) \end{array}\right], \end{aligned} \tag{3.21}$$

yields a well-defined vector relative degree of $\{2,\dots,2\}$ *at* $x^\star$ *if and only if* (3.15) *holds and*

$$\dim\left(\mathrm{Im}\left(\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)\right)\cap\ker\left(\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c^\star)}\right)\right)=1. \tag{3.22}$$

holds at $x^\star$*.*

In Theorem 3.3.1 we showed that the virtual output $s\circ h(x_c)$ yields a well-defined relative degree under the hypotheses (3.15). In Theorem 3.3.5 we augment the virtual output with the projection operator

$$\begin{aligned}\pi&:\mathbb{R}^N\to\mathcal{L}\\ \pi(x_c)&=\varpi\circ h(x_c).\end{aligned} \tag{3.23}$$

Theorem 3.3.5 states that if we have enough actuation, at least one control will appear in the tangential dynamics, and that by satisfying condition (3.22), it is possible to choose the output, $\pi(x_c)$, such that the tangential dynamics are partially feedback linearizable and the tangential dynamics of the transformed system have the form (3.18) in a neighbourhood of a point $x^\star$.

Proof. Let $x^\star=(x_c^\star,x_v^\star)$ be any point that satisfies[9] $h(x_c^\star)\in\gamma$. Assume that (3.15) and (3.22) hold. We will show that the virtual output yields a well-defined vector relative degree of $\{2,\dots,2\}$ at $x^\star$. This is true if and only if the $p\times m$ decoupling matrix

$$\begin{bmatrix}L_gL_f\lambda(x^\star)\\ L_gL_f\pi(x^\star)\end{bmatrix}=\begin{bmatrix}ds_{h(x_c^\star)}\circ dh_{x_c^\star}g_v(x_c^\star)\\ d\varpi_{h(x_c^\star)}\circ dh_{x_c^\star}g_v(x_c^\star)\end{bmatrix}=\begin{bmatrix}ds_{h(x_c^\star)}\\ d\varpi_{h(x_c^\star)}\end{bmatrix}dh_{x_c^\star}\,g_v(x_c^\star) \tag{3.24}$$

is full rank at $x^\star$. By Lemma 3.3.4 the matrix $\left[(ds_{h(x_c^\star)})^\top(d\varpi_{h(x_c^\star)})^\top\right]^\top$ is non-singular. Therefore the decoupling matrix will be full rank at $x^\star$ if the $p\times m$ matrix $dh_{x_c^\star}\,g_v(x_c^\star)$ is surjective, i.e., full rank.

To show that $dh_{x_c^\star}\,g_v(x_c^\star)$ is surjective, we note that by (3.15)

$$p=\dim\left(\mathrm{Im}\left(\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)\right)+\ker\left(\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c^\star)}\right)\right)$$

which implies

$$\begin{aligned}\dim\left(\mathrm{Im}\left(\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)\right)\right)=p-\dim\left(\ker\left(\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c^\star)}\right)\right)+\\ \dim\left(\mathrm{Im}\left(\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)\right)\cap\ker\left(\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c^\star)}\right)\right).\end{aligned}$$

[9]Note that if σ is not closed we require that $h(x_c^\star)\in\gamma\cap V$ where V is described in Section 3.3.3

By Assumption 2, the $(p-1)\times p$ matrix $\frac{\partial s}{\partial y}$ is full rank on γ and hence

$$\dim\left(\mathrm{Im}\left(\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)\right)\right) = p-1+\dim\left(\mathrm{Im}\left(\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)\right)\cap\ker\left(\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c^\star)}\right)\right).$$

However, since (3.22) holds, we conclude that

$$\dim\left(\mathrm{Im}\left(\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)\right)\right) = p$$

and we have shown that $dh_{x_c^\star} g_v(x_c^\star)$ is surjective as required.

Conversely, suppose that output function of (3.21) yields a well-defined vector relative degree of $\{2,\ldots,2\}$ at some $x^\star = (x_c^\star, x_v^\star) \in \mathbb{R}^n$ with $h(x_c^\star)\in\gamma$. This means that the decoupling matrix (3.24) is full rank at $x^\star$. By Lemma 3.3.4 this implies that $dh_{x_c^\star} g_v(x_c^\star)$ is full rank (surjective). Thus conditions (3.15) and (3.22) hold trivially.

□

In reference to Figure 3.2, motions in the output space along the path (i.e., tangent to the path) lie in the kernel of $\frac{\partial s}{\partial y}$. Intuitively, condition (3.22) means that, in a given configuration, control must be mapped into this kernel so that motions along the curve are feedback linearizable. Recall that for transverse feedback linearization, Theorem 3.3.1 required control to be mapped into all of $\mathbb{R}^p$, with the exception of the kernel of $\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c)}$, where Theorem 3.3.5 requires that the conditions of Theorem 3.3.1 hold *and* that control is mapped into

$$\ker\left(\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c)}\right).$$

These facts lead to the following corollary.

Corollary 3.3.6. *Given a mechanical system* (3.13)*, and a path γ satisfying Assumptions 1 and 2, the system is locally transversely feedback linearizable with respect to* (3.17)*, and the tangential dynamics are partially feedback linearizable, yielding the normal form*

$$\begin{aligned}\dot\eta_1 &= f^0(\eta_1,\eta_2,\xi) + g^\pitchfork(\eta_1,\eta_2,\xi)v^\pitchfork + g_1^\parallel(\eta_1,\eta_2,\xi)v_1^\parallel + g_2^\parallel(\eta_1,\eta_2,\xi)v_2^\parallel\\ \dot\eta_2 &= A^\parallel\eta_2 + B^\parallel v_2^\parallel\\ \dot\xi &= A\xi + Bv^\pitchfork.\end{aligned} \tag{3.25}$$

at a point $x^\star = \mathrm{col}(x_c^\star, x_v^\star)\in\mathbb{R}^n$ if

$$\mathrm{Im}\left(\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)\right) \simeq \mathbb{R}^p. \tag{3.26}$$

Given a system of the form (3.13), and a path satisfying Corollary 3.3.6, we partially define our coordinate transformation as $\eta_2 = (\pi(x_c), L_f\pi(x))$ and ξ in the usual way. After completing the coordinate transformation with $n^\star - 2$ additional functions

$$\eta_1 := \varphi(x) = \mathrm{col}(\varphi_1(x), \ldots, \varphi_{n^\star-2}(x)),$$

we define a feedback transformation in a similar manner to (3.5) to get (3.25).

Since the state η_2 corresponds to the arc length of the target path, and since the pair $(A^{\|}, B^{\|})$ is controllable, we can effectively design $v_2^{\|}$ in order to accomplish the design specifications on the path and hence complete **Step 4** of the path following design procedure introduced in Section 3.1.

Remark 3.3.7. If a system satisfies Corollary 3.3.6 and has $m = p = N$ (i.e. the system is fully actuated, and the number of degrees of freedom equals the dimension of the output), then our procedure yields a fully linear system. In this case, the η_1-subsystem does not exist.

Remark 3.3.8. We conjecture that it is possible to generalize Theorem 3.3.5 and Corollary 3.3.6 to consider $\bar{c}$-dimensional output sets, where $\bar{c} \geq 1$, as follows. First, at a point on the output set, we find c, the dimension of the tangential control which may cause observable motions of the output along the path, where

$$c = \dim\left(\mathrm{Im}\left(\left.\frac{\partial h}{\partial x_c}\right|_{x_c=x_c^\star} g_v(x_c^\star)\right) \cap \ker\left(\left.\frac{\partial s}{\partial y}\right|_{y=h(x_c^\star)}\right)\right).$$

If $\bar{c} > c > 0$, this means we may partially feedback linearize some, but not all, of the tangential dynamics which cause observable output motions along the output set. If $c = \bar{c}$, this means that we may partially feedback linearize *all* of these dynamics. Since the output set is an embedded submanifold, c cannot be greater than $\bar{c}$.

Our main results rely on augmenting the virtual output with a function $\pi(x_c) = \varpi \circ h(x_c)$, such that the system yields a well-defined vector relative degree. Coming up with such a function for $\bar{c} > 1$ requires more care than in the 1-dimensional case, and will require substantial generalizations of the development in Sections 3.3.3 and 3.3.4.

Example 3.3.4. Recall Example 3.3.1, where a magnetized puck of mass M is on a frictionless flat surface with two orthogonally positioned actuators which apply forces on

the puck, whose model is

$$\begin{aligned}\dot{x}_{c_1} &= x_{v_1}\\ \dot{x}_{c_2} &= x_{v_2}\\ \dot{x}_{v_1} &= \frac{1}{M}\tau_1\\ \dot{x}_{v_2} &= \frac{1}{M}\tau_2,\end{aligned}$$

with output $y = h(x_c) = \mathrm{col}(x_{c_1}, x_{c_2})$. In this example, we have seen that

$$\frac{\partial h}{\partial x_c} g_v(x_c) = \frac{1}{M}\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}, \quad M > 0,$$

has full rank along the path. Therefore by Corollary 3.3.6, this system is transversely feedback linearizable, and the η-subsystem is linearizable for any path satisfying Assumptions 1 and 2. The path

$$\sigma(\theta) = \begin{bmatrix}\cos\theta\\ \sin\theta\end{bmatrix}$$

satisfies these assumptions, and may be expressed as $\gamma = \{y \in \mathbb{R}^p : s(y) = y_1^2 + y_2^2 - 1 = 0\}$. We use the augmented virtual output

$$\hat{y} = \begin{bmatrix}\lambda(x_c)\\ \pi(x_c)\end{bmatrix} = \begin{bmatrix}s \circ h(x_c)\\ \varpi \circ h(x_c)\end{bmatrix},$$

where $\lambda(x_c) = s \circ h(x_c)$ and $\pi(x_c) = \varpi \circ h(x_c)$. The function $\varpi(y)$ describes the arc length of the output projected onto the path. Since the path is a unit circle, we may use the angle of the output with respect to the origin,

$$\varpi(y) = \arg(y_1 + iy_2).$$

Therefore, our augmented virtual output is

$$\hat{y} = \begin{bmatrix}\lambda(x_c)\\ \pi(x_c)\end{bmatrix} = \begin{bmatrix}s \circ h(x_c)\\ \varpi \circ h(x_c)\end{bmatrix} = \begin{bmatrix}x_{c_1}^2 + x_{c_2}^2 - 1\\ \arg(x_{c_1} + ix_{c_2})\end{bmatrix}$$

According to our procedure, the coordinate transformation is $T : x \mapsto (\eta, \xi)$ where

$$T(x) = \mathrm{col}(\pi(x_c), L_f\pi(x), \lambda(x_c), L_f\lambda(x)),$$

and the feedback transformation is

$$\begin{bmatrix}\tau_1\\ \tau_2\end{bmatrix} = \begin{bmatrix}L_gL_f\pi(x)\\ L_gL_f\lambda(x)\end{bmatrix}^{-1}_{x=T^{-1}(\eta,\xi)} \left(\begin{bmatrix}-L_f^2\pi(x)\\ -L_f^2\lambda(x)\end{bmatrix}^{-1}_{x=T^{-1}(\eta,\xi)} + \begin{bmatrix}v_2^{\parallel}\\ v^{\pitchfork}\end{bmatrix} \right)$$

Since $m = N = p$, the coordinate and corresponding feedback transformation yield the fully linear system, with $(\eta, \xi) := \text{col}(\eta_1^2, \eta_2^2, \xi_1, \xi_2)$, such that

$$\begin{aligned}
\dot{\eta}_1^2 &= \eta_2^2 \\
\dot{\eta}_2^2 &= v_2^{\parallel} \\
\dot{\xi}_1 &= \xi_2 \\
\dot{\xi}_2 &= v^{\pitchfork}.
\end{aligned}$$

Therefore, the objective of feedback linearizing the transverse dynamics, and partially linearizing the tangential dynamics corresponding to observable output motions, is achieved. This system is exceptional because the functions $\lambda(x_c)$ and $\pi(x_c)$ are valid over the entire path, which is generally not the case.

$\triangle$

The previous example illustrates transverse feedback linearization, including imposing more structure on the η-subsystem, which ends up as an exact linearization of the simple mechanical system given the path. Notice that the puck's dynamics were already linear; however, they are now linear with respect to a nonlinear path. Recall that a basic feasibility requirement is that $m > p - 1$. Therefore, even though Example 3.3.3 revealed that the mechanical system was transversely feedback linearizable, imposing more structure on the η-subsystem is impossible, since $m = p - 1$. This is intuitively obvious by inspecting Figure 3.5. We now illustrate a similar idea via example.

Example 3.3.5. Recall Example 3.3.3, and suppose that one additional control, τ_2 is added which affects the orientation of the puck, whose mass is M with unit rotational inertia. Now $m = p = 2$, which means we satisfy the basic feasibility requirement for Corollary 3.3.6. This system, along with a linear path, is shown in Figure 3.8. The model of this system is

$$\begin{aligned}
\dot{x}_{c_1} &= x_{v_1} \\
\dot{x}_{c_2} &= x_{v_2} \\
\dot{x}_{c_3} &= x_{v_3} \\
\dot{x}_{v_1} &= \frac{1}{M}\tau_1 \\
\dot{x}_{v_2} &= 0 \\
\dot{x}_{v_3} &= \tau_2,
\end{aligned}$$

where $y = h(x_c) = \text{col}(x_{c_1}, x_{c_2})$ is the position of the puck, and x_{c_3} is the puck's orientation, which is affected by τ_2. The parameterized path is $\sigma(\theta) = \text{col}(\theta, \theta)$, and satisfies

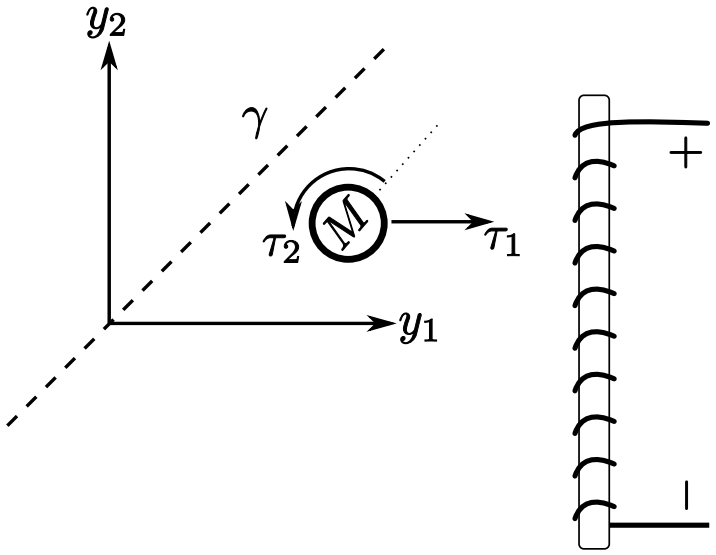


Figure 3.8: Puck with position and rotation on a flat surface

Assumptions 1 and 2, so that it may be expressed as $\gamma = \{y \in \mathbb{R}^2 : s(y) = y_1 - y_2 = 0\}$. Intuitively, τ_2 cannot cause any observable tangential motions along this path. We find

$$\frac{\partial h}{\partial x_c} g_v(x_c) = \frac{1}{M} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad M > 0,$$

whose rank is $1 < p$, which means that condition (3.26) fails, and the we cannot impose further structure on the η-subsystem, even though transverse feedback linearization is possible. Therefore, to perform transverse feedback linearization, we use the coordinate transformation $T : x \mapsto (\eta, \xi)$ such that

$$T(x) = \text{col}(\varphi_1(x), \ldots, \varphi_4(x), \lambda(x_c), L_f\lambda(x)),$$

where $\lambda(x_c) = s \circ h(x_c)$ and $\varphi(x) = \text{col}(\varphi_1(x), \ldots, \varphi_4(x))$ is chosen such that T is a diffeomorphism. After performing transverse feedback linearization, our system, with coordinates $(\eta, \xi) := \text{col}(\eta_1, \xi) = \text{col}(\eta_1^1, \eta_2^1, \eta_3^1, \eta_4^1, \xi_1, \xi_2)$, is expressed as

$$\begin{aligned} \dot{\eta}_1 &= f^0(\eta_1, \xi) + g^{\pitchfork}(\eta_1, \xi) v^{\pitchfork} + g^{\parallel}(\eta_1, \xi) v_1^{\parallel} \\ \dot{\xi}_1 &= \xi_2 \\ \dot{\xi}_2 &= v^{\pitchfork}, \end{aligned}$$

where $\dim(\eta_1) = 4$, $\dim\left(v^{\pitchfork}\right) = 1$ and $\dim\left(v_1^{\parallel}\right) = 1$. This example shows that some tangential controls are not capable of driving the output along the path.

△

Practical Considerations

Certain practical considerations are important to discuss in reference to the results of Theorem 3.3.5, and the related ideas. For example, we require that the output of the system be in a tubular neighbourhood of the path, and this may seem impractical. However, suppose our output lies outside of the tubular neighbourhood of the path, at which point we must be careful to "turn off" our tangential feedback transformation, since the virtual output augmented with $\pi(x_c)$ may not yield a well-defined vector relative degree. Our transversal control system will drive the output into the tubular neighbourhood in finite time, at which point we may "turn on" our coordinate and feedback transformation used to impose the additional structure on the η-subsystem, as well as implement the corresponding feedback control law.

Another practical consideration is how to choose the beginning and end of a path. For a closed curve, such as the curve of Figure 3.7, the endpoint and start point of the curve coincide with the same point. In the case of a non-closed curve, more care is required. Consider the 1-dimensional path in the plane in Figure 3.9. A start and end point of the curve must be defined along the embedded path, which in practice will be dependent on the particular task being executed by the mechanical system. Any non-closed path will always be a connected subset of an embedded curve. We discuss three distinct cases, corresponding to outputs y_a, y_b and y_c respectively, in this figure.

- Output y_a lies within the tubular neighbourhood of the path, and hence the projection and parameterization are defined, and Theorem 3.3.5 is applicable. This represents the ideal scenario.
- Output y_b does not lie within the tubular neighbourhood of the path, as its orthogonal projection maps to multiple points on the path. However, it is clear that as y_b approaches the path (through the transverse closed-loop dynamics), it will eventually enter the tubular neighbourhood and Theorem 3.3.5 will be applicable.
- Output y_c presents a more challenging situation. Transversal control will *not* drive this output into the tubular neighbourhood of the path. Rather, the transversal controller will drive y_c toward the curve of which the path is a subset (i.e. toward the nearest dashed line in Figure 3.9).

There are many different ways to deal with the scenario corresponding to the output y_c in Figure 3.9. Solving this problem is objective-dependent, and is more of a practical implementation issue than a theoretical issue. As such, much literature is mute on this topic (see [54], [32], [34] among others). Some suggestions for avoiding this problem are

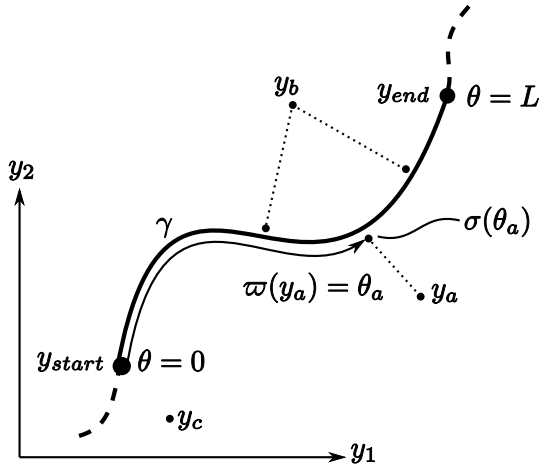


Figure 3.9: A 1-dimensional non-closed curve in the output space illustrating cases where the projection function does not work

- Have the path start and end near boundaries of the feasible workspace. That way the system physically cannot have an output equivalent to that of y_c in Figure 3.9. This suggestion has two main drawbacks: perhaps the desired path does not start/end near the boundaries of the workspace, and being near these boundaries typically corresponds to manipulator singularities

- Extend the path such that the projection and parameterization are valid before the start point and after the end point. Using this approach, one may suitably design the tangential speed profile to accommodate the fact that y_{start} and y_{end} are, practically speaking, the start and end points of the path.

- To perform some *a priori* initialization, such as using a tracking controller to move the output into the tubular neighbourhood of the path.

Chapter 4

Application to a Planar Five-bar Manipulator

In order to demonstrate the theoretical contribution of Chapter 3, we use a simple mechanical system known as a "five-bar linkage robot". Five-bar linkage robots in a parallelogram configuration [50] have been studied in robotics for decades [5], [26], [28]. In this chapter we use the term "manipulator" and "robot" interchangeably, we may refer to this mechanical system as the "five-bar robot", or simply "five-bar". Such a configuration potentially has dynamic and structural advantages over serial manipulator configurations; however, we emphasize that the application of our main results do not rely on these advantages.

4.1 Dynamic Model

For simplicity we consider a fully actuated planar five-bar linkage robot, that is, with two-degrees-of-freedom (2-DOF) as illustrated in Figure 4.1. The full derivation of this model is well documented (we refer the interested reader to [33] , [50]); therefore we will provide a concise overview of the derivation that will serve as the basis for control design.

As discussed in Section 3.2.1, the standard method of deriving the equations of motion for mechanical systems is via the Euler-Lagrange equations. In order to come up with the Lagrangian function (3.7), the kinetic energy, $K(q, \dot{q})$, the potential energy, $V(q)$, and the dissipation function, $R(\dot{q})$, are required.

Finding the kinetic energy of a system entails finding the generalized mass matrix, since $K(q, \dot{q}) = \frac{1}{2}\dot{q}^\top M(q)\dot{q}$. In the case of the five-bar linkage robot in consideration, the mass

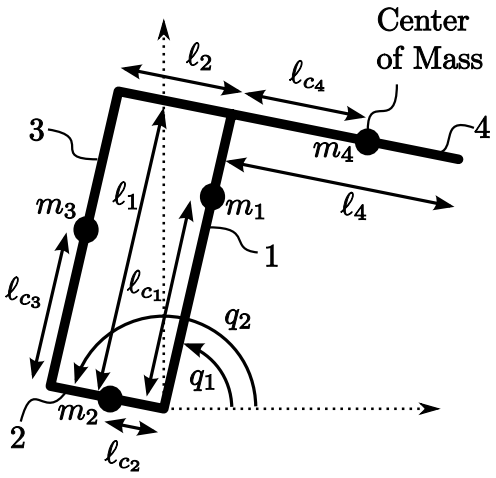


Figure 4.1: Two degree-of-freedom five-bar linkage robot. In this chapter the robot is constrained to move within the plane of the page.

matrix has the form [50, Section 6.4.1]

$$M(q) = \begin{bmatrix} M_{11}(q) & M_{12}(q) \\ M_{21}(q) & M_{22}(q) \end{bmatrix},$$

where

$$\begin{aligned} M_{11}(q) &= m_1\ell_{c_1}^2 + m_3\ell_{c_3}^2 + m_4\ell_1^2 + I_1 + I_3 \\ M_{12}(q) &= M_{21}(q) = (m_3\ell_2\ell_{c_3} - m_4\ell_1\ell_{c_4})\cos(q_2 - q_1) \\ M_{22}(q) &= m_2\ell_{c_2}^2 + m_3\ell_2^2 + m_4\ell_{c_4}^2 + I_2 + I_4, \end{aligned} \tag{4.1}$$

where $\{m_i, I_i, \ell_{c_i}\}$, $i \in \mathbf{4}$ denote the mass, inertia and distances to the centre of mass for the link i, as shown in Figure 4.1. The variables ℓ_1, ℓ_2 are the lengths of the sides of the parallelogram in Figure 4.1. From the expressions (4.1) we see that if

$$m_3\ell_2\ell_{c_3} = m_4\ell_1\ell_{c_4}, \tag{4.2}$$

then the matrix

$$M(q) = \begin{bmatrix} M_{11}(q) & M_{12}(q) \\ M_{21}(q) & M_{22}(q) \end{bmatrix} = \begin{bmatrix} M_{11} & 0 \\ 0 & M_{22} \end{bmatrix},$$

is diagonal and constant. A five-bar linkage robot that satisfies this constraint is said to be *dynamically decoupled*.

In order to obtain the Lagrangian function, we require the potential energy function, which is composed of the configuration-dependent gravitational potential energies of each link. Summing up these energies, we obtain

$$V = g \sin q_1 (m_1 \ell_{c_1} + m_3 \ell_{c_3} + m_4 \ell_1) + g \sin q_2 (m_2 \ell_{c_2} + m_3 \ell_2 - m_4 \ell_{c_4}). \tag{4.3}$$

If the links of the robot are designed such that

$$\begin{aligned} m_1 \ell_{c_1} + m_3 \ell_{c_3} + m_4 \ell_1 &= 0 \\ m_2 \ell_{c_2} + m_3 \ell_2 - m_4 \ell_{c_4} &= 0 \end{aligned} \tag{4.4}$$

then the potential energy function (4.3) is identically equal to zero and the robot is said to be *gravity balanced*. The advantage of mechanically designing the robot to be both dynamically decoupled and gravity balanced is that the dynamic model is greatly simplified. The Lagrangian function becomes

$$L(q, \dot{q}) = \frac{1}{2} \dot{q}^T M \dot{q}.$$

The internal dissipation of the robot is modeled using the simplest possible Rayleigh dissipation function

$$R(\dot{q}) = \frac{1}{2} \sum_{i=1}^{2} b_i \dot{q}_i^2 \tag{4.5}$$

where $b_i \geq 0$, $i \in \mathbf{2}$ are real and constant. This friction model describes any friction linearly dependent on velocity in the system, such as friction in the bearings.

Working out the Euler-Lagrange equations,

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} (q, \dot{q}) \right) - \frac{\partial L}{\partial q} (q, \dot{q}) + \frac{\partial R}{\partial \dot{q}} = \tau,$$

in detail for the dynamically decoupled and gravity balanced robot, the equations of motion simplify to

$$\begin{aligned} M_{11} \ddot{q}_1 + b_1 \dot{q}_1 &= \tau_1 \\ M_{22} \ddot{q}_2 + b_2 \dot{q}_2 &= \tau_2, \end{aligned} \tag{4.6}$$

where $\tau := \mathrm{col}\,(\tau_1, \tau_2)$ are the applied torques acting on the system. We define our generalized coordinates as $x_c = \mathrm{col}(x_{c_1}, x_{c_2}) := \mathrm{col}(q_1, q_2)$ with velocities $x_v = \mathrm{col}(x_{v_1}, x_{v_2}) := \mathrm{col}(\dot{q}_1, \dot{q}_2)$. By representing these two, second order, ordinary differential equations in state-space form with state $x = \mathrm{col}(x_c, x_v)$, we obtain the linear time-invariant control

system

$$\begin{aligned}
\dot{x}_{c_1} &= x_{v_1} \\
\dot{x}_{c_2} &= x_{v_2} \\
\dot{x}_{v_1} &= -\frac{b_1}{M_{11}} x_{v_1} + \frac{1}{M_{11}} \tau_1 \\
\dot{x}_{v_2} &= -\frac{b_2}{M_{22}} x_{v_2} + \frac{1}{M_{22}} \tau_2,
\end{aligned} \tag{4.7}$$

which may be expressed in the form of Section 3.3 as

$$\begin{aligned}
\dot{x} &= \begin{bmatrix} x_v \\ f_v(x) \end{bmatrix} + \begin{bmatrix} 0_{N\times m} \\ g_v(x_c) \end{bmatrix} \tau, \\
&= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{b_1}{M_{11}} & 0 \\ 0 & 0 & 0 & -\frac{b_2}{M_{22}} \end{bmatrix} \begin{bmatrix} x_c \\ x_v \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{1}{M_{11}} & 0 \\ 0 & \frac{1}{M_{22}} \end{bmatrix} \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \\
&=: Fx + G\tau
\end{aligned} \tag{4.8}$$

with $f_v(x) := \left[-\frac{b_1}{M_{11}} x_{v_1} - \frac{b_2}{M_{22}} x_{v_2} \right]^T$ and $g_v(x_c) = \begin{bmatrix} \frac{1}{M_{11}} & 0 \\ 0 & \frac{1}{M_{22}} \end{bmatrix}$. Since the states x_c are joint angles, and x_v are velocities, the state space of this system is $\mathcal{X} := \mathbb{S}^1 \times \mathbb{S}^1 \times \mathbb{R}^2$ where $\mathbb{S}^1$ is the unit circle, i.e., $\mathbb{S}^1 = \mathbb{R} \bmod 2\pi$.

4.2 Manipulator Path Following

In Section 3.2.1, we take the system output, (3.11), as the variable we are interested in controlling. Let the output of (4.8), $y = \operatorname{col}(y_1, y_2) \in \mathbb{R}^2$, denote the position of the end-point of the robot. In terms of the states x_c, the variable y can be expressed as

$$y = h(x_c) := \begin{bmatrix} \ell_1 \cos x_{c_1} - \ell_4 \cos x_{c_2} - \ell_4 \\ \ell_1 \sin x_{c_1} - \ell_4 \sin x_{c_2} - \ell_1 \end{bmatrix}, \tag{4.9}$$

where the origin of the output space corresponds to the joint configuration $x_{c_1} = \frac{\pi}{2}$, $x_{c_2} = \pi$. Model (4.8) with output (4.9) belong to the class of systems with the form (3.13).

The control objective is to make the output y approach and traverse a circular path of radius r, centered at the origin of the output space, given by

$$\sigma(\theta) = \begin{bmatrix} r \cos\left(\frac{\theta}{r}\right) \\ r \sin\left(\frac{\theta}{r}\right) \end{bmatrix}. \tag{4.10}$$

This path satisfies Assumptions 1 and 2, and furthermore, it is a unit speed parameterization, which means that σ is parameterized by its arc length (see Section 3.3.3). The circular path may be written as

$$\gamma := \left\{(y_1, y_2) \in \mathbb{R}^2 : s(y) := y_1^2 + y_2^2 - r^2 = 0\right\}. \tag{4.11}$$

Figure 4.2 illustrates the five-bar linkage robot with the output y and the path γ. This simple path is chosen in order to clearly demonstrate the application of the main theoretical results of Chapter 3.

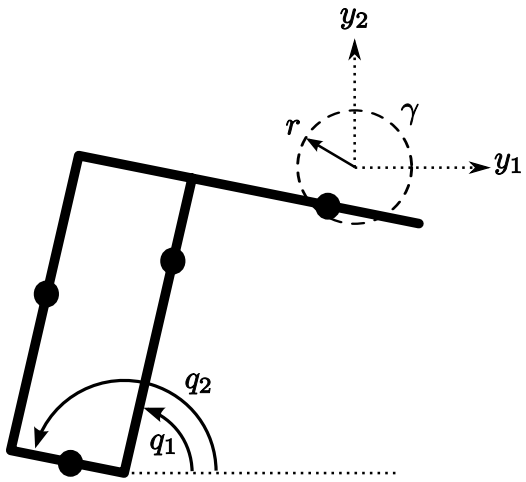


Figure 4.2: Two degree-of-freedom five-bar linkage robot showing the output and path

Before proceeding with our approach, we remind the reader that our system model is already linear and time-invariant. It may seem bizarre to feedback linearize the dynamics of an already linear system; however, this reinforces a key aspect of path following via transverse feedback linearization. Regardless of the original system dynamics, we seek to use coordinates describing these dynamics *with respect to* a set[1]. For a general path, the path following manifold, $\Gamma^\star$, may not be a subspace, which implies that the transverse and tangential coordinates may not be linear in general, even if the system dynamics are linear.

One exception is the case where both the system dynamics, as well as the path in output space, are linear, in which case, $\Gamma^\star$ is indeed a subspace. Linear coordinate and feedback transformations, [19], are sufficient to obtain the normal form (3.2). In this thesis we consider more general paths, and refer the interested reader to [38, Section 3.3] for a more detailed discussion on the LTI case.

[1]The path following manifold, which is the maximal controlled-invariant subset of the lift of the path from output space to state-space.

Application of main results

Our objective is to transversely feedback linearize the five-bar linkage robot (4.7) with output (4.9) and path (4.11), and to control the motion of the output along the set by partially feedback linearizing the tangential dynamics. Since we are interested in feedback linearizing both the transverse and tangential dynamics, we appeal to Corollary 3.3.6, whose basic feasibility requirement, $m > p - 1$, is satisfied. In the system under consideration

$$\frac{\partial h}{\partial x_c} = \begin{bmatrix} -\ell_1 \sin x_{c_1} & \ell_4 \sin x_{c_2} \\ \ell_1 \cos x_{c_1} & -\ell_4 \cos x_{c_2} \end{bmatrix},$$

and

$$g_v(x_c) = \begin{bmatrix} \frac{1}{M_{11}} & 0 \\ 0 & \frac{1}{M_{22}} \end{bmatrix},$$

where

$$\frac{\partial h}{\partial x_c} g_v(x_c) = \begin{bmatrix} -\frac{\ell_1}{M_{11}} \sin x_{c_1} & \frac{\ell_4}{M_{22}} \sin x_{c_2} \\ \frac{\ell_1}{M_{11}} \cos x_{c_1} & -\frac{\ell_4}{M_{22}} \cos x_{c_2} \end{bmatrix},$$

whose determinant is $\frac{\ell_1 \ell_4}{M_{11} M_{22}} \sin(x_{c_1} - x_{c_2})$. Clearly $\frac{\partial h}{\partial x_c} g_v(x_c)$ is non-singular, and therefore has rank $p = 2$, if and only if the configuration corresponding to

$$x_{c_1} - x_{c_2} = k\pi, \; k \in \mathbb{Z} \tag{4.12}$$

is avoided. Condition (4.12) corresponds to the physical situation where the parallelogram in Figure 4.1 collapses and all the links are collinear. As long as the radius r of the circle γ is chosen sufficiently small, this condition will not occur on the path following manifold. Since Corollary 3.3.6 is satisfied, we choose our augmented virtual output as

$$\hat{y} = \begin{bmatrix} \lambda(x_c) \\ \pi(x_c) \end{bmatrix},$$

where

$$\begin{aligned} \lambda(x_c) &= s \circ h(x_c) \\ &= (\ell_1 \cos x_{c_1} - \ell_4 \cos x_{c_2} - \ell_4)^2 + (\ell_1 \sin x_{c_1} - \ell_4 \sin x_{c_2} - \ell_1)^2 - r^2. \end{aligned}$$

We partially define the coordinate transformation and represent the transversal coordinates as

$$\xi := \operatorname{col}(\lambda, L_{Fx}\lambda).$$

The function $\pi(x_c)$ is chosen according to method in Section 3.3.4 as

$$\pi(x_c) = \varpi \circ h(x_c),$$

where $\varpi(y) = \arg\min_{\theta \in \mathcal{L}} \|y - \sigma(\theta)\|$ is the projection of the output onto the path, and $\mathcal{L} = [0, 2\pi r]$. In the case of a circle, we may use the scaled angle of the output along the path, i.e.

$$\varpi(y) = r \arg(y_1 + iy_2).$$

This projection describes the arc length of the output along the path. It is true that this projection is not defined when the endpoint is exactly at the origin of the output space. However, we point out that the results of Chapter 3 are *local* in nature; in a neighbourhood of the path, this projection is well-defined, where in this case, due to the circular path, the projection is well defined on all of $\mathbb{R}^2$ except the origin. The projection is illustrated in Figure 4.3. Since $m = N = p$, we choose $\varphi(x) := \text{col}(\pi(x_c), L_{Fx}\pi(x))$. Consider the

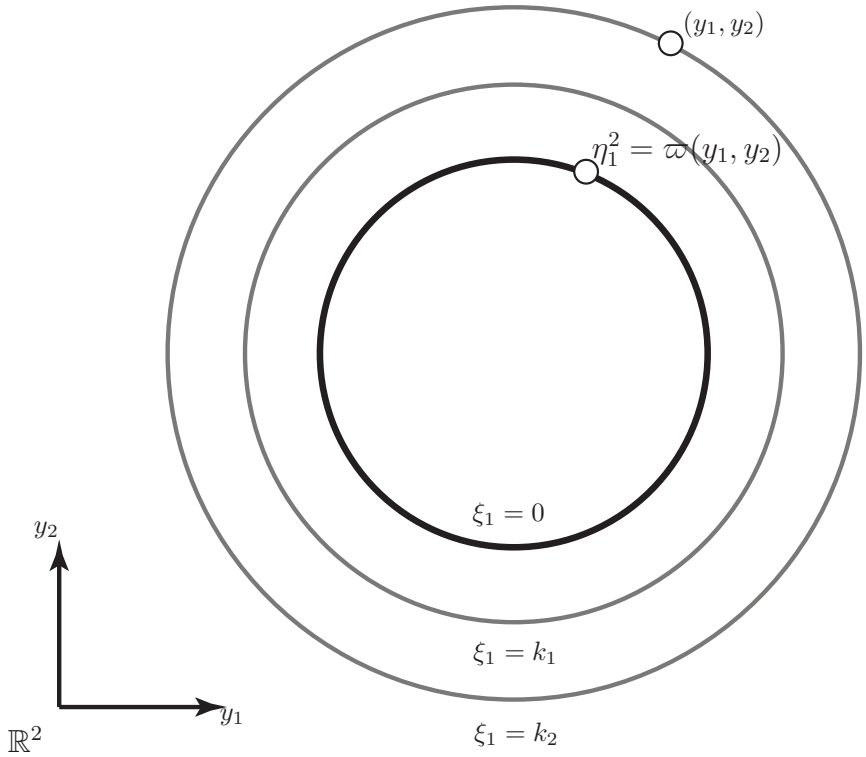


Figure 4.3: The level set $\{(\eta, \xi) : \xi_1 = 0\}$ corresponds to the desired path while the sets $\xi_1 = k_i$, k_i constant, foliate a neighbourhood of the target path. The coordinate η_1^2 represents the projection of (y_1, y_2) onto the desired path.

coordinate transformation

$$\begin{aligned} T :& U \subseteq \mathcal{X} \to T(U) \subseteq \mathbb{S}^1 \times \mathbb{R}^3 \\ & x \mapsto (\eta, \xi) = (\eta_2, \xi) = (\eta_1^2, \eta_2^2, \xi_1, \xi_2) \end{aligned}$$

where

$$\begin{bmatrix} \eta_1^2 \\ \eta_2^2 \\ \xi_1 \\ \xi_2 \end{bmatrix} = \begin{bmatrix} \pi(x_c) \\ \dfrac{\partial \pi(x_c)}{\partial x_c} x_v \\ \lambda(x_c) \\ \dfrac{\partial \lambda(x_c)}{\partial x_c} x_v \end{bmatrix}. \tag{4.13}$$

From Section 3.3.2 we know that locally (4.13) is a diffeomorphism, i.e., when the system output is in a neighbourhood of the path. However, we point out that this example is exceptional, as, although our results are local, we are able to find a single coordinate transformation, (4.13), to get the normal form (3.2) over the entire path. In general, this is not always possible. In (η, ξ)-coordinates the system takes the form

$$\begin{aligned} \dot{\eta}_1^2 &= \eta_2^2 \\ \dot{\eta}_2^2 &= L_{Fx}^2 \pi(x)|_{x=T^{-1}(\eta,\xi)} + L_G L_{Fx} \pi(x)|_{x=T^{-1}(\eta,\xi)} \tau \\ \dot{\xi}_1 &= \xi_2 \\ \dot{\xi}_2 &= L_{Fx}^2 \lambda(x)|_{x=T^{-1}(\eta,\xi)} + L_G L_{Fx} \lambda(x)|_{x=T^{-1}(\eta,\xi)} \tau. \end{aligned}$$

By Theorems 3.3.1 and 3.3.5, our augmented virtual output yields a well defined relative degree with (4.8), and therefore, we take our feedback transformation as

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} L_G L_{Fx} \pi(x) \\ L_G L_{Fx} \lambda(x) \end{bmatrix}^{-1}_{x=T^{-1}(\eta,\xi)} \left(\begin{bmatrix} -L_{Fx}^2 \pi(x) \\ -L_{Fx}^2 \lambda(x) \end{bmatrix}^{-1}_{x=T^{-1}(\eta,\xi)} + \begin{bmatrix} v_2^{\parallel} \\ v^{\pitchfork} \end{bmatrix} \right) \tag{4.14}$$

where $(v_2^{\parallel}, v^{\pitchfork}) \in \mathbb{R}^2$ are new control inputs. The closed-loop system in (η_2, ξ)-coordinates after the feedback (4.14) has the form

$$\begin{aligned} \dot{\eta}_1^2 &= \eta_2^2 \\ \dot{\eta}_2^2 &= v_2^{\parallel} \\ \dot{\xi}_1 &= \xi_2 \\ \dot{\xi}_2 &= v^{\pitchfork}. \end{aligned} \tag{4.15}$$

Notice that (4.15) is fully linear, and because of that, the η_1-subsystem of (3.18) does not appear in the transformed coordinates. In (4.15), we find that the ξ dynamics are a double integrator, and govern output motions transverse to the path, whereas the η dynamics are also a double integrator, and govern output motions tangential to the path.

4.2.1 Transversal and Tangential Control Design

We stabilize the origin of the transversal subsystem in (4.15), thereby solving part (i) of PFP, by means of a PID compensator,

$$v^{\pitchfork}(\xi) = -K_1\xi_1 - K_2\xi_2 - K_3\int_0^t \xi_1(\tau)d\tau, \tag{4.16}$$

with positive gains K_i, $i \in \mathbf{3}$. Since $v^{\pitchfork}(0) = 0$, $\xi = 0$ is an equilibrium point of the closed-loop transversal subsystem. Physically this means that if the robot's end-point has an initial condition on the circle with initial velocity tangent to the circle, then it will remain on the circle for all future time. This property is referred to as path invariance [35].

Achieving the desired motion along the path is equivalent to making sure that either the angular velocity $\eta_2^2(t)$ approaches a desired reference profile $\eta_2^{2\text{ref}}(t)$ or that η_1^2 approach a desired position $\eta_1^{2\text{ref}} \in \mathbb{S}^1$. These goals can be achieved using the tangential control input by means of a simple proportional feedback with feedforward action

$$v^{\parallel} = -K_4\left(\eta_1^2 - \eta_1^{2\text{ref}}\right) + \dot{\eta}_2^{2\text{ref}}(t) - K_5\left(\eta_2^2 - \eta_2^{2\text{ref}}(t)\right). \tag{4.17}$$

When tracking a velocity profile we set $K_4 = 0$. When stabilizing a particular position on the path we set $\eta_2^{2\text{ref}}(t) = 0$. In summary, to implement the overall closed loop controller we must implement the coordinate transformation (4.13), the feedback transformation (4.14) and the transversal and tangential control laws (4.16), (4.17).

4.2.2 Observer Design

The underlying assumption in our development up to this point is that the entire state of system (4.7) is available for feedback. In the experimental apparatus this is not the case, as only states x_c are available for feedback. Let y_m denote the measured output, then $y_m = Cx$ where

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

We use an observer to obtain estimates $\hat{x} := \text{col}\,(\hat{x}_{c_1}, \hat{x}_{c_2}, \hat{x}_{v_1}, \hat{x}_{v_2})$ for the states x. The observer dynamics are

$$\dot{\hat{x}} = F\hat{x} + G\tau + LC(x - \hat{x}). \tag{4.18}$$

where

$$L := \begin{bmatrix} L_{c_1} & 0 \\ 0 & L_{c_2} \\ L_{v_1} & 0 \\ 0 & L_{v_2} \end{bmatrix} \in \mathbb{R}^{4\times 2}$$

is chosen so that the matrix $F + LC$ is Hurwitz. This is possible because the pair (F, C) is observable. Re-writing the dynamics (4.18) in terms of estimation error, $e := x - \hat{x}$, we obtain the dynamics

$$\dot{e} = (F + LC)e;$$

therefore $F + LC$ being Hurwitz ensures $\hat{x}$ approaches x exponentially. Although the error dynamics decay exponentially, the separation principle does not apply, since there is no guarantee that a state estimate error, e, will not make our state leave a neighbourhood of the path following manifold, under the closed loop control law. However, we assume that our initial state estimate is sufficiently close to the actual state. Practically speaking, if the robot is starting from rest, knowing the initial state requires knowing the initial joint angles, which are measurable.

A block diagram of the overall system is provided in Figure 4.4. The state estimate $\hat{x}$ is used to implement the control law (4.14), (4.16), (4.17).

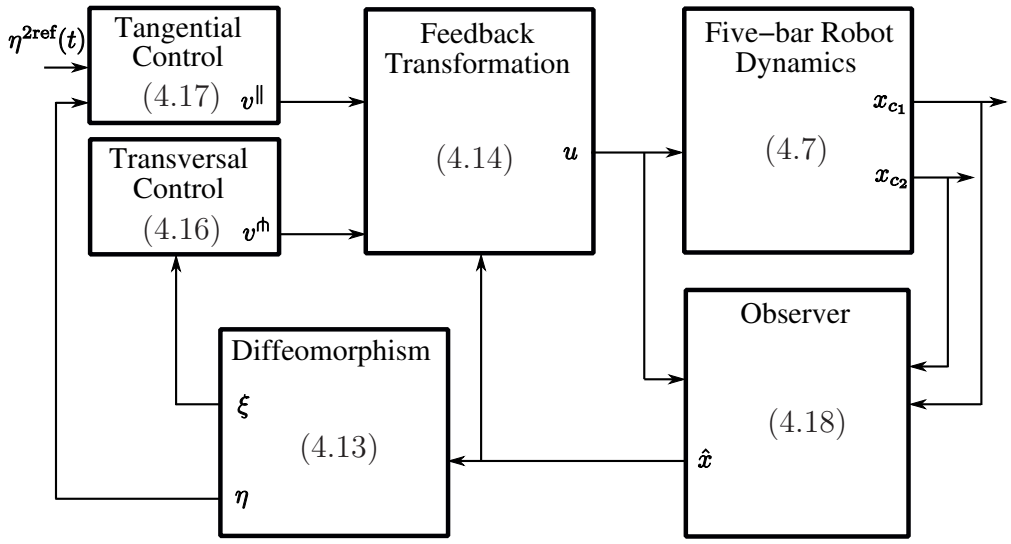


Figure 4.4: Block diagram of five-bar linkage robot feedback control system with equation references

4.3 Experimental Study

4.3.1 Experimental Setup

The experimental platform includes a five-bar linkage robot designed and fabricated at the University of Waterloo [33]. This robot is equipped with counterbalances so that it may

be gravity balanced and dynamically decoupled, as described in Section 4.1. The five-bar linkage robot is actuated by direct drive brushed DC motors with optical encoders. Photographs of this robot are found in Figure 4.5. A Simulink real-time target data acquisition card produces the analog control signals to control the motors via PWM amplifiers, and reads the optical encoder outputs. The sample time is 2 milliseconds.

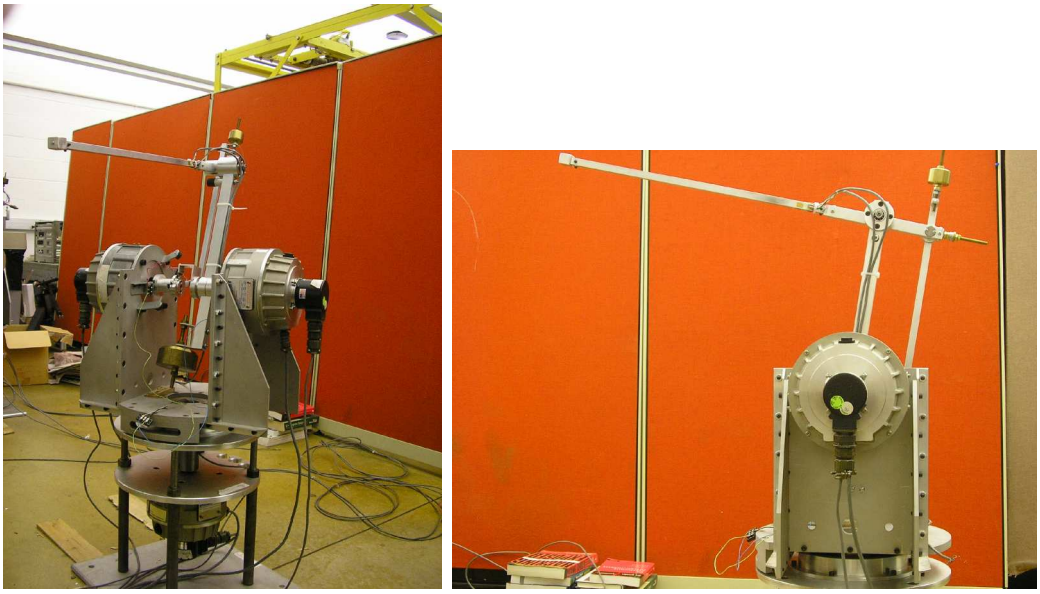


Figure 4.5: Photographs of a gravity-balanced and dynamically decoupled five-bar linkage robot at the University of Waterloo

In the control design we do not take actuator limitations into account, and must be careful not to saturate the DC motors. We appeal to simulation to verify that the control effort is reasonable.

4.3.2 Parameter Identification

The model parameters b_1, b_2, M_{11} and M_{22} of (4.7), were found using the system identification procedure in [33], which is based on measuring characteristics of the step response of second order LTI systems. Table 4.1 presents the results of this procedure along with the lengths ℓ_1 and ℓ_4.

Parameter	Units	Value
b_1	$\frac{kg\,m}{s}$	0.25
b_2	$\frac{kg\,m}{s}$	0.12
M_{11}	$kg\,m^2$	0.32
M_{22}	$kg\,m^2$	0.05
ℓ_1	m	0.3
ℓ_4	m	0.593

Table 4.1: Five-bar manipulator model parameters

4.3.3 Gain Selection

We use a Linear Quadratic Regulator (LQR) to determine the gains of (4.16) and (4.17). For the transversal controller our gains are calculated using the weight matrices $Q^{\pitchfork} := \text{diag}(\text{q}_{11}, \text{q}_{22}, \text{q}_{33})$ and $r^{\pitchfork}$, where $\text{q}_{11} > 0$ penalizes non-zero values of ξ_1, $\text{q}_{22} > 0$ penalizes non-zero values of ξ_2, and $\text{q}_{33} > 0$ penalizes non-zero values of the integral term in (4.16). The scalar $r^{\pitchfork} > 0$ penalizes the transversal control effort, $v^{\pitchfork}$. Since the control objective is to faithfully follow the prescribed path, we choose the relative weighting $\text{q}_{11} > \text{q}_{33} > \text{q}_{22} > r^{\pitchfork}$ so that ξ_1 is driven to zero quickly at the expense of possibly higher control effort. We arrive at the tangential controller gains using the weight matrices $Q^{\parallel}$ and $r^{\parallel}$, where $r^{\parallel} > 0$ is a scalar. The diagonal matrix with positive entries, $Q^{\parallel}$, has dimension 2×2 when the tangential control objective is stabilizing a particular position on the path. In the case of tracking a velocity profile, $Q^{\parallel} > 0$ is a scalar. We select $r^{\parallel} > r^{\pitchfork}$ to allow more control effort to regulate the transversal dynamics than to stabilize the tangential dynamics.

In order to ensure that our observer estimates $\hat{x}$ converge quickly to x we ideally choose high gains on our observer. However due to noise we are unable to make the observer gains arbitrarily high.

4.3.4 Simulation Results

Simulation results for the closed-loop system in Figure 4.4 are now presented to better illustrate path following for the five-bar linkage robot using transverse feedback linearization. Consider a desired velocity profile for the motion of the end-point along the circular path γ given by the reference signal[2]

$$\eta_2^{2\text{ref}}(t) = \begin{cases} 1.5\pi \text{ rad/s} & 0 \leq t < 3\text{s} \\ -1.5\pi \text{ rad/s} & t \geq 3\text{s}. \end{cases}$$

[2]Although we defined our projection to map to the arc length along the path, we will scale the projection such that it is representative of the angle of the output with respect to the center of the circular path; this is a more appropriate projection given that the path is circular.

To allow a fair comparison between simulation and subsequent experimental results, we choose the model parameters of the five-bar linkage robot according to the physical parameters identified in Section 4.3.2. Typical simulation results are shown in Figure 4.6. The simulations illustrate that $\|\xi\| \to 0$ and hence $y(t) \to \gamma$, or in other words, that our output asymptotically (in fact exponentially) approaches the path. They also show the end-point $y(t)$ moving along the path with the desired speed profile because $|\eta_2^2(t) - \eta_2^{2\text{ref}}(t)| \to 0$. The third chart in Figure 4.6 shows the required control effort, i.e., required motor torque, which is feasible given the actuators.

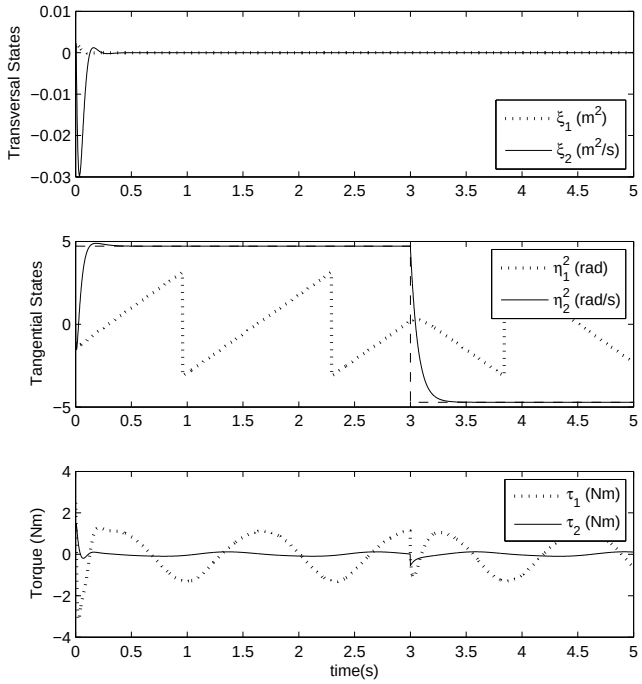


Figure 4.6: Simulation of five-bar linkage robot end-point path following with velocity tracking on a circle of radius 0.05m in the output space

4.3.5 Experimental Results

Three experiments are considered to test the proposed path following control strategy with a circular path γ of radius 0.05 m.

Experiment 1: In the first experiment we stabilize the circle γ and simultaneously stabilize a particular position $\eta_1^{2\text{ref}} = \frac{\pi}{2}$ on γ. Figure 4.7 shows the robot end-point position and the circle γ in output space. Note that the end-point does not cut across the circle

in order to reach the desired position, but rather follows the circular path. The initial condition is chosen so that it is within the domain of definition of the diffeomorphism (4.13) and the feedback transformation (4.14). Figure 4.8 plots ξ, η and u versus time.

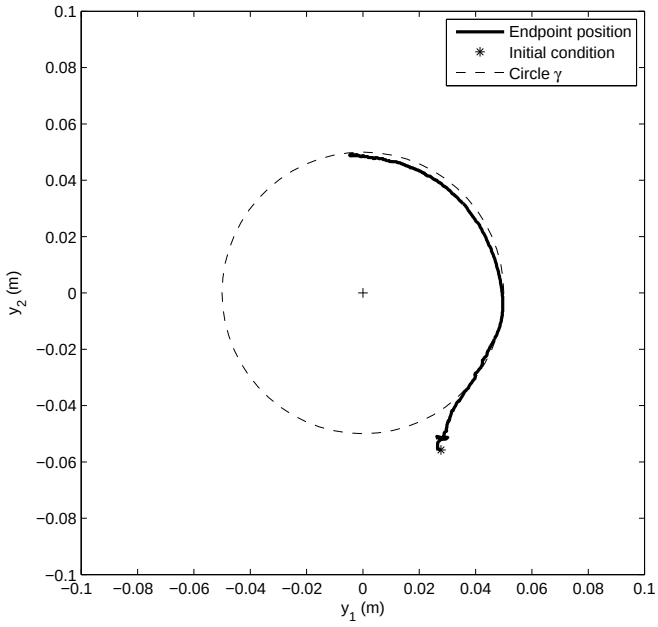


Figure 4.7: End-point trajectory in output space for stabilizing desired position on the circle

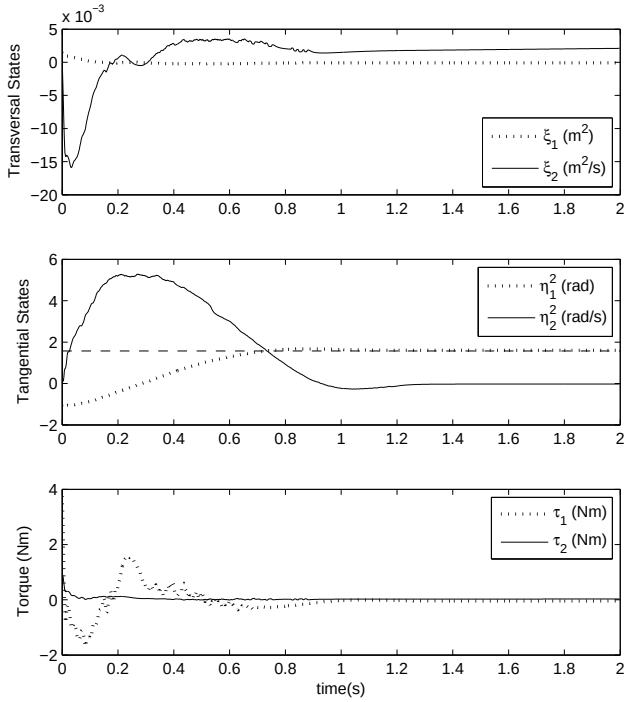


Figure 4.8: Transformed states and control effort for end-point position control

Experiment 2: The second experiment performed demonstrates tracking a velocity profile along the circle. The velocity profile, $\eta_2^{2\text{ref}}(t)$, used in this experiment contains a step change which causes the robot end-point to reverse its direction of traversal. Figure 4.9 plots ξ, η_2 and u versus time. This figure does not show the initial transients as it is meant only to demonstrate tracking a velocity profile. Notice that after the transient behaviour has decayed, the end-point does not leave the circle since $\|\xi\|$ remains small.

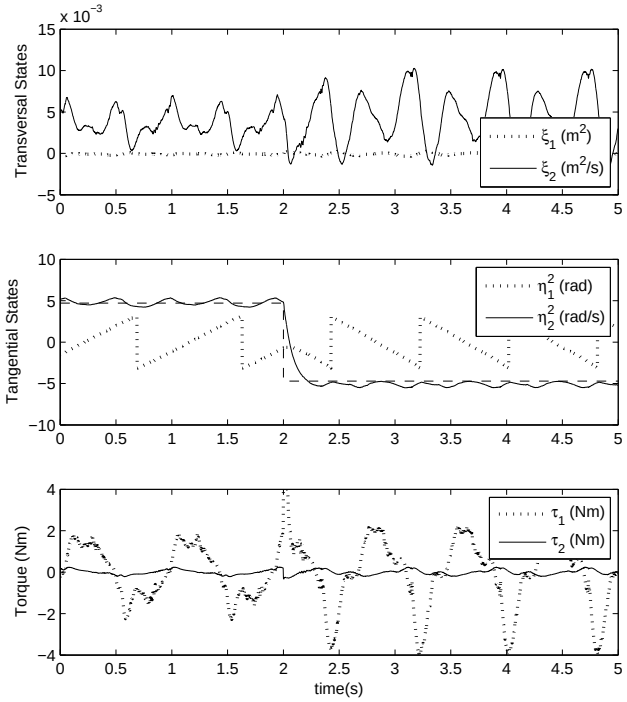


Figure 4.9: Transformed states and control effort for end-point velocity profile tracking

Experiment 3: This experiment demonstrates one of the key advantages of path following over trajectory tracking as discussed in the introduction. Here the five-bar linkage robot end-point tracks a desired velocity profile as in *Experiment 2*. We then temporarily constrain the end-point from moving in the tangential direction of traversal by physically obstructing the path. The end-point is free to move in any other direction. The results shown in Figure 4.10 clearly demonstrate an important feature of path following control, as the end-point remains on the path throughout. Tracking of the desired velocity profile resumes after the obstruction is removed without ever deviating from the path.

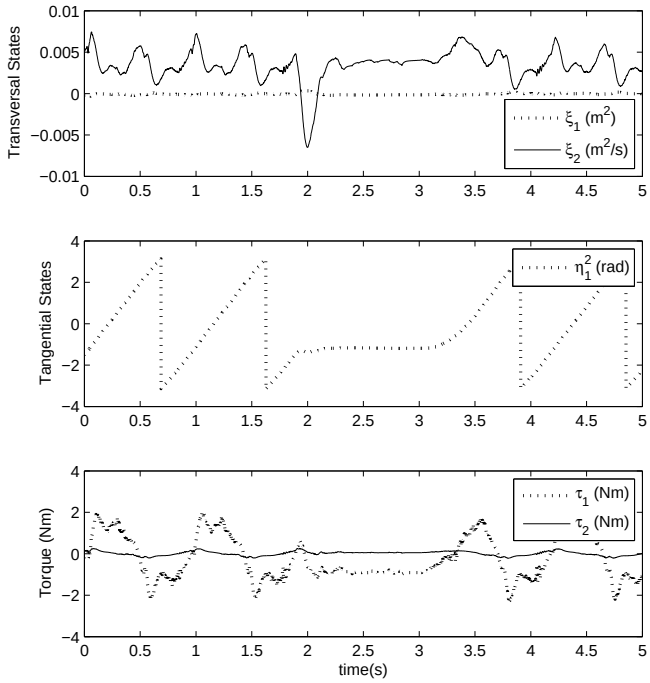


Figure 4.10: Demonstration of path following control by showing transformed states and control effort with temporarily obstructed output

4.3.6 Discussion

The results of *Experiment 1* illustrate important aspects of path following. In Figure 4.7, the end-point position in output space approaches the circle quickly (due to our gain selection), and proceeds to track the desired angular position while traversing the circle. Transverse feedback linearization allows us to accomplish the path following objectives using two linear PID compensators in (η_2, ξ) coordinates.

We attribute unmodeled friction as contributing to discrepancies between experimental and simulation results. Figure 4.9 shows oscillatory behaviour in η_2 and ξ, suggesting an unmodeled coulomb friction. Also, consider ξ_2 in Figure 4.8, where there appears to be a steady state error even in the presence of integral control. We attribute this to the disturbance caused by friction in the state estimation of $\hat{x}$. Since we do not measure x_v, their estimates are susceptible to biases by unmodeled disturbances.

Figure 4.10 captures a main feature of path following control. Notice that while the path is blocked, the end-point remains on the path. Also notice that while blocked, the motor torques are constant, due to the fact that the control system is not tracking a

changing trajectory. Once the path becomes clear again, the robot end-point continues *along the path* as before. Contrast this to trajectory tracking, in which case the motor torques would be changing when the end-point is mechanically constrained as the control system attempts to track a point in the output space. When releasing the obstruction using a trajectory tracking controller, we would expect large torques to allow the end-point to "catch up", and as a result, end-point could *leave the path*, precisely the behaviour we are motivated to avoid.

Chapter 5

Application to a Five-bar Manipulator with a Flexible Link

In Chapter 4 a linear, fully actuated mechanical system was studied in the context of the results presented in Chapter 3. In this chapter a more challenging system is studied, that is, a five-bar linkage robot whose last link is a flexible beam with a 3-dimensional output. The increase in complexity is largely due the fact that a flexible beam is modeled by a partial differential equation, and that a flexible beam control system is non-minimum phase and underactuated.

Controlling flexible structures has drawn the attention of engineers and researchers for decades. In robotics, some of the pioneering research was done attempting to control the endpoint position of a single flexible link. Early work on modeling and control was done by Cannon and Schmitz [10]. Challenges in the research included modeling and inherent difficulties controlling non-minimum phase systems. Many of these challenges were cleverly overcome by redefining the system output, as in Wang and Vidyasagar [59], Chodavarapu and Spong [13], and others. Since then, much progress has been made in modeling and control multi-link flexible manipulators, a summary of which may be found in [7]. It is worthwhile to mention the four control objectives identified by the authors of this survey, ordered from least to most challenging in terms of control:

- Regulating the endpoint position of the flexible manipulator
- Rest to rest motion of the flexible manipulator endpoint in a fixed time
- Joint space trajectory tracking while minimizing link oscillations
- Tracking a desired flexible manipulator endpoint (most challenging)

In this chapter we aim to undertake the fourth bullet, and have the tip of the flexible manipulator follow a desired path.

The reason this chapter focuses on the five-bar linkage robot is because it is a parallel drive manipulator, and has all three motors located near the base. This means that the inertias seen by each joint are lower than in conventional serial link manipulators. Hence, the five-bar linkage robot is a natural candidate for light-weight and high speed applications, which also coincide with applications well suited toward thin, light-weight, and therefore flexible, links.

5.1 Dynamic Model

This chapter considers a five-bar linkage robot with its last link flexible in the horizontal direction only. This control system model has been presented in Wang and Vidyasagar [58]. In this section we first present the modeling of a flexible beam, then proceed to incorporate the flexible beam model with the dynamics of the manipulator.

5.1.1 Modeling Flexible Beam

The simplest example of transverse vibrations in a structure is a beam flexible in only one direction. For the flexible beam of length ℓ, assuming small deflections, we obtain the classic Euler-Bernoulli model,

$$\frac{\partial^2 w(a,t)}{\partial t^2} + EI\frac{\partial^4 w(a,t)}{\partial a^4} = 0, \tag{5.1}$$

where the deflection, $w : [0, \ell] \times \mathbb{R} \to \mathbb{R}$, is a function of two independent variables, the location on the link, a, and time, and EI is a constant representing a mechanical property known as the *flexural rigidity* of the beam [20]. The Euler-Bernoulli model is standard for modeling flexible beams, and has been used in the literature for decades [10], [6], [59], [58], [7], [15]. We point out that the model (5.1) omits damping. Figure 5.1 graphically illustrates the deflection of the flexible beam.

The Euler-Bernoulli beam is modeled by a partial differential equation (PDE), and is referred to as an infinite-dimensional system. Researchers often convert this PDE into a set of ordinary differential equations (ODE's) for analysis and control. The reason is because systems described by ODE's are finite dimensional. The difference between finite and infinite-dimensional systems may be intuitively explained as follows. With an ODE, we must keep track of a finite number of states to characterize the behaviour of a system, for example, the positions and velocities of rigid bodies in a mechanical system. Whereas

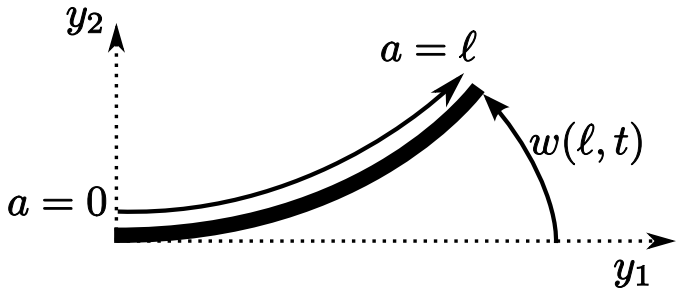


Figure 5.1: Illustration of flexible beam deflection in the plane with particular coordinates.

for a PDE, we must keep track of an infinite number of states, for example, the position and velocity of every infinitesimally small point along a flexible beam. Converting a PDE into a set of ODE's in some sense corresponds to a discretization of the original PDE. For discretization of (5.1), two main approaches have been presented in the literature; the *assumed modes method* and *Finite Element Method (FEM)*. A comparison of these methods is found in [18] and [55]. Some key features of FEM versus assumed modes include:

- Assumed modes provides some physical insights into the system, for example the notion of natural frequencies, where FEM does not
- FEM allows for physically meaningful generalized coordinates, namely displacements and rotations of each node, whereas the generalized coordinates associated with each mode in the assumed modes method do not have clear physical meanings
- Fewer mathematical computations are required in calculating the beam inertia matrix using FEM, whereas typically assumed modes requires fewer differential equations

In this thesis, we exclusively consider the assumed modes approach due to the advantages in having fewer differential equations, a clear notion of natural frequencies, and because the motivating literature in Section 5.2 uses this approach.

The assumed modes approach describes the vibration as a summation of deflections corresponding to a particular frequency mode, and is given by

$$w(a,t) = \sum_{j=1}^{\tilde{N}} q_j(t)\phi_j(a), \qquad \tilde{N} \to \infty \tag{5.2}$$

where $q_j(t)$ is the "weight" of the j^{th} mode, which is the generalized coordinate of that particular vibratory degree-of-freedom. Each mode has an associated frequency, which increases with j. The function $\phi_j(a)$ defines the shape of the j^{th} mode, and is called an

eigenfunction. We simply regard the eigenfunctions as describing a particular mode shape, and omit the associated technical background, for which the interested reader may consult [51]. The functions $\phi_j(a)$ are dependent on the boundary conditions used to model the beam. In this thesis, we consider *clamped free* boundary conditions, meaning that one end of the beam is constrained in terms of position and orientation (Belleza *et al.* [6] provide a more extensive discussion on eigenfunctions for the Euler-Bernoulli beam). The clamped-free eigenfunctions are given by

$$\phi_j(a) = c_j \left[(\sin k_j a - \sinh k_j a) - \frac{\sin k_j \ell + \sinh k_j \ell}{\cos k_j \ell + \cosh k_j \ell} (\cos k_j a - \cosh k_j a) \right], \tag{5.3}$$

where the constants k_j are solutions to

$$\cos k_j \ell \cosh k_j \ell + 1 = 0.$$

and c_j is a normalizing constant so that

$$\int_0^\ell \phi_j(a)^2 da = 1$$

Figure 5.1 describes the link deflection in coordinates corresponding to the clamped-free boundary conditions. As mentioned, we are interested in obtaining a finite number of ODE's as our system model, which is possible by truncating the summation of (5.2). Luckily, the influence of a particular mode on the beam deflection decreases with higher frequencies. Therefore, we assume

$$w(a,t) \approx \sum_{j=1}^{\tilde{N}} q_j(t)\phi_j(a),$$

for some sufficiently large integer $\tilde{N}$. In [56], $\tilde{N} = 3$ was experimentally determined to provide a sufficiently accurate model on five-bar linkage robot with a final flexible link.

5.1.2 Dynamic Model of Five-bar Manipulator with a Flexible Link

General model

In [58], a model for a class of manipulators is presented. This class of systems consists of 3-DOF fully actuated rigid manipulators, with the addition of a final link, flexible in the

horizontal direction only (i.e., in no configurations does gravity affect the vibrations). The class includes elbow, spherical, cylindrical and five-bar linkage robot manipulators [50], each with a final flexible link. Each manipulator in this class has a rotational base joint. This joint is not affected by the generalized masses seen by the other joints. That is, the 3×3 block of the inertia matrix corresponding to the rigid body manipulator has the form

$$\begin{bmatrix} M_{11}(q) & M_{12}(q) & 0 \\ M_{21}(q) & M_{22}(q) & 0 \\ 0 & 0 & M_{33}(q) \end{bmatrix}.$$

It is assumed that the hub inertias of each link (i.e., the constant rotational inertia on the motors themselves) are included in the $M_{11}(q), M_{22}(q)$ and $M_{33}(q)$ terms. The Euler-Lagrange equations for this class of manipulators are

$$\begin{aligned} M_{11}(q)\ddot{q}_1 + M_{21}(q)\ddot{q}_2 - \tfrac{1}{2}\tfrac{\partial M_{33}(q)}{\partial q_1}\dot{q}_3^2 + b_1\dot{q}_1 - \sum_{j=4}^{N} \rho \int_0^{\ell} \phi_j(a) \tfrac{\partial x_\ell(a,q)}{\partial q_1} da \dot{q}_j \dot{q}_3 \\ + \tfrac{1}{2}\tfrac{\partial M_{11}(q)}{\partial q_1}\dot{q}_1^2 + \dot{q}_2\dot{q}_1\tfrac{\partial M_{11}(q)}{\partial q_2} + \dot{q}_2^2\left(\tfrac{\partial M_{12}(q)}{\partial q_2} - \tfrac{1}{2}\tfrac{\partial M_{22}(q)}{\partial q_1}\right) + \tfrac{\partial V}{\partial q_1} &= \tau_1 \end{aligned} \tag{5.4}$$

$$\begin{aligned} M_{22}(q)\ddot{q}_2 + M_{21}(q)\ddot{q}_1 - \tfrac{1}{2}\tfrac{\partial M_{33}(q)}{\partial q_2}\dot{q}_3^2 + b_2\dot{q}_2 - \sum_{j=4}^{N} \rho \int_0^{\ell} \phi_j(a) \tfrac{\partial x_\ell(a,q)}{\partial q_2} da \dot{q}_j \dot{q}_3 \\ + \tfrac{1}{2}\tfrac{\partial M_{22}(q)}{\partial q_2}\dot{q}_2^2 + \dot{q}_2\dot{q}_1\tfrac{\partial M_{22}(q)}{\partial q_1} + \dot{q}_1^2\left(\tfrac{\partial M_{21}(q)}{\partial q_1} - \tfrac{1}{2}\tfrac{\partial M_{11}(q)}{\partial q_2}\right) + \tfrac{\partial V}{\partial q_2} &= \tau_2 \end{aligned} \tag{5.5}$$

$$\begin{aligned} M_{33}(q)\ddot{q}_3 + \tfrac{\partial M_{33}(q)}{\partial q_1}\dot{q}_1\dot{q}_3 + \tfrac{\partial M_{33}(q)}{\partial q_2}\dot{q}_2\dot{q}_3 + b_0\dot{q}_3 + \sum_{j=4}^{N} \rho \int_0^{\ell} \phi_j(a) x_\ell(a,q) da \ddot{q}_j \\ + \sum_{j=4}^{N} \rho \int_0^{\ell} \phi_j(a) \tfrac{\partial x_\ell(a,q)}{\partial q_1} da \dot{q}_j \dot{q}_1 + \sum_{j=4}^{N} \rho \int_0^{\ell} \phi_j(a) \tfrac{\partial x_\ell(a,q)}{\partial q_2} da \dot{q}_j \dot{q}_2 &= \tau_3 \end{aligned} \tag{5.6}$$

$$\begin{aligned} \rho \int_0^{\ell} \phi_j(a) x_\ell(a,q) da \ddot{q}_3 + \ddot{q}_j + \int_0^{\ell} \phi_j(a) \tfrac{\partial x_\ell(a,q)}{\partial q_1} da \dot{q}_3 \dot{q}_1 + \int_0^{\ell} \phi_j(a) \tfrac{\partial x_\ell(a,q)}{\partial q_2} da \dot{q}_3 \dot{q}_2 \\ + q_j \omega_j^2 &= 0. \end{aligned} \tag{5.7}$$

In these equations, $j \in \{4 \ldots N\}$ denotes the index of the modes of vibration, and $\phi_j(a)$ are the clamped-free eigenfunctions corresponding to each mode of vibration of the beam of length ℓ. The term $x_\ell(a,q)$ is the shortest distance from point a on the flexible link to the axis of rotation of the base joint, denoted y_3. The term $x_\ell(a,q)$ is graphically illustrated in Figure 5.2.

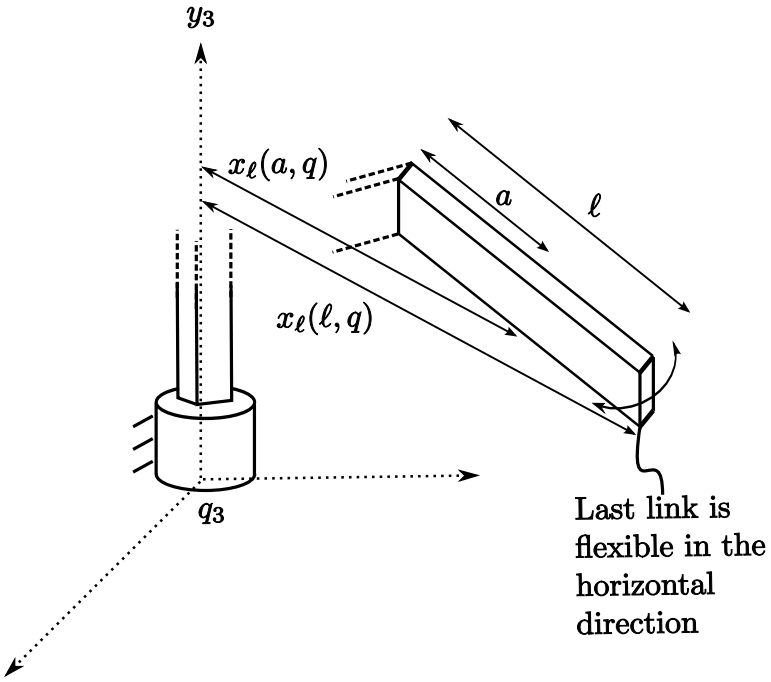


Figure 5.2: Illustration of the term $x_\ell(a,q)$ for a class of manipulators with a horizontal flexibility in the last link.

Model of dynamically decoupled and gravity balanced five-bar linkage robot

The general model for a class of manipulators, which includes the five-bar manipulator with horizontal flexibility in the last link is presented in (5.4), (5.5), (5.6) and (5.7). For a dynamically decoupled ((4.2) must be satisfied) and gravity balanced ((4.4) must be satisfied) five-bar manipulator, these equations of motion may be greatly simplified. Consider the five-bar linkage robot in Figure 5.3. If this robot is dynamically decoupled and gravity balanced[1], the 3×3 block of the inertia matrix corresponding to the rigid body manipulator has the form

$$\begin{bmatrix} M_{11} & 0 & 0 \\ 0 & M_{22} & 0 \\ 0 & 0 & M_{33}(q) \end{bmatrix},$$

[1]Notice that the center of mass of link 1, m_1, extends below the link in Figure 5.3, a necessity for gravity balancing.

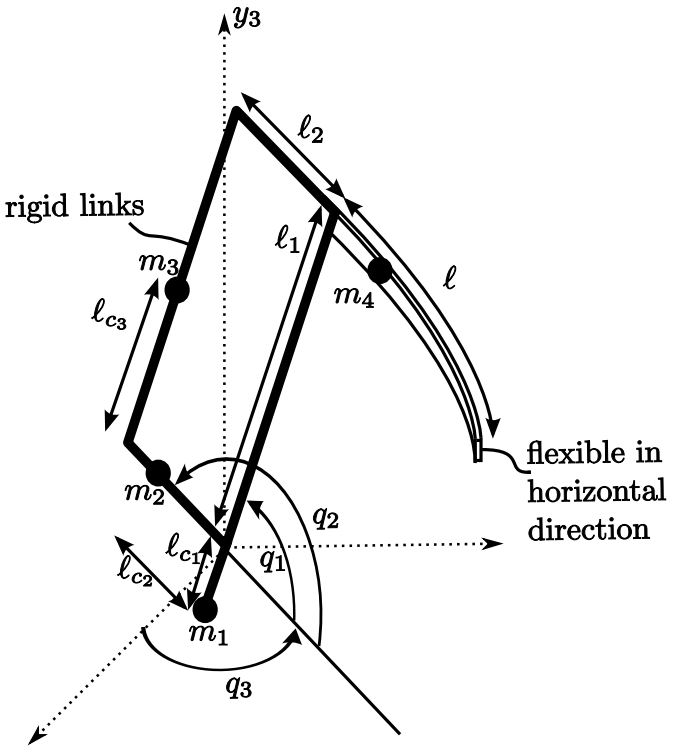


Figure 5.3: A five-bar manipulator with horizontal flexibility in the last link

where

$$M_{11} = m_1\ell_{c_1}^2 + m_3\ell_{c_3}^2 + m_4\ell_1^2 + I_1 + I_3$$
$$M_{22} = m_2\ell_{c_2}^2 + m_3\ell_2^2 + m_4\ell_{c_4}^2 + I_2 + I_4$$
$$M_{33}(q) = M_{11}\cos^2 q_1 + M_{22}\cos^2 q_2 + I_0$$

and $\{m_i, I_i, \ell_{c_i}\}$, $i \in \mathbf{4}$ denote the mass, inertia and distances to the center of mass for the link i, I_0 denotes the hub inertia of the base joint. A 3-DOF gravity balanced and dynamically decoupled five-bar linkage robot does not have linear dynamics, as was the case with the 2-DOF manipulator. Since the manipulator is gravity balanced, the potential energy of the mechanical system does not contain any gravitation terms. However, potential energy corresponding to flexible link deflection [59] is

$$V = \frac{1}{2}\int_0^\ell EI\left(\frac{\partial^2 w}{\partial a^2}\right)^2 da.$$

For the five-bar linkage robot

$$x_\ell(a, q) = \ell_1\cos q_1 - a\cos q_2.$$

In order to simplify notation let

$$\begin{aligned} c_i &:= \cos q_i \\ s_i &:= \sin q_i, \qquad i \in \mathbf{3}, \end{aligned}$$

unless c_i appears in a subscript, where it denotes the center of mass of a link. Also, we define

$$\begin{aligned} \Phi_j &:= \int_0^\ell \phi_j(a)da \\ \Phi_{j_a} &:= \int_0^\ell \phi_j(a)ada \\ \Psi_j(q) &:= \int_0^\ell \phi_j(a)(\ell_1 c_1 - ac_2)da \quad = \quad \ell_1\Phi_j c_1 - \Phi_{j_a} c_2. \end{aligned} \tag{5.8}$$

The equations of motion (5.4), (5.5), (5.6) and (5.7), in the instance of a gravity balanced and dynamically decoupled five-bar linkage robot simplify to

$$M_{11}\ddot{q}_1 + M_{11}\cos q_1 \sin q_1 \dot{q}_3^2 + b_1\dot{q}_1 + \rho\ell_1 \sum_{j=4}^{N} \Phi_j \dot{q}_j \sin q_1\, \dot{q}_3 \;\; = \tau_1 \tag{5.9}$$

$$M_{22}\ddot{q}_2 + M_{22}\cos q_2 \sin q_2 \dot{q}_3^2 + b_2\dot{q}_2 - \rho \sum_{j=4}^{N} \Phi_{j_a} \dot{q}_j \sin q_2\, \dot{q}_3 \;\; = \tau_2 \tag{5.10}$$

$$\begin{aligned} M_{33}(q)\ddot{q}_3 - 2(M_{11}\cos q_1 \sin q_1 \dot{q}_1 + M_{22}\cos q_2 \sin q_2 \dot{q}_2)\,\dot{q}_3 + b_0\dot{q}_3 \\ +\rho \sum_{j=4}^{N} \Psi_j \ddot{q}_j - \rho\ell_1 \sum_{j=4}^{N} \Phi_j \sin q_1\, \dot{q}_1\dot{q}_j + \rho \sum_{j=4}^{N} \Phi_{j_a} \sin q_2\, \dot{q}_2\dot{q}_j \;\; = \tau_3 \end{aligned} \tag{5.11}$$

$$\rho\Psi_j\ddot{q}_3 + \ddot{q}_j - \ell_1\Phi_j \sin q_1\, \dot{q}_3\dot{q}_1 + \Phi_{j_a} \sin q_2\, \dot{q}_3\dot{q}_2 + q_j\omega_j^2 \;\; = 0, \tag{5.12}$$

where $j \in \{4 \ldots N\}$ denotes the index of the generalized coordinates of the modes of vibration. Since these equations are still quite complex, we apply a preliminary feedback cancellation to eliminate many of the coriolis and centripetal terms[2]. The preliminary

[2]Canceling stable dynamics will simplify the model equations; however, the key features of this system cannot be canceled, and therefore this cancellation in no way trivializes the problem.

feedback is

$$\begin{aligned}
\tau_1 &= u_1 + b_1\dot{q}_1 + M_{11}\cos q_1 \sin q_1 \dot{q}_3^2 + \rho\ell_1 \sum_{j=4}^{N} \Phi_j \dot{q}_j \sin q_1\, \dot{q}_3 \\
\tau_2 &= u_2 + b_2\dot{q}_2 + M_{22}\cos q_2 \sin q_2 \dot{q}_3^2 - \rho \sum_{j=4}^{N} \Phi_{j_a} \dot{q}_j \sin q_2\, \dot{q}_3 \\
\tau_3 &= u_3 + b_0\dot{q}_3 - 2(M_{11}\cos q_1 \sin q_1 \dot{q}_1 + M_{22}\cos q_2 \sin q_2 \dot{q}_2)\, \dot{q}_3 \\
&\quad -\rho\ell_1 \sum_{j=4}^{N} \Phi_j \sin q_1\, \dot{q}_1 \dot{q}_j + \rho \sum_{j=4}^{N} \Phi_{j_a} \sin q_2\, \dot{q}_2 \dot{q}_j,
\end{aligned} \tag{5.13}$$

where u_1, u_2, and u_3 are the new control inputs. After the preliminary feedback cancellation (5.13), with the approximate model of the deflection of the flexible beam, the remaining system model is still nonlinear, and can be expressed as

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = u$$

where

$$M(q) = \begin{bmatrix}
M_{11} & 0 & 0 & 0 & 0 & \dots & 0 \\
0 & M_{22} & 0 & 0 & 0 & \dots & 0 \\
0 & 0 & M_{33}(q) & \rho\Psi_4(q) & \rho\Psi_5(q) & \dots & \rho\Psi_N(q) \\
0 & 0 & \rho\Psi_4(q) & 1 & 0 & \dots & 0 \\
0 & 0 & \rho\Psi_5(q) & 0 & 1 & \dots & 0 \\
\vdots & \vdots & \vdots & \vdots & & \ddots & \\
0 & 0 & \rho\Psi_N(q) & 0 & & & 1
\end{bmatrix},$$

$$C(q,\dot{q}) = \begin{bmatrix}
0 & 0 & 0 & \dots & 0 \\
0 & 0 & 0 & \dots & 0 \\
0 & 0 & 0 & \dots & 0 \\
-\ell_1\Phi_4 s_1 \dot{q}_3 & \Phi_{4_a} s_2 \dot{q}_3 & 0 & \dots & 0 \\
-\ell_1\Phi_5 s_1 \dot{q}_3 & \Phi_{5_a} s_2 \dot{q}_3 & 0 & \dots & 0 \\
\vdots & \vdots & \vdots & & \\
-\ell_1\Phi_N s_1 \dot{q}_3 & \Phi_{N_a} s_2 \dot{q}_3 & 0 & \dots & 0
\end{bmatrix},$$

$$G(q) = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & \dots & 0 \\
0 & 0 & 0 & 0 & 0 & & \\
0 & 0 & 0 & 0 & 0 & & \\
0 & 0 & 0 & \omega_4^2 & 0 & & \\
0 & 0 & 0 & 0 & \omega_5^2 & & \\
\vdots & & & & & \ddots & \\
0 & & & & & & \omega_N^2
\end{bmatrix},$$

$$u = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

and $M(q)$ is invertible (see Section 3.2.1). In order to invert $M(q)$, we define

$$M' := -\rho^2 \sum_{i=4}^{n} \Psi_i(q)^2 + M_{33}(q),$$

which is the determinant of the $(N-2) \times (N-2)$ minor of $M(q)$ obtained by removing the first two rows and columns. We appeal to the following useful matrix identity in order to invert $M(q)$:

$$\begin{aligned} A^{-1} &= \begin{bmatrix} a & b & c & d & \cdots \\ b & 1 & 0 & 0 & \cdots \\ c & 0 & 1 & 0 & \cdots \\ d & 0 & 0 & 1 & \cdots \\ \vdots & & & & \ddots \end{bmatrix}^{-1} \\ &= \frac{1}{\det A} \begin{bmatrix} 1 & -b & -c & -d & \cdots \\ -b & -c^2 - d^2 \cdots + a & bc & bd & \cdots \\ -c & bc & -b^2 - d^2 \cdots + a & cd & \cdots \\ -d & bd & cd & -b^2 - c^2 \cdots + a & \cdots \\ \vdots & & & & \ddots \end{bmatrix} \end{aligned}$$

where

$$\det A = a - b^2 - c^2 - d^2 \ldots.$$

Taking the inverse of $M(q)$ we get the following symmetric matrix:

$$M^{-1}(q) = \frac{1}{M'} \begin{bmatrix} \frac{M'}{M_{11}} & 0 & 0 & \cdots & & & \\ & \frac{M'}{M_{22}} & 0 & \cdots & & & \\ & & 1 & -\rho\Psi_4(q) & -\rho\Psi_5(q) & \cdots & -\rho\Psi_N(q) \\ & & & M' + \rho^2\Psi_4(q)^2 & \rho^2\Psi_4(q)\Psi_5(q) & \cdots & \rho^2\Psi_4(q)\Psi_N(q) \\ & & & & M' + \rho^2\Psi_5(q)^2 & \cdots & \rho^2\Psi_5(q)\Psi_N(q) \\ & & & & & \ddots & \\ & & & & & & M' + \rho^2\Psi_N(q)^2 \end{bmatrix}$$

Dropping the argument of the functions $\Psi_j(q)$ for space considerations, and solving for $\ddot{q}$ we obtain

$$
\begin{aligned}
\ddot{q} = & \\
& \frac{-1}{M'} \begin{bmatrix}
0 & 0 & 0 \dots 0 \\
0 & 0 & 0 \dots 0 \\
\rho \ell_1 s_1 \dot{q}_3 \sum_{j=4}^{N} \Psi_j \Phi_j & -\rho s_2 \dot{q}_3 \sum_{j=4}^{N} \Psi_j \Phi_{j_a} & 0 \dots 0 \\
-\ell_1 \Phi_4 s_1 \dot{q}_3 M' - \rho^2 \ell_1 \Psi_4 s_1 \dot{q}_3 \sum_{j=4}^{N} \Psi_j \Phi_j & \Phi_{4_a} s_2 \dot{q}_3 M' + \rho^2 \Psi_4 s_2 \dot{q}_3 \sum_{j=4}^{N} \Psi_j \Phi_{j_a} & 0 \dots 0 \\
-\ell_1 \Phi_5 s_1 \dot{q}_3 M' - \rho^2 \ell_1 \Psi_5 s_1 \dot{q}_3 \sum_{j=4}^{N} \Psi_j \Phi_j & \Phi_{5_a} s_2 \dot{q}_3 M' + \rho^2 \Psi_5 s_2 \dot{q}_3 \sum_{j=4}^{N} \Psi_j \Phi_{j_a} & 0 \dots 0 \\
\vdots & \vdots & \vdots \\
-\ell_1 \Phi_N s_1 \dot{q}_3 M' - \rho^2 \ell_1 \Psi_N s_1 \dot{q}_3 \sum_{j=4}^{N} \Psi_j \Phi_j & \Phi_{N_a} s_2 \dot{q}_3 M' + \rho^2 \Psi_N s_2 \dot{q}_3 \sum_{j=4}^{N} \Psi_j \Phi_{j_a} & 0 \dots 0
\end{bmatrix} \dot{q} \\
& - \frac{1}{M'} \begin{bmatrix}
0 \\
0 \\
-\rho \sum_{j=4}^{N} \Psi_j \omega_j^2 q_i \\
M' \omega_4^2 q_4 + \rho^2 \Psi_4 \sum_{j=4}^{N} \Psi_j \omega_j^2 q_j \\
M' \omega_5^2 q_5 + \rho^2 \Psi_5 \sum_{j=4}^{N} \Psi_j \omega_j^2 q_j \\
\vdots \\
M' \omega_N^2 q_N + \rho^2 \Psi_N \sum_{j=4}^{N} \Psi_j \omega_j^2 q_j
\end{bmatrix} + \begin{bmatrix}
\frac{1}{M_{11}} u_1 \\
\frac{1}{M_{22}} u_2 \\
\frac{1}{M'} u_3 \\
\frac{-\rho \Psi_4}{M'} u_3 \\
\frac{-\rho \Psi_5}{M'} u_3 \\
\vdots \\
\frac{-\rho \Psi_N}{M'} u_3
\end{bmatrix}.
\end{aligned}
$$

In order to make the equations of motion more manageable, we further define the following:

$$\Omega_j(q, \dot{q}) := -\rho \ell_1 s_1 \Phi_j \dot{q}_3 \dot{q}_1 + \rho s_2 \Phi_{j_a} \dot{q}_3 \dot{q}_2 + \rho \omega_j^2 q_j.$$

The equations of motion can be written in the form

$$\begin{aligned}
\ddot{q}_1 &= \tfrac{1}{M_{11}} u_1 \\
\ddot{q}_2 &= \tfrac{1}{M_{22}} u_2 \\
\ddot{q}_3 &= \tfrac{1}{M'} \sum_{j=4}^{N} \Psi_j(q)\Omega_j(q,\dot{q}) + \tfrac{1}{M'} u_3 \\
\ddot{q}_4 &= \tfrac{-1}{M'} \rho \Psi_4(q) \sum_{j=4}^{N} \Psi_j(q)\Omega_j(q,\dot{q}) - \tfrac{\Omega_4(q,\dot{q})}{\rho} - \tfrac{1}{M'} \rho \Psi_4(q) u_3 \\
&\vdots \\
\ddot{q}_N &= \tfrac{-1}{M'} \rho \Psi_N(q) \sum_{j=4}^{N} \Psi_j(q)\Omega_j(q,\dot{q}) - \tfrac{\Omega_N(q,\dot{q})}{\rho} - \tfrac{1}{M'} \rho \Psi_N(q) u_3.
\end{aligned} \tag{5.14}$$

In order to represent (5.14) in the structured form (3.10), let $x := (x_c, x_v)$ with $x_c := q$ and $x_v := \dot{q}$ such that

$$\begin{aligned}
\mathrm{col}(x_{c_1}, \ldots, x_{c_N}) &:= \mathrm{col}(q_1, \ldots, q_N) \\
\mathrm{col}(x_{v_1}, \ldots, x_{v_N}) &:= \mathrm{col}(\dot{q}_1, \ldots, \dot{q}_N).
\end{aligned}$$

Then

$$\dot{x} = f(x) + g(x)u = \begin{bmatrix} x_v \\ f_v(x) \end{bmatrix} + \begin{bmatrix} 0_{N\times m} \\ g_v(x_c) \end{bmatrix} u,$$

where $x_c \in \mathbb{R}^N$, $x_v \in \mathbb{R}^N$, $u \in \mathbb{R}^3$,

$$f_v(x) = \begin{bmatrix} 0 \\ 0 \\ \tfrac{1}{M'} \sum_{j=4}^{N} \Psi_j(x_c)\Omega_j(x) \\ \tfrac{-\rho}{M'} \Psi_4(x_c) \sum_{j=4}^{N} \Psi_j(x_c)\Omega_j(x) - \tfrac{\Omega_4(x)}{\rho} \\ \vdots \\ \tfrac{-\rho}{M'} \Psi_N(x_c) \sum_{j=4}^{N} \Psi_j(x_c)\Omega_j(x) - \tfrac{\Omega_N(x)}{\rho} \end{bmatrix} \tag{5.15}$$

and

$$g_v(x_c)u = \begin{bmatrix} \tfrac{1}{M_{11}} & 0 & 0 \\ 0 & \tfrac{1}{M_{22}} & 0 \\ 0 & 0 & \tfrac{1}{M'} \\ 0 & 0 & \tfrac{-\rho}{M'} \Psi_4(x_c) \\ 0 & 0 & \tfrac{-\rho}{M'} \Psi_5(x_c) \\ \vdots & \vdots & \vdots \\ 0 & 0 & \tfrac{-\rho}{M'} \Psi_N(x_c) \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}. \tag{5.16}$$

We are interested in controlling the position of the tip of the flexible link on the five-bar manipulator. Therefore, as the output, we choose the Cartesian position of the flexible link endpoint in the output space:

$$y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = h(x_c) = \begin{bmatrix} (\ell_1 \cos x_{c_1} - \ell \cos x_{c_2}) \cos x_{c_3} - w(x_c) \sin x_{c_3} \\ (\ell_1 \cos x_{c_1} - \ell \cos x_{c_2}) \sin x_{c_3} + w(x_c) \cos x_{c_3} \\ \ell_1 \sin x_{c_1} - \ell \sin x_{c_2} - \ell_1 \end{bmatrix}, \tag{5.17}$$

where $w(x_c) \approx \sum_{j=4}^{N} \phi_j(\ell) x_{c_j}$ is an approximation of the tip deflection (see Section 5.1.1). A further approximation, valid for small deflections, is that the tip of the beam always lies along a perpendicular line passing through the undeflected tip position. The output is illustrated in Figure 5.4.

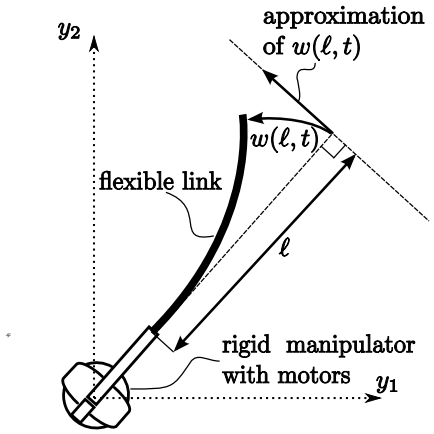


Figure 5.4: Model of link flexibility

5.2 Background and Motivation

Much literature has considered the control of a five-bar linkage robot with its last link flexible (for example Trautman [56], Subrahmanyan and Seshu [52] and Rossi, Zuo and Wang [40]), since it is a non-minimum phase system, and inherently difficult to control. The most relevant contribution to this thesis is a negative result published by Wang and Vidyasagar [58], revealing that feedback linearization is *not* an appropriate control strategy for a class of manipulators, including the five-bar. This negative result serves to motivate investigating our approach to path following applied to the five-bar with a flexible last

link. We are further motivated by literature suggesting path following may remove performance limitations inherent when a tracking approach is taken with non-minimum phase systems [1]. Why the focus on the five-bar linkage robot and not another manipulator? The 3-DOF five-bar is a parallel drive manipulator, and has all three motors located near the base, meaning that the inertias seen by each joint are lower than in serial manipulators. Hence, the motor actuators see a substantially lower inertia, especially in combination with a flexible beam, than in traditional robots.

Summary

In [58], Wang and Vidyasagar investigate applying both input-output and full-state feedback linearization to a class of manipulators with a flexible last link, whose model was presented in this thesis in Section 5.1. The authors first consider input-output feedback linearization. As an output, they choose

$$y = h(x_c) = \begin{bmatrix} h_1(x_c) \\ h_2(x_c) \\ h_3(x_c) \end{bmatrix} = \begin{bmatrix} x_{c_1} \\ x_{c_2} \\ \ell x_{c_3} - w(x_c) \end{bmatrix}, \tag{5.18}$$

where $w(x_c)$ is the deflection given by (5.2), ℓ is the length of the flexible link, and x_{c_1}, x_{c_2}, and x_{c_3} are the joint angles. This unusual output consists of the *reflected tip position*, shown in Figure 5.5, and the two remaining configuration states. The motivation behind using the reflected tip position, rather than the actual tip position, is to preserve a well-defined relative degree of the output h_3 with respect to the manipulator dynamics, as well as other advantages, discussed in detail by Wang and Vidyasagar in [59]. Although using the reflected tip output is effective in mitigating vibrations, this output complicates the output tracking problem.

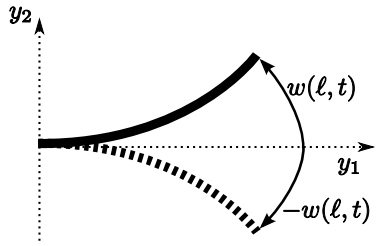


Figure 5.5: Illustration of the reflected tip position

Applying input-output feedback linearization to a manipulator with equations of motion (5.4)-(5.7) with output (5.18), relies on the approximation

$$L_g L_f h_3 \approx \frac{\ell + x_\ell(\ell)}{\bar{M}'}$$

being non-zero, where $\bar{M}' := -\rho^2 \sum_{j=4}^{N} \left(\int_0^{\ell} \phi_j x_\ell da \right)^2 + M_{33}(q)$. Otherwise the vector relative degree is not well-defined. Physically, the situation where $x_\ell(\ell) \approx -\ell$ corresponds to the tip of the flexible link "flipping back" and lying at a distance ℓ from the base joint axis, y_3 (see Figure 5.2). The result is that no feedback transformation exists for a feasible subset of the configuration space for the given output. Wang and Vidyasagar point out that this condition, which relies on the tip of the flexible link "flipping back", may be avoided. Indeed this condition may not even be feasible for certain manipulators, such as the single flexible link, or the five-bar linkage robot (depending on the joint angle limits). Under the assumption that the configuration where $L_g L_f h_3 = 0$ is avoided, Wang and Vidyasagar analyze the zero dynamics of the class of manipulators with the given output. They conclude that the zero dynamics are oscillatory, making the system inherently difficult to control.

Considering the configurations where the feedback transformation fails, as well as the non-minimum phase nature of the input-output feedback linearized system, Wang and Vidyasagar set out to examine full-state feedback linearizability. Applying Theorem 5.2.3 in [25] reveals that this class of manipulators is *not* full-state feedback linearizable. Intuitively, there will always exist dynamics which are uncancellable due to the underactuation of the system.

In conclusion, Wang and Vidyasagar proposed two approaches for the linearization of a class of manipulators with a flexible last link: input-output and full-state linearization. For input-output linearization, the authors resorted to using a fictitious output to preserve relative degree, and found that the linearization is possible given caveats that certain configurations cause failure, and the resulting system is non-minimum phase. Furthermore, they found that full-state linearization fails.

5.3 Flexible Manipulator Path Following

We investigate whether path following is applicable for the five-bar manipulator with a last flexible link using transverse feedback linearization. Furthermore, as in Chapter 4, our objective is to partially linearize the tangential dynamics. We have chosen the following output, which corresponds to the Cartesian coordinates of the tip of the flexible link, not a fictitious output commonly used in the literature (see [59], [58], [40], [13]),

$$y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = h(x_c) = \begin{bmatrix} (\ell_1 \cos x_{c_1} - \ell \cos x_{c_2}) \cos x_{c_3} - w(x_c) \sin x_{c_3} \\ (\ell_1 \cos x_{c_1} - \ell \cos x_{c_2}) \sin x_{c_3} + w(x_c) \cos x_{c_3} \\ \ell_1 \sin x_{c_1} - \ell \sin x_{c_2} - \ell_1 \end{bmatrix}, \tag{5.19}$$

where $w(x_c) \approx \sum_{j=4}^{N} \phi_j(\ell) x_{c_j}$. We would like this output to approach and traverse a parameterized path in the output space, $\sigma : \mathbb{D} \to \mathbb{R}^3$, satisfying Assumptions 1 and 2. This implies that we can express the path as

$$\gamma := \sigma(\mathbb{D}) = \left\{ y \in \mathbb{R}^3 : s(y) = 0 \right\},$$

for some smooth function $s : \mathbb{R}^3 \to \mathbb{R}^2$. We are interested in solving the path following problem by feedback linearizing the transverse dynamics, and also partially feedback linearizing the tangential dynamics. Since $m = p$, we satisfy the basic feasibility requirement to do this. We check if this is possible using Corollary 3.3.6. For the system under consideration, we have

$$\frac{\partial h}{\partial x_c} = \begin{bmatrix} -\ell_1 s_1 c_3 & \ell s_2 c_3 & -(\ell_1 c_1 - \ell c_2) s_3 - \phi_4(\ell) x_{c_4} c_3 & -\phi_4(\ell) s_3 & \dots & -\phi_N(\ell) s_3 \\ -\ell_1 s_1 s_3 & \ell s_2 s_3 & (\ell_1 c_1 - \ell c_2) c_3 - \phi_4(\ell) x_{c_4} s_3 & \phi_4(\ell) c_3 & \dots & \phi_N(\ell) c_3 \\ \ell_1 c_1 & -\ell c_2 & 0 & 0 & \dots & 0 \end{bmatrix},$$

and

$$g_v(x_c) = \begin{bmatrix} \frac{1}{M_{11}} & 0 & 0 \\ 0 & \frac{1}{M_{22}} & 0 \\ 0 & 0 & \frac{1}{M'} \\ 0 & 0 & \frac{-\rho}{M'} \Psi_4(x_c) \\ 0 & 0 & \frac{-\rho}{M'} \Psi_5(x_c) \\ \vdots & \vdots & \vdots \\ 0 & 0 & \frac{-\rho}{M'} \Psi_N(x_c) \end{bmatrix}$$

so that

$$\frac{\partial h}{\partial x_c} g_v(x_c) = \begin{bmatrix} -\frac{\ell_1}{M_{11}} s_1 c_3 & \frac{\ell}{M_{22}} s_2 c_3 & -\frac{1}{M'}(\ell_1 c_1 - \ell c_2) s_3 - \frac{\phi_4(\ell)}{M'} x_{c_4} c_3 + \frac{\rho}{M'} s_3 \sum_{j=4}^{N} \Psi_j(x_c) \phi_j(\ell) \\ -\frac{\ell_1}{M_{11}} s_1 s_3 & \frac{\ell}{M_{22}} s_2 s_3 & \frac{1}{M'}(\ell_1 c_1 - \ell c_2) c_3 - \frac{\phi_4(\ell)}{M'} x_{c_4} s_3 - \frac{\rho}{M'} c_3 \sum_{j=4}^{N} \Psi_j(x_c) \phi_j(\ell) \\ \frac{\ell_1}{M_{11}} c_1 & -\frac{\ell}{M_{22}} c_2 & 0 \end{bmatrix},$$

whose determinant is equal to

$$\frac{\ell \ell_1}{M_{11} M_{22} M'} \sin(x_{c_1} - x_{c_2}) \left(\ell_1 \cos x_{c_1} - \ell \cos x_{c_2} - \rho \sum_{j=4}^{N} \Psi_j(x_c) \phi_j(\ell) \right) \tag{5.20}$$

which equals zero if

$$\sin(x_{c_1} - x_{c_2}) = 0.$$

or

$$\ell_1 \cos x_{c_1} - \ell \cos x_{c_2} - \rho \sum_{j=4}^{N} \Psi_j(x_c)\phi_j(\ell) = 0. \tag{5.21}$$

Physically, the first condition, $\sin(x_{c_1} - x_{c_2}) = 0$ when the parallelogram making up the five-bar linkage robot collapses, and each link is collinear (we encountered this same singular configuration in Chapter 4, equation (4.12)). The second condition is not as intuitive. The term $\ell_1 \cos x_{c_1} - \ell \cos x_{c_2}$ is the distance from the y_3 axis (the axis of rotation of the base joint) to the tip of the last link assuming there is no deflection. In the case of the rigid body manipulator Jacobian, $\ell_1 \cos x_{c_1} = \ell \cos x_{c_2}$ corresponds to the singular configuration where the tip of the last link is directly along the axis of rotation of the base motor. However, in the flexible link case we have the additional term,

$$-\rho \sum_{j=4}^{N} \Psi_j(x_c)\phi_j(\ell),$$

which is from the model of the vibrating beam. Physically, it is difficult to provide concrete reasoning as to why (5.21) must be avoided. Before simulating path following for this system, we numerically check for which configurations (5.21) is encountered.

We will choose our path to avoid the configurations where $\det\left(\frac{\partial h}{\partial x_c} g_v(x_c)\right) = 0$, transversely feedback linearize this system, and also partially linearize all of the tangential dynamics which cause observable output motions along the path. Satisfying the conditions of Corollary 3.3.6 for the flexible five-bar manipulator, a coordinate and feedback transformation will put our system in the normal form (3.2), and furthermore, the tangential dynamics will have additional structure of the form (3.18). In new coordinates, our system becomes

$$\begin{aligned} \dot{\eta}_1 &= f^0(\eta_1, \eta_2, \xi) + g^{\pitchfork}(\eta_1, \eta_2, \xi) v^{\pitchfork} + g_2^{\parallel}(\eta_1, \eta_2, \xi) v_2^{\parallel} \\ \dot{\eta}_2 &= A^{\parallel} \eta_2 + B^{\parallel} v_2^{\parallel} \\ \dot{\xi} &= A\xi + B v^{\pitchfork}, \end{aligned} \tag{5.22}$$

with $\xi \in \mathbb{R}^4$, $\eta_2 \in \mathbb{R}^2$, $\eta_1 \in \mathbb{R}^{2(N-3)}$, and

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \qquad B = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}, \qquad A^{\parallel} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \qquad B^{\parallel} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

When performing a simulation, we work out these equations in detail for a given path.

System (5.22) has interesting features not captured in the rigid planar five-bar manipulator of Chapter 4. For instance, $p = 3$, and therefore the dimension of the ξ-subsystem is $2(p-1) = 4$. Also, the existence of an η_1-subsystem indicates that not all tangential dynamics cause observable motions of the output y. Notice that the dimension of η_1 depends on the number of modes of vibration used in the model; this subsystem can have quite a high dimension. Since $m = p$, no controls, $v_1^{\|}$, appear in the η_1 dynamics. The differential equation

$$\dot{\eta}_1 = f^0(\eta_1, 0, 0)$$

describes the zero dynamics of the system given an augmented virtual output. Intuitively, the zero dynamics correspond to uncontrollable vibrations of the flexible link while the link tip is along the path, and at the desired location on the path (i.e. $\eta_2 = 0$).

5.4 Simulation Results

In this simulation, we will restrict the number of modes of vibration to the lowest three modes. In [56], choosing the lowest three modes is justified according to the desired control bandwidth; higher frequencies contain low-amplitude modes among substantial noise. Thus, the desired control bandwidth excludes higher modes as a result of avoiding noisy frequencies. Our structured model is

$$\begin{aligned}
\dot{x}_c &= x_v \\
\dot{x}_v &= \begin{bmatrix} 0 \\ 0 \\ \frac{1}{M'}\sum_{j=4}^{N}\Psi_j(x_c)\Omega_j(x) \\ \frac{-\rho}{M'}\Psi_4(x_c)\sum_{j=4}^{N}\Psi_j(x_c)\Omega_j(x) - \frac{\Omega_4(x)}{\rho} \\ \frac{-\rho}{M'}\Psi_5(x_c)\sum_{j=4}^{N}\Psi_j(x_c)\Omega_j(x) - \frac{\Omega_5(x)}{\rho} \\ \frac{-\rho}{M'}\Psi_6(x_c)\sum_{j=4}^{N}\Psi_j(x_c)\Omega_j(x) - \frac{\Omega_6(x)}{\rho} \end{bmatrix} + \begin{bmatrix} \frac{1}{M_{11}} & 0 & 0 \\ 0 & \frac{1}{M_{22}} & 0 \\ 0 & 0 & \frac{1}{M'} \\ 0 & 0 & \frac{-\rho}{M'}\Psi_4(x_c) \\ 0 & 0 & \frac{-\rho}{M'}\Psi_5(x_c) \\ 0 & 0 & \frac{-\rho}{M'}\Psi_6(x_c) \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}
\end{aligned}$$

with output

$$y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = h(x_c) = \begin{bmatrix} (\ell_1 \cos x_{c_1} - \ell \cos x_{c_2}) \cos x_{c_3} - w(x_c) \sin x_{c_3} \\ (\ell_1 \cos x_{c_1} - \ell \cos x_{c_2}) \sin x_{c_3} + w(x_c) \cos x_{c_3} \\ \ell_1 \sin x_{c_1} - \ell \sin x_{c_2} - \ell_1 \end{bmatrix},$$

where[3] $w(x_c) = \sum_{j=4}^{6} \phi_j(\ell) x_{c_j}$. In order to demonstrate path following via transverse feedback linearization for the five-bar linkage robot with a flexible last link, we present a simulation using a 1-dimensional path in the output space. We also partially linearize the tangential dynamics.

Path and corresponding output

We choose an ellipse as the embedded path in the 3-dimensional output space. We express the ellipse as the intersection of an infinitely long cylinder of radius r, with a plane whose slope and offset are given by parameters b and d respectively. Therefore our path is

$$\gamma := \left\{ (y_1, y_2, y_3) \in \mathbb{R}^3 : s(y) = 0 \right\},$$

where

$$s(y) = \begin{bmatrix} y_2^2 + (y_3 - \ell_1)^2 - r^2 \\ y_1 - b y_2 - d \end{bmatrix}. \tag{5.23}$$

An illustration of this path is found in Figure 5.6. Given this path, according to Theo-

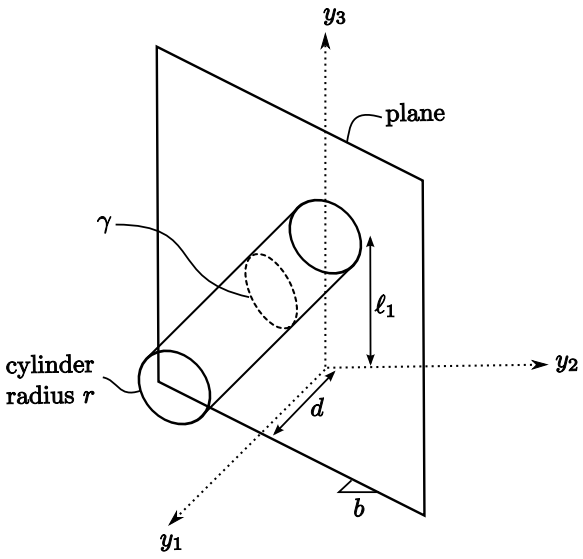


Figure 5.6: Path used for five-bar linkage robot with a flexible link

rem 3.3.1, our virtual output is

$$\hat{y} = \lambda(x_c) = s \circ h(x_c),$$

[3]Technically $w(x_c)$ refers to the *exact* deflection of the tip; however, to avoid notational confusion, we take $w(x_c)$ to refer to the approximate tip deflection.

where $h(x_c)$ is given in (5.17). In order to impose more structure on the η dynamics, as in (5.22), we augment the virtual output with a function which describes the flexible tip position along the ellipse. According the procedure in Chapter 3, we use the projection $\varpi(y) = \arg\min_{\theta \in \mathcal{L}} \|y - \sigma(\theta)\|$, where $\mathcal{L} = [0, 2\pi]$[4]. We use angle of the output with respect to the center of the cylinder,

$$\varpi(y) = \arg(y_2 + (y_3 - \ell_1)i).$$

Intuitively this is a suitable projection: given an angle with respect to the cylinder, one can uniquely identify the endpoint position along the ellipse. Therefore we augment the virtual output with

$$\pi(x_c) = \varpi \circ h(x_c).$$

Recalling, from Section 3.1, that in order to find the coordinate transformation, we require two maps: $\varphi(x)$ which is $2(N-p+1)$ dimensional, and $\mathrm{col}(\lambda(x_c), L_f\lambda(x))$, which is $2(p-1)$ dimensional. The function $\pi(x_c)$ provides 2 of the $2(N-p+1)$ functions comprising $\varphi(x)$. Fortunately, finding the remaining $2(N-p)$ functions is not necessary; these functions govern uncontrollable and unobservable dynamics related to the "internal" vibrations of the system.

Model and simulation parameters

In order to quickly access and manipulate signals, Simulink® is used to implement the continuous time simulation. The simulation uses a fixed-step Dormand-Prince ODE solver with a 2ms sample period. In order to optimize the speed of the simulation, embedded MATLAB blocks are used to where possible. Table 5.1 provides the model parameters used in this simulation. Kinematically, the model is the same as the rigid five-bar linkage robot of Chapter 4. The remaining parameters are chosen so that they loosely correspond to physically realistic values presented in [56].

[4]Chapter 3 assumes that σ is a unit speed parameterization; however, this is not necessary. In Chapter 4 we used a unit speed parameterization, but in this case, we do not.

Parameter	Units	Value
M_{11}	$kg\,m^2$	1
M_{22}	$kg\,m^2$	0.5
ℓ_1	m	0.3
ℓ	m	0.593
ρ	kg/m	0.1
ω_4	rad/s	50
ω_5	rad/s	110
ω_6	rad/s	250
r	m	0.05
d	m	0.4
b	$-$	0.1

Table 5.1: Simulation model parameters

Also, in Table 5.2, the derived constant parameters are provided. The calculations to obtain these derived parameters rely on the actual model parameters, and are given by equations (5.3) and (5.8).

Parameter	Value
$\phi_4(\ell)$	2.253494824644334
$\phi_5(\ell)$	3.749225463985054
$\phi_6(\ell)$	1.873501229335111
Φ_4	0.623815551929889
Φ_5	0.358157464285832
Φ_6	0.547338825746055
Φ_{4_a}	0.261775453410670
Φ_{5_a}	0.183762806307797
Φ_{6_a}	0.134584564095619

Table 5.2: Calculated values based on model parameters for beam flexibility

Although it may seem absurd to present parameters with 15 decimals of precision, recall that the Euler-Bernoulli beam model omits damping, and therefore the model will have oscillatory zero-dynamics. Slight numerical inaccuracies in the mathematical model may cause the zero dynamics to become unstable.

Verifying that Corollary 3.3.6 is satisfied along path

Earlier in this section we commented that we must avoid configurations where the determinant (5.20) is equal to zero. When considering only 3 modes of vibration in the model, this determinant becomes

$$\frac{\ell\ell_1}{M_{11}M_{22}M'}\sin(x_{c_1}-x_{c_2})\left(\ell_1\cos x_{c_1}-\ell\cos x_{c_2}-\rho\sum_{j=4}^{6}\Psi_j(x_c)\phi_j(\ell)\right).$$

This determinant goes to zero when either the manipulator links are collinear, or when

$$\ell_1\cos x_{c_1}-\ell\cos x_{c_2}-\rho\sum_{j=4}^{6}\Psi_j(x_c)\phi_j(\ell)=0. \tag{5.24}$$

Recall that by definition

$$\begin{aligned}\Psi_j(x_c) &= \int_0^{\ell}\phi_j(a)(\ell_1\cos x_{c_1}-a\cos x_{c_2})da\\ &= \quad \ell_1\Phi_j\cos x_{c_1}-\Phi_{j_a}\cos x_{c_2}.\end{aligned}$$

Since $\Psi_j(x_c)$ is a function of x_{c_1} and x_{c_2} only, we numerically plot when condition (5.24) occurs in Figure 5.7. In this plot, for interest's sake, we vary the length of the link ℓ_1 and notice its effect on the singular configurations. Keep in mind that for our simulation, $\ell_1 = 0.3$m.

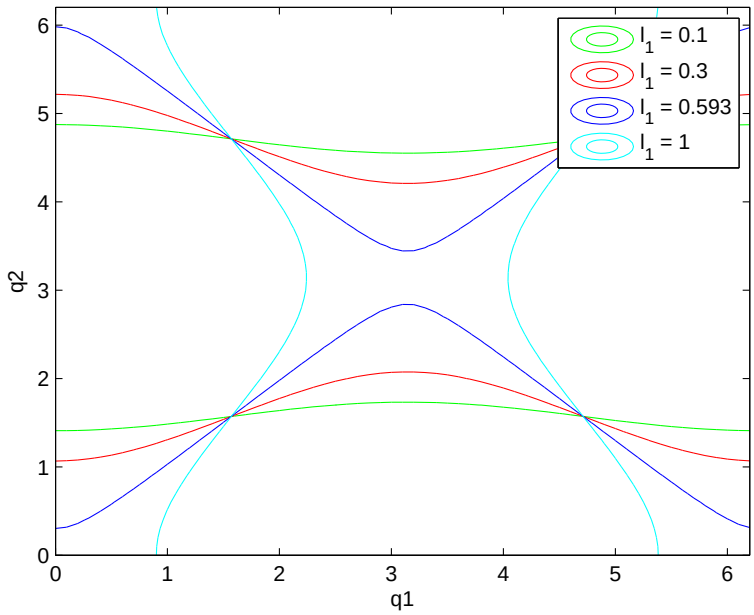


Figure 5.7: Five-bar manipulator joint configurations causing the condition (3.26) to fail

In the output space, these unacceptable regions are shown in Figure 5.8, along with the five-bar manipulator to scale.

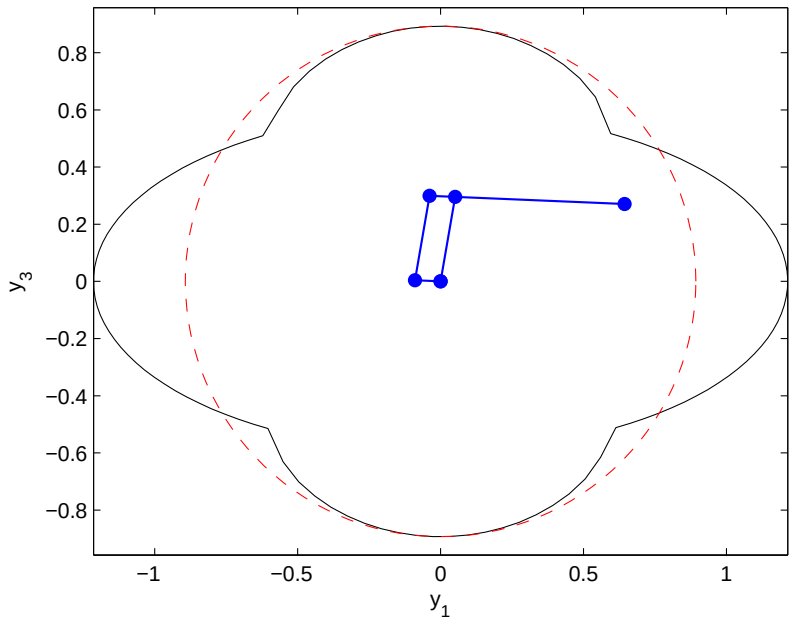


Figure 5.8: Configurations in the output space causing the condition (3.26) to fail, where the dotted line indicates the maximum reach of the manipulator

From Figure 5.8 is clear that unacceptable configurations lie toward the extremities of the feasible workspace. Therefore we have confirmed that these configurations are avoidable, and that our elliptic path does not contain any unacceptable configurations. We may now proceed with simulating this system. Three simulations are presented, in order to illustrate path following via transverse feedback linearization for the five-bar linkage robot with a last flexible link.

Simulation 1

The control objective of the first simulation is to track a step position reference in the tangential subsystem. Implicit in all of the simulations is the objective to stabilize the path following manifold, which is equivalent to stabilizing the origin of the ξ-subsystem.

In reference to (5.22), we use as the control law a PID compensator of the form

$$v_1^{\pitchfork} = -K_{p_1}\xi_1 - K_{d_1}\xi_2 - K_{i_1}\int_0^t \xi_1(\tau)d\tau$$

$$v_2^{\pitchfork} = -K_{p_2}\xi_3 - K_{d_2}\xi_4 - K_{i_2}\int_0^t \xi_3(\tau)d\tau,$$

where $v^{\pitchfork} = \mathrm{col}(v_1^{\pitchfork}, v_2^{\pitchfork}) \in \mathbb{R}^2$ and $K_{p_j}, K_{d_j}, K_{i_j}$, $j \in \mathbf{2}$ are positive gains. Furthermore, we design the tangential control law as a PD compensator of the form

$$v^{\parallel} = -K_{p_3}\left(\eta_1^2 - \eta_1^{2\mathrm{ref}}\right) - K_{d_3}\eta_2^2,$$

where $\eta_1^{2\mathrm{ref}}$ corresponds to the position on the ellipse we wish to stabilize. In this experiment, $\eta_1^{2\mathrm{ref}} = \pi/3$ rad. Table 5.3 contains all of the controller gains, chosen such that the closed loop dynamics of (5.22) are exponentially stable.

Parameter	Value
K_{p_1}	10000
K_{d_1}	1000
K_{i_1}	100
K_{p_2}	10000
K_{d_2}	1000
K_{i_2}	100
K_{p_3}	5
K_{d_3}	5

Table 5.3: Controller gains for flexible manipulator simulation

Figure 5.9 shows the Cartesian position of the flexible tip in output space tracking a step reference in the η_2-subsystem. In Figure 5.10, the corresponding transformed states are presented. Notice that due to the relative weighting of controller gains, the transversal dynamics are much faster than the tangential dynamics, as expected. Finally, Figure 5.11 shows the control effort of each manipulator motor, along with the generalized coordinates (or "weights") of the deflection modes. Notice that due to the fact that only horizontal vibrations are permitted, the vibrations only affect the base joint, as is evident by inspecting the control effort plot.

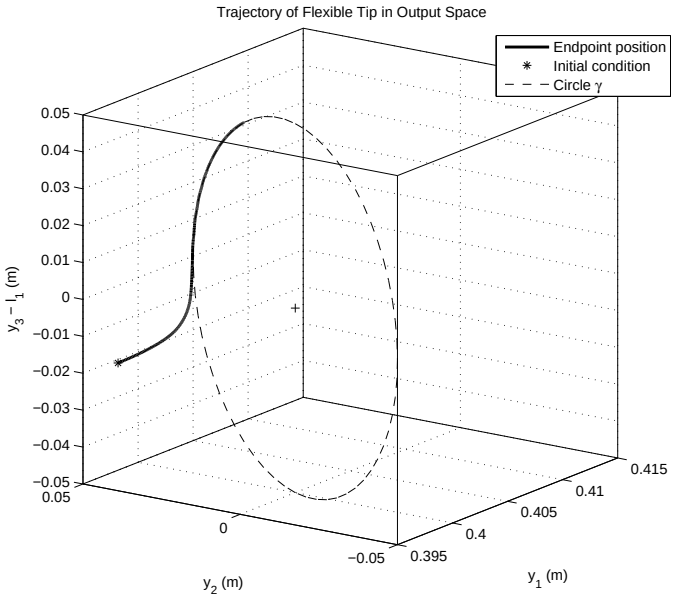


Figure 5.9: Flexible tip in output space

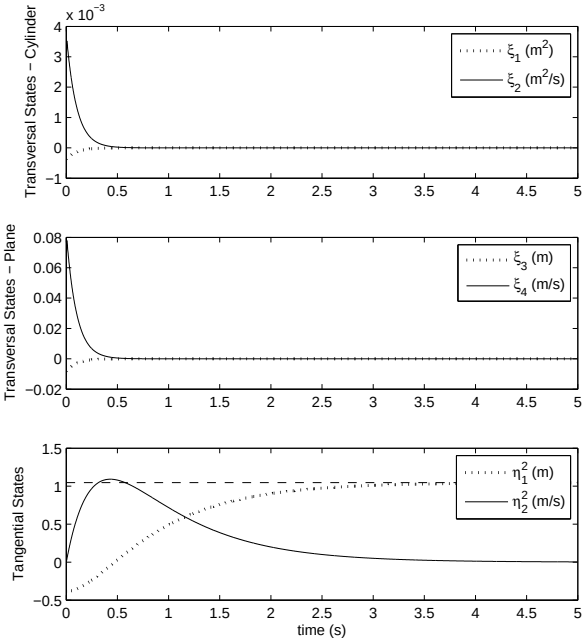


Figure 5.10: Transversal and tangential states for simulation 1

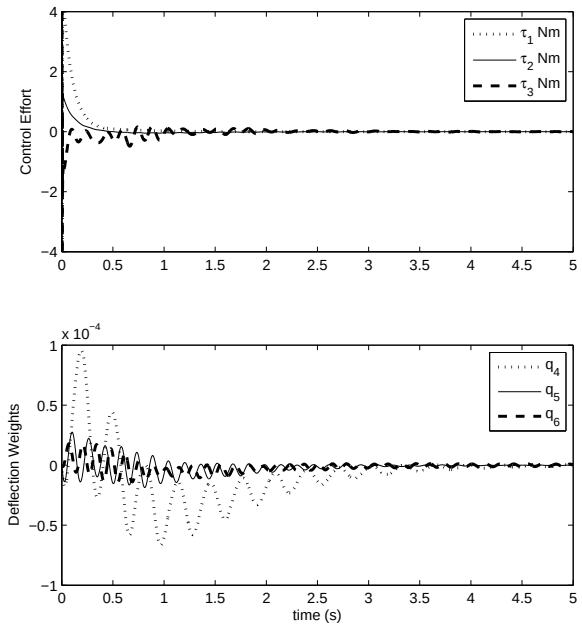


Figure 5.11: Control effort and deflection weights for simulation 1

Simulation 2

The control objective of the second simulation is to track a velocity profile in the tangential subsystem. For this simulation, in reference to (5.22), we use the same PID compensator as in the previous simulation for the transversal subsystem. We design the tangential control law as a compensator of the form

$$v^{\parallel} = -K_{d_3}\left(\eta_2^2 - \eta_2^{2\text{ref}}\right),$$

where we choose $K_{d_3} = 5$ and $\eta_2^{2\text{ref}}$ is the speed profile along the ellipse we wish to track. In this experiment,

$$\eta_2^{2\text{ref}} = \begin{cases} 2 \text{ rad/s} & 0 \leq t < 3\text{s} \\ -2 \text{ rad/s} & t \geq 3\text{s}. \end{cases}$$

Figure 5.12 shows the Cartesian position of the flexible tip in output space tracking the desired speed profile. In Figure 5.13, the corresponding transformed states are presented. Clearly the transversal states are quickly regulated to zero, while the tangential states represent the angle and angular velocity about the cylinder in output space. Finally, Figure 5.14 shows the control effort of each manipulator motor, along with the generalized coordinates (or "weights") of the deflection modes.

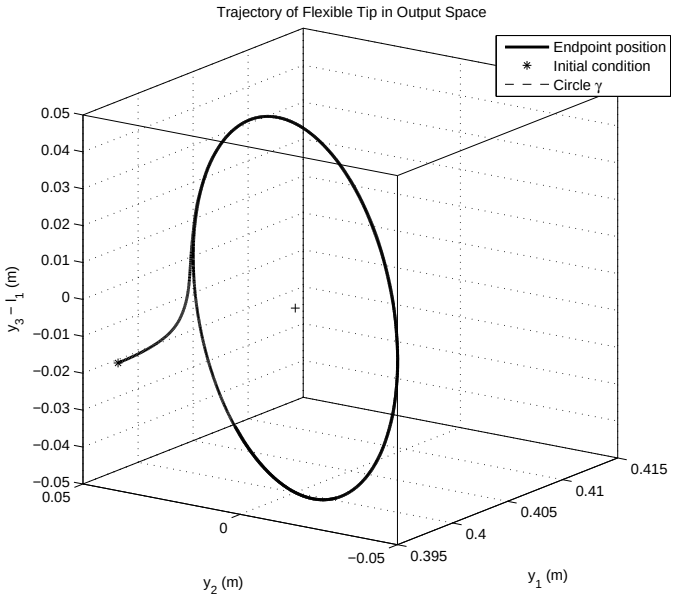


Figure 5.12: Flexible tip in output space - tracking speed profile

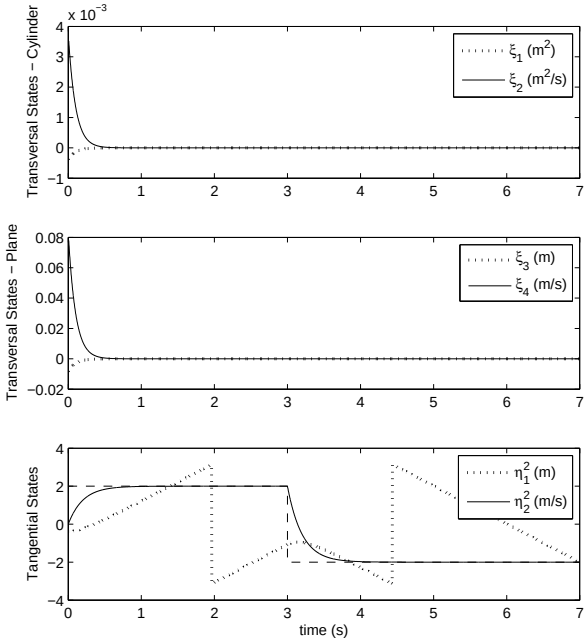


Figure 5.13: Transversal and tangential states for simulation 2

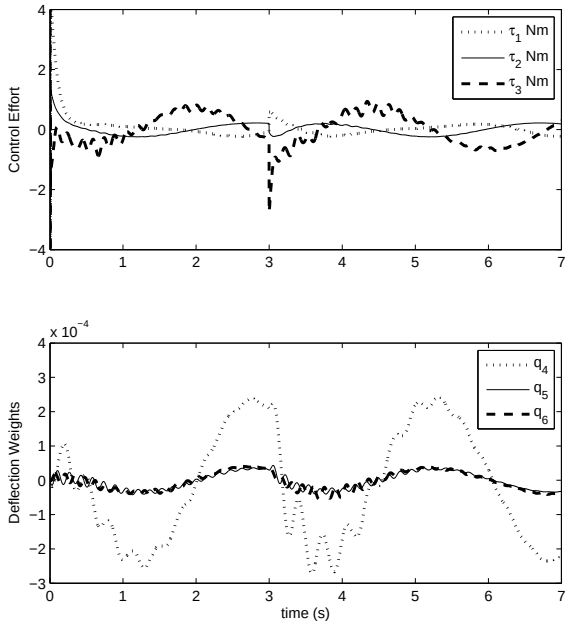


Figure 5.14: Control effort and deflection weights for simulation 2

Simulation 3

The objective of the third simulation is to demonstrate inherent advantages of path following by imposing an unmodeled disturbance and observing the behaviour of the closed-loop system. For this simulation, in reference to (5.22), we use the same PID compensator as in the previous simulation for the transversal subsystem. As before, we use the tangential control law

$$v^{\parallel} = -K_{d_3}\left(\eta_2^2 - \eta_2^{2\text{ref}}\right),$$

where we choose $K_{d_3} = 5$ and $\eta_2^{2\text{ref}}$ is the speed profile along the ellipse we wish to track. In this experiment $\eta_2^{2\text{ref}} = 2$ rad/s. The unmodeled disturbance prevents any vertical motion of the final link of the manipulator. This is done in simulation by temporarily constraining the non-base joint motors q_1 and q_2. This disturbance acts from $3\text{s} < t < 5\text{s}$. The base joint remains unaffected by the disturbance, and therefore horizontal manipulator motion remains possible.

The output space trajectory is shown in Figure 5.15, along with the location of the disturbance. Figure 5.16 shows the transformed states. Since the disturbance discontinuously changes the system dynamics, the resulting "impact" (not physically feasible, since the joint velocities change instantaneously) induces vibrations causing the endpoint to briefly

leave the path. Since the base joint is unaffected by the disturbance, the transversal subsystem is quickly regulated after "impact". Recovering from the impact is also apparent in the tangential dynamics. Finally, Figure 5.17 shows the control effort of each manipulator motor, along with the generalized coordinates (or "weights") of the deflection modes. Interestingly, the controls τ_1 and τ_2 are attempting to help regulate the transversal subsystem; however, due to the nature of the disturbance, their efforts "fall on deaf ears" (i.e., it is as if motors 1 and 2 are jammed).

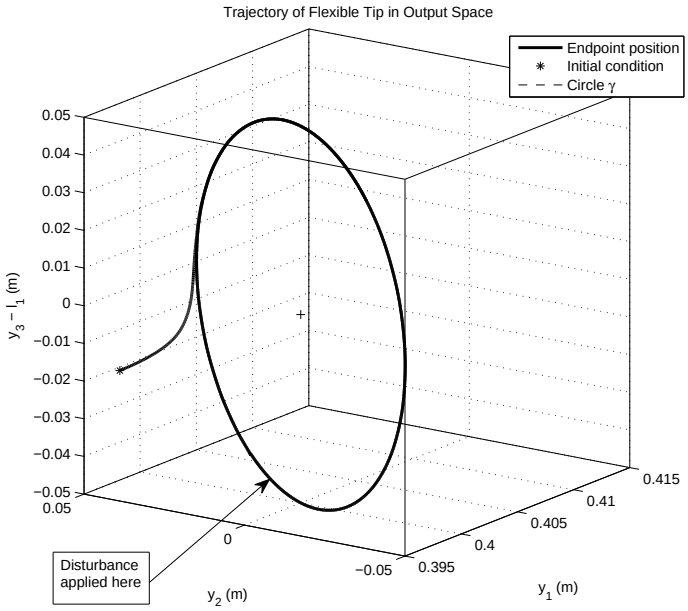


Figure 5.15: Flexible tip in output space - tracking speed profile with unmodeled disturbance

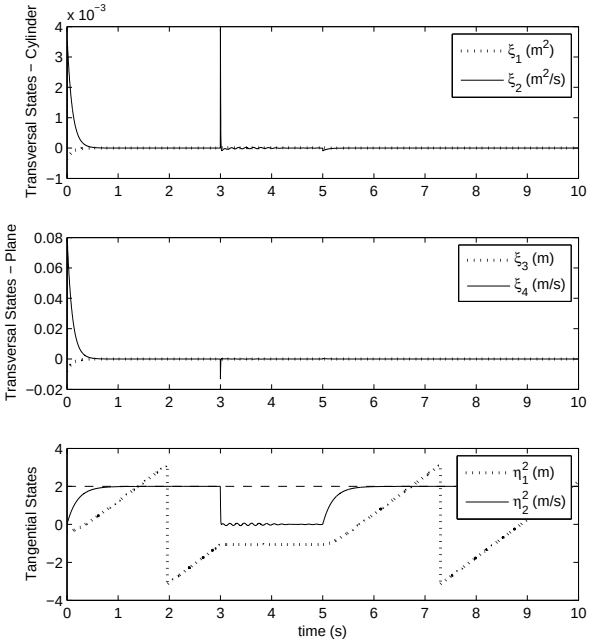


Figure 5.16: Transversal and tangential states for simulation 3

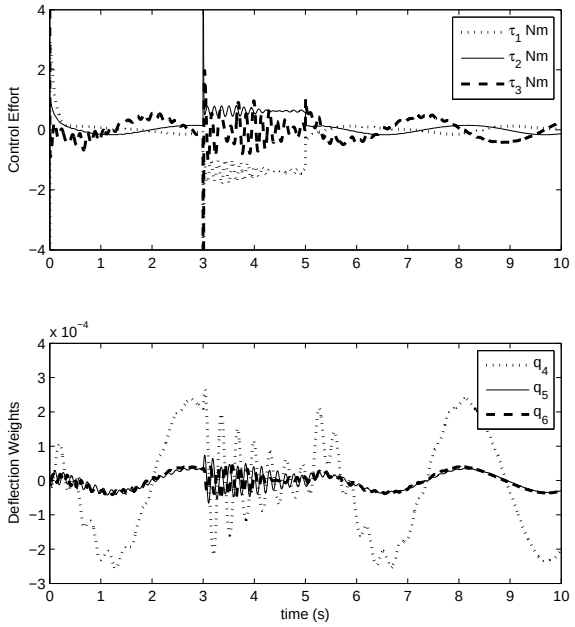


Figure 5.17: Control effort and deflection weights for simulation 3

5.5 Discussion

In this chapter we have presented a model for a five-bar linkage robot with its last link flexible in the horizontal direction. The five-bar robot is well-suited to applications with flexible links since both have the advantage of low inertias, and therefore the capability of fast movement, or motion with low control effort. Previously in the literature, it has been shown that input-state feedback linearization is not possible for this model. Input-output feedback linearization is possible, although singular configurations exist which divide the manipulator workspace. As well, the output must be chosen to be the reflection of the tip position to preserve a well-defined relative degree.

The control of this flexible manipulator fits into the path following formulation under mild restrictions, as Corollary 3.3.6 is satisfied everywhere in the manipulator workspace with the exception of some of its extremities (as shown in Figure 5.8). The output, (5.17), corresponds to the *actual* tip position, not a reflected tip position, and any number of modes may be incorporated into the deflection model, while preserving a well-defined vector relative degree. It is also remarkable that the degree of underactuation, which equals the number of modes of vibration, does not restrict the application of path following to this system.

Three simulations have been presented to demonstrate path following for the endpoint of a five-bar linkage robot with a flexible last link on an elliptic path. Enough tangential control exists to partially feedback linearize the tangential dynamics. Simulation 1 demonstrates the simple objective of getting onto the path, and tracking a desired location on the path. Since they are linear, both the closed-loop η_2 and ξ-subsystems exhibit exponential convergence (Figure 5.10), while the base motor cancels the vibrations of the flexible link (Figure 5.11).

In addition to stabilizing points on the path, path following is useful for tracking desired speeds along the path. Simulation 2 demonstrates tracking a velocity profile. The velocity reference is a constant velocity, which instantaneously switches directions. As expected, the transversal states are quickly regulated, due to relatively high controller gains, and the η_2 states track the desired reference velocity exponentially (Figure 5.13). Notice that the sudden change in direction induces deflections with higher amplitudes, which makes sense intuitively (Figure 5.14), though the output remains on the path as required.

There are advantages simply by formulating our model and selecting our outputs in the context of path following, as opposed to input-output feedback linearization, as have been previously discussed. In addition to these advantages are the advantages inherent to path following, as opposed to trajectory tracking. Simulation 3 demonstrates one such advantage. In this simulation, a constant velocity along the path is tracked, until suddenly, an unmodeled disturbance locks joints q_1 and q_2 (see Figure 5.3). Vibrations are induced

during the sudden locking of the joints (see Figure 5.17). During this time, the transversal controller successfully mitigates these vibrations sufficiently, such that the flexible tip settles on the path. The tangential controller is not able to drive the endpoint along the path. After 2 seconds, the joints become unlocked once again, and the tangential controller begins once more to drive the link tip along the path, picking up where it left off (see Figure 5.16). As has been extensively discussed to this point, this is in stark contrast to the behaviour of a tracking controller under the same disturbance.

Chapter 6

Conclusions and Future Work

Path following is a natural approach for controlling mechanical systems for a variety of applications. Using set stabilization to implement path following has the main advantage of guaranteeing invariance of the path.

In this thesis, we consider a broad class of nonlinear mechanical control systems, of the form (3.13). Our main results provide easily checkable conditions under which the path following problem can be solved as an instance of set stabilization. It is the structure of the class of mechanical systems which makes the conditions of our main results easy to check.

Transverse feedback linearization is a method of converting the dynamics of a mechanical system into a transversal subsystem and a tangential subsystem, with respect to the set we wish to stabilize. The transversal subsystem is decoupled from the tangential subsystem, and is LTI. Stabilizing the origin of the transversal subsystem is equivalent to stabilizing a set which corresponds to the mechanical system output being along a desired path. Given a mechanical system subject to certain assumptions and conditions, we provide necessary and sufficient conditions for the system, along with a "virtual output", to yield a well-defined vector relative degree, in Theorem 3.3.1. A system which satisfies these conditions is transversely feedback linearizable.

It is possible to impose structure on the tangential dynamics using partial feedback linearization. Imposing this structure ultimately allows the motion along the path to be governed by an LTI, controllable subsystem. Necessary and sufficient conditions for obtaining a structured tangential subsystem are described in Theorem 3.3.5. An easily checkable condition in Corollary 3.3.6 reveals whether the mechanical system can be transversely feedback linearized while partially linearizing the tangential dynamics.

The main results been experimentally applied to the planar five-bar manipulator successfully. Next, the main results were applied to a five-bar manipulator whose last link is

flexible. Under mild conditions, Corollary 3.3.6 holds for this manipulator, for which simulations were presented. This is notable, as previous literature shows standard input-output or full state linearization does not work on this robot.

Future Work

The future theoretical contributions stemming from this thesis will include generalizing certain results. For instance, an obvious extension to the contribution of Chapter 3 would be to generalize the concept of a 1-dimensional path in the output space, to a higher dimensional set. Another theoretical contribution will be further analysis of the results presented in this thesis. For example, quantifying the closed-loop system's performance in the presence of disturbances, parametric uncertainty, etc. Finally, we believe it is worth investigating whether or not a vector relative degree of $\{2, \dots, 2\}$ for a mechanical system with virtual output, (3.14), is the *only possible* vector relative degree. If this is proved, then the conditions of Corollaries 3.3.3 and 3.3.6 become necessary and sufficient.

Further experimental work will support the theory presented in this thesis. For example, presenting experimental results on controlling the five-bar manipulator with a flexible link would be an impressive contribution. In this thesis, we thoroughly investigated two robotic systems, both of which were sufficiently actuated in such a way that the tangential dynamics were partially feedback linearizable. Looking at systems where this is not the case would be an interesting future contribution. Also, investigating more interesting paths would be useful from a practical perspective. We have only considered lines, circles and ellipses in this thesis for simplicity. However, more complex paths will yield much rich discussion from a practical perspective. Future work on expanding on this thesis surely promises to be interesting and exciting.

APPENDICES

Appendix A

Positive Definite Matrices are Invertible

A positive definite matrix[1] $A \in \mathbb{R}^{n \times n}$ is invertible

Proof. We use contradiction.

$$\begin{aligned} & A \quad \text{is not invertible} \\ \Rightarrow \quad & \det(A) = 0 \\ \Rightarrow \quad & \ker(A) \neq \{0\} \\ \Rightarrow \quad & \exists\, x \in \mathbb{R}^n \neq 0 \text{ such that } Ax = 0 \\ \Rightarrow \quad & \exists\, x \in \mathbb{R}^n \neq 0 \text{ such that } x^T Ax = 0 \end{aligned}$$

This is a contradiction. Therefore A is invertible if it is positive definite. □

[1] see Definition 2.2.11

Bibliography

[1] A.P. Aguiar, J.P. Hespanha, and P. Kokotović. Path-following for nonminimum phase systems removes performance limitations. *IEEE Transactions on Automatic Control*, 50(2):234–239, 2005.

[2] G. Antonelli, S. Chiaverini, and G. Fusco. An algorithm for online inverse kinematics with path tracking capability under velocity and acceleration constraints. In *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*, volume 5, pages 5079 –5084 vol.5, 2000.

[3] G. Antonelli, S. Chiaverini, and G. Fusco. Experiments of on-line path following under joint limits for an industrial robot manipulator. In *Control Applications, 2002. Proceedings of the 2002 International Conference on*, volume 1, pages 513 – 518 vol.1, 2002.

[4] H. Arai, T. Takubo, Y. Hayashibara, and K. Tanie. Human-Robot Cooperative Manipulation Using a Virtual Nonholonomic Constraint. In *IEEE International Conference on Robotics and Automation*, volume 4, pages 4063–4069, 2000.

[5] Haruhiko Asada and Kamal Youcef-Toumi. Analysis and design of a direct-drive arm with a five-bar-link parallel drive mechanism. *Journal of Dynamic Systems, Measurement, and Control*, 106(3):225–230, 1984.

[6] F. Bellezza, L. Lanari, and G. Ulivi. Exact modeling of the flexible slewing link. In *IEEE International Conference on Robotics and Automation*, volume 1, pages 734 – 739, May 1990.

[7] M. Benosman and G. Le Vey. Control of flexible manipulators: A survey. *Robotica*, 22:533–545, 2004.

[8] J.E. Bobrow, S. Dubowsky, and J.S. Gibson. On the time-optimal control of robotic manipulators with actuator constraints. In *American Control Conference*, volume 2, pages 782–787, 1983.

[9] J.E. Bobrow, S. Dubowsky, and J.S. Gibson. Time-optimal control of robotic manipulators along specified paths. *The International Journal of Robotics Research*, 4(3):3–17, 1985.

[10] R. H. Cannon and Eric Schmitz. Initial Experiments on the End-Point Control of a Flexible One-Link Robot. *International Journal on Robotics Research*, 3(3):62–75, 1984.

[11] Ming-Yang Cheng and Cheng-Chien Lee. On real-time contour error estimation for contour following tasks. In *Advanced Intelligent Mechatronics. Proceedings, 2005 IEEE/ASME International Conference on*, pages 1047 –1052, 24-28 2005.

[12] G.T.-C. Chiu and M. Tomizuka. Contouring control of machine tool feed drive systems: a task coordinate frame approach. *Control Systems Technology, IEEE Transactions on*, 9(1):130 –139, Jan. 2001.

[13] Prasad A. Chodavarapu and Mark W. Spong. On noncollocated control of a single flexible link. In *International Conference on Robotics and Automation*, pages 1101–1106, April 1996.

[14] L. Consolini, M. Maggiore, C. Nielsen, and M. Tosques. Path following for the pvtol aircraft. *submitted to Automatica*, April 2010.

[15] R. Curtain and K. Morris. Transfer functions of distributed parameter systems: A tutorial. *Automatica*, 45(5):1101–1116, 2009.

[16] K. Erkorkmaz and Y. Altintas. High speed contouring control algorithm for cnc machine tools. In *Proceedings of the ASME Dynamic Systems and Control Division, ASME International Mechanical Engineering Congress and Exposition*, page 463469, 1998.

[17] M. Galicki. Path following by the end-effector of a redundant manipulator operating in a dynamic environment. *Robotics, IEEE Transactions on*, 20(6):1018 – 1025, Dec. 2004.

[18] S. S. Ge, T. H. Lee, and G. Zhu. Comparison studies between assumed modes method and finite element method in modeling a single-link flexible manipulator. In *Proceedings of the IEEE International Conference on Intelligent Control and Instrumentation*, pages 376–381, 1995.

[19] S. I. Grossman. *Elementary Linear Algebra*. Thomson Learning, 1994.

[20] R. B. Guenther and J. W. Lee. *Partial differential equations of mathematical physics and integral equations*. Prentice-Hall, Edgewood Cliffs, NJ, 1988.

[21] V. Guillemin and A. Pollack. *Differential Topology*. Prentice-Hall, Englewood Cliffs NJ, 1974.

[22] J. Hauser and R. Hindman. Maneuver regulation from trajectory tracking: Feedback linearizable systems. In *Proc. of the IFAC symposium on Nonlinear Control Systems Design*, pages 595 – 600, Tahoe City, CA, USA, June 1995.

[23] J. M. Hollerbach. Dynamic scaling of manipulator trajectories. *ASME Journal of Dynamic Systems, Measurement and Control*, 106:102–106, 1984.

[24] H.-P. Huang and N.H. McClamroch. Time-optimal control for a robotic contour following problem. *Robotics and Automation, IEEE Journal of*, 4(2):140 –149, Apr 1988.

[25] A. Isidori. *Nonlinear Control Systems*. Springer, New York, 3rd edition, 1995.

[26] H. Kazerooni. Design and analysis of a statically balanced direct-drive manipulator. *Control Systems Magazine, IEEE*, 9(2):30 –34, Feb 1989.

[27] H. K. Khalil. *Nonlinear Systems*. Prentice Hall, Upper Saddle River, NJ, 3rd edition, 2002.

[28] J.Y. Kim. Task based kinematic design of a two dof manipulator with a parallelogram five-bar link mechanism. *Mechatronics*, 16(6):323 – 329, 2006.

[29] Y. Koren. Cross-coupled biaxial computer control for manufacturing systems. *ASME Journal of Dynamic Systems, Measurement and Control*, 102:265 –272, Dec. 1980.

[30] S. M. LaValle. *Path Planning*. Cambridge University Press, New York, 2006.

[31] Dongjun Lee and P.Y. Li. Passive bilateral control and tool dynamics rendering for nonlinear mechanical teleoperators. *Robotics, IEEE Transactions on*, 21(5):936 – 951, Oct. 2005.

[32] P.Y. Li and R. Horowitz. Passive velocity field control of mechanical manipulators. *Robotics and Automation, IEEE Transactions on*, 15(4):751 –763, Aug. 1999.

[33] D. R. Madill. *Modelling and Control of a Haptic Interface: A Mechatronics Approach*. PhD thesis, University of Waterloo, 1998.

[34] U. Mettin, P. X. La Hera, D. O. Morales, A. S. Shiriaev, L. B. Freidovich, and S. Westerberg. Path-constrained trajectory planning and time-independent motion control: Application to a forestry crane. In *14th International Conference on Advanced Robotics (ICAR), Proceedings*, 2009.

[35] C. Nielsen, C. Fulford, and M. Maggiore. Path following using transverse feedback linearization: Application to a maglev positioning system. *Automatica*, 46(3):585–590, 2010.

[36] C. Nielsen and M. Maggiore. On local transverse feedback linearization. *SIAM Journal on Control and Optimization*, 47(5):2227–2250, 2008.

[37] Christopher Nielsen. ECE 688 course notes. University of Waterloo, Winter 2009.

[38] Christopher Nielsen. *Set Stabilization using Transverse Feedback Linearization*. PhD thesis, University of Toronto, 2009.

[39] A. Pressley. *Elementary Differential Geometry*. Springer, New York, 2000.

[40] M. Rossi, K. Zuo, and D. Wang. Issues in the design of passive controllers for flexible link robots. In *IEEE International Conference on Robotics and Automation*, volume 1, pages 321–326, San Diego, CA., May 1994.

[41] Michael Scheint, Marion Sobotka, and Martin Buss. Virtual holonomic constraint approach for planar bipedal walking robots extended to double support. In *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, pages 8180 –8185, Dec. 2009.

[42] Burak Sencer and Yusuf Altintas. Modeling and control of contouring errors for five-axis machine tools—part ii: Precision contour controller design. *Journal of Manufacturing Science and Engineering*, 131(3):031007, 2009.

[43] K. G. Shin and N. D. McKay. Minimum-time control of robotic manipulators with geometric path constraints. *IEEE Transactions on Automatic Control*, AC-30(6):531–541, 1985.

[44] A. Shiriaev, J.W. Perram, and C. Canudas-de Wit. Constructive tool for orbital stabilization of underactuated nonlinear systems: Virtual constraints approach. *Automatic Control, IEEE Transactions on*, 50(8):1164 – 1176, Aug. 2005.

[45] A.S. Shiriaev, L.B. Freidovich, and S.V. Gusev. Transverse linearization for controlled mechanical systems with several passive degrees of freedom. *Automatic Control, IEEE Transactions on*, 55(4):893 –906, April 2010.

[46] Bruno Siciliano. Kinematic control of redundant robot manipulators: A tutorial. *Journal of Intelligent and Robotic Systems*, 3(3):201–212, 1990.

[47] R. Skjente, T.I. Fossen, and P.V. Kokotović. Robust output maneuvering for a class of nonlinear systems. *Automatica*, 40(3):373–383, March 2004.

[48] J.-J.E. Slotine and H.S. Yang. Improving the efficiency of time-optimal path-following algorithms. *Robotics and Automation, IEEE Transactions on*, 5(1):118 –124, Feb 1989.

[49] M. Song, T. Tarn, and N. Xi. Integration of task scheduling, action planning and control in robotic manufacturing systems. *Proceedings of the IEEE*, 88(7):1097–1107, 2000.

[50] M. W. Spong and M. Vidyasagar. *Robot Dynamics and Control*. John Wiley and Sons, 1989.

[51] Walter A. Strauss. *Partial Differential Equations - An Introduction*. John Wiley and Sons, New Jersey, 2nd edition, 2008.

[52] P.K. Subrahmanyan and P. Seshu. Dynamics of a flexible five bar manipulator. *Computers and Structures*, 63(2):283–294, 1997.

[53] T. Takubo, H. Arai, and K. Tanie. Virtual Nonholonomic Constraint for Human-Robot Cooperation in 3-D space. In *IEEE International Conference on Intelligent Robots and Systems*, volume 5, pages 300–305, 2000.

[54] T. Tarn, N. Xi, and A. Bejczy. Path-based approach to integrated planning and control for robotic systems. *Automatica*, 32(12):1675–1687, 1996.

[55] R.J. Theodore and Ghosal A. Comparison of the assumed modes method and finite element models for flexible multilink manipulators. *International Journal of Robotic Systems*, 2:91111, 1995.

[56] C. Trautman. Modelling and control of flexible link manipulators with nonlinear vibrations and two degrees of vibrational freedom. Master's thesis, University of Waterloo, 1995.

[57] A. van der Schaft. *$\mathcal{L}_2$-Gain and Passivity Techniques in Nonlinear Control*. Springer, New York, 2nd edition, 2000.

[58] D. Wang and M. Vidyasagar. Control of a class of manipulators with a single flexible link: Part 1 feedback linearization. *Journal of Dynamic Systems, Measurement and Control*, 113:655–662, 1991.

[59] D. Wang and M. Vidyasagar. Transfer functions for a single flexible link. *International Journal of Robotics Research*, 10(5), 1991.

[60] David W.L. Wang. ECE 682 course notes. University of Waterloo, Fall 2009.

[61] X. Zhang, A. Behal, D.M. Dawson, and J. Chen. A novel passive path following controller for a rehabilitation robot. In *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, volume 5, pages 5374 – 5379 Vol.5, Dec. 2004.