Synchronized closed-path following for a mobile robot and an Euler-Lagrange system

by

Yuqian Li

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, 2013

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

We propose and solve a synchronized path following problem for a differential drive robot modeled as a dynamic unicycle and an Euler-Lagrange system. Each system is assigned a simple closed curve in its output space. The outputs of systems must approach and traverse their assigned curves while synchronizing their motions along the paths. The synchronization problems we study in this thesis include velocity synchronization and position synchronization. Velocity synchronization aims to force the velocities of the systems be equal on the desired paths. Position synchronization entails enforcing a positional constraint between the systems modeled as a constraint function on the paths. After characterizing feasible positional constraints, a finite-time stabilizing control law is used to enforce the position constraint.

Acknowledgements

I would like to thank my supervisor first for all of his support, help, guidance and resourceful meetings during my Master's studies. I am particularly grateful for giving me the great opportunity to work under your supervision. The time we spent working together will be a precious memory in my life.

I am thankful to all the people helped me through my MASc degree, Dr. David Wang, Dr. Shreyas Sundaram, Dr. Stephen L. Smith, Dr. Ravi Mazumdar, Adeel Akhtar, Andre Hladio and Rajan Gill. I would also like to thank my thesis readers Dr. Daniel Davison and Dr. Dana Kulić.

I appreciate the love and support of my family and friends, in completing my thesis.

Dedication

To my parents and Shouhong Zhang.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

This thesis studies a synchronized path following problem for a mobile robot and an Euler-Lagrange system. Path following problems entail designing control algorithms that drive the output of a control system towards a given path in its output space with no timing law assigned to the desired traversal of the path. This distinguishes path following controllers from trajectory tracking controllers. The latter force the system output to track a reference signal that evolves as a function of time. The authors of [3] highlight the fundamental differences between path following and trajectory tracking. A key difference is that a path following controller can guarantee output invariance of a path. More precisely, path following controllers can ensure that if a control system's output is initialized on its desired path, with velocity tangent to the path, the output remains on the path for all future time. In the tracking framework, if a system's output is initialized on the path, but does not coincide with the reference point, then the output may leave its desired path in order to approach the reference point [39]. Path following controllers are more appropriate than trajectory tracking for certain applications such as robotic deburring [34], bipedal robots [49], exercise and rehabilitation machines [34, 63], teleoperation [33], obstacle avoidance [20] and human robot interaction [9, 59].

Synchronization, in the context of control systems, considers the design of control laws that enforce the corresponding flows of multiple dynamic systems to satisfy specific synchronization constraints [36]. Synchronization is currently of great interest in various research communities, including sensor networks [44, 41, 46], communications [50, 30], robotics [45, 57, 13], spacecraft formation flight [62, 60] and biological systems [48]. Developments in technology and requirements for high efficiency in production processes have resulted in widespread use of multi-agent systems [8]. Synchronization control technology is widely used in production processes when tasks cannot be carried out by a single

system [47]. Examples include manufacturing [21], automotive applications and teleoperation [47].

The objective of this thesis is a combination of these two control ideas, i.e., synchronized path following. We present an approach to designing synchronized path following controllers for a differential drive robot and an Euler-Lagrange system applicable to Jordan curves. We first make the position of each system approach and traverse a prescribed closed path in its output space. Then, once on their desired paths, we design control laws to synchronize the motions of the systems along their paths to satisfy a large set of constraints. Synchronizing the motions of multiple dynamic systems on Jordan curves has some affinity to synchronizing the phases of coupled oscillators [54].

The synchronized path following controllers proposed in this thesis have the following desirable properties. First, the closed-loop differential drive robot and the Euler-Lagrange system each achieve output invariance of their desired paths. This property is achieved using only local information by making each system stabilize a suitable set, called the path following manifold, in its state space. Second, the synchronization criteria is defined on each system's path following manifold and modeled as a virtual constraint between the systems. The synchronization constraint is enforced by control laws that use information exchanged between the systems. The constraint is invariant for the closed-loop systems in the sense of that, if the systems are initially synchronized, they remain synchronized for all future time.

1.1 Motivation

Synchronization is applicable when two or more systems are required to work together under a cooperative scheme. Path following is suitable when the systems should move along a path precisely. Synchronized path following, the combination of two control strategies, is used in situations where multiple systems must work cooperatively while moving along their working paths to carry out tasks. To further motivate the study of synchronized path following, we present the following example.

Example: Delivery system in manufacturing

Multiple robots are commonly used in manufacturing [8] to cooperatively and periodically deliver objects to the delivery destination. Suppose one wants to build a delivery system which consists of a robotic manipulator and a ground vehicle. The manipulator is moving

on its path while holding an object token from the delivery starting point. The manipulator is required to place the object on a vehicle which is also following its working path on the ground. After the object is placed on the vehicle, the vehicle carries the object to the final delivery destination and the manipulator returns to the starting point following its path to take another object. In this situation, first we want to control the manipulator and the vehicle to ensure that they stay on their assigned paths respectively. Generally speaking, in manufacturing, the working paths are closed and fixed and we usually want the systems to stay on their paths all the time. Compared to path following, trajectory tracking, as discussed previously, does not guarantee the path invariance. Thus, in this situation, path following is preferable to trajectory tracking. Second, synchronization is needed to make the manipulator and the vehicle arrive synchronously at the point where the manipulator places the object on the vehicle.

1.2 Literature Review

Path following for mobile robots has been deeply investigated in the last decade. An application of path following control to a general n-trailer mobile robot to traverse a curve with varying curvature is presented in [7]. The path following problem is formulated as an output regulation problem. Another application to a tractor-trailer system is proposed in [10] in which a Lyapunov-based control design is applied to asymptotically stabilize the trailer to paths of constant curvature. Adaptive control, graph theory and Lyapunov techniques are applied to design path following control algorithms for a group of mobile robots in [22] that follow an assigned path while maintaining a prescribed inter-robot formation. A feedback control law for a differential drive robot to follow planar curves is presented in [38]. The proposed control law does not involve the computation of the projected position of the unicycle to the path. Alternatively, transverse feedback linearization is used for path following control design for a car-like robot [6] and PVTOL aircraft [16]. The main advantage of this approach is that it guarantees path invariance. Another interesting approach to solve the path following problem is using sliding mode control. The authors of [4] proposed a path following controller using sliding mode control for a dynamic unicycle (see Section 2.1 for definition of a dynamic unicycle). One major advantage of using sliding mode control is that it is robust to localization error. A path following controller for a kinematic unicycle is proposed in [19] by globally, asymptotically stabilizing the motion of the unicycle on a circle using a passivity-based approach.

There are mainly two types of synchronization [8] in the literature, master-slave (or leader-follower) [58, 32] and cooperative schemes [12, 64, 47, 14]. In the master-slave

(leader-follower) case, synchronization is achieved by making all the slaves (followers) synchronize with one master (leader) [32]. In other words, the master or leader system determines the synchronized behaviour of all the other systems. In the second type of synchronization, cooperative schemes, there are inter-connections among all the systems such that each system has influence on the combined dynamics [8].

1.2.1 Path Following Using Serret-Frenet Frames

Serret-Frenet frames are widely used in many papers for solving path following problems. Consider a smooth, unit-speed parameterized curve $\sigma(\cdot)$ in $\mathbb{R}^3$. A Serret-Frenet frame is an orthonormal basis for $\mathbb{R}^3$, defined at each point p, along the curve. The Serret-Frenet frame at a point p consists of a tangent (T), normal (N), binormal (B) unit vector as shown in Figure 1.1.

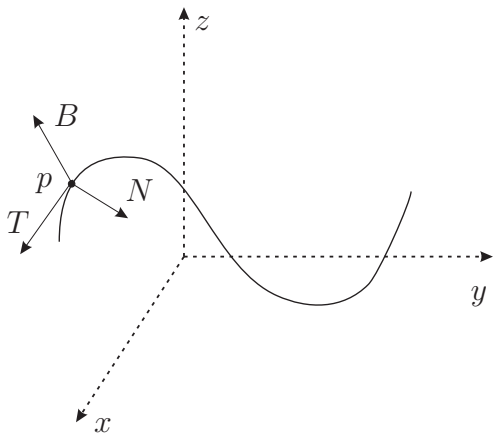


Figure 1.1: Serret-Frenet frames

In [18, Chapter 9] and [7], the path following problem is solved for a unicycle described in a set of coordinates $(s, l, \tilde{\theta})$ based on Serret-Frenet frames. One limitation of this approach is that the transformation to the coordinates $(s, l, \tilde{\theta})$ is local since it relies on the projection of the unicycle to its closest point on the path. In Chapter 4 we use the Serret-Frenet formulas to construct zero level set representations of parameterized curves.

1.2.2 Path Parametrization as a Reference Signal

In many path following problems, the desired paths are given in parameterized form. The parameter in the path parametrization is called the *path variable*. This setting is more general than the common tracking problem, in which the path variable is a given function of time [52]. The parametrization is used as a reference signal and the derivatives of the path variable are treated as extra control inputs. This method is applied in [1]. Those authors, as well as the authors of [17, 3, 37], demonstrate that performance limitations in reference tracking due to unstable zero-dynamics can be removed by using a path following approach. This approach is also used in [52] where path following is referred as the output maneuvering problem. The path following control goal is achieved by solving a geometric task and a dynamic task. The geometric task requires making the system to approach and get on the desired path. The dynamic task involves making the system achieve desired motions along the path, such as tracking a desired velocity profile.

1.2.3 Feedback Linearization

Exact feedback linearization of a nonlinear system simplifies the design of control laws because linear design approaches can be used. Sufficient and necessary conditions for a nonlinear system to be feedback linearizable are presented in [28]. If a nonlinear system is feedback linearizable, one can apply the approach in [52] to solve the path following problem by dividing it into two tasks: a geometric task and a dynamic task.

Transverse feedback linearization involves feedback linearizing that portion of a system's dynamics that determine whether the system is approaching the desired path. The authors in [39] show that transverse feedback linearization guarantees the invariance of simple closed curves and use this method to design path following controllers for a 5-DOF maglev positioning system. Path following control design using transverse feedback linearization for a planar vertical takeoff and landing of aircraft is presented in [16]. It is interesting to note that the path following control design proposed in [16] is applicable to any Jordan curve whose maximum curvature is sufficiently close to that of a circle of equal of length. In [25], the authors showed that the path following problem for a large class of mechanical systems can be solved by designing feedback controllers based on transverse feedback linearization.

1.2.4 Synchronized Path Following and Trajectory Tracking

A passivity-based synchronization control law is developed in [27] for synchronizing the motions of general nonlinear systems while they are following their respective paths. Adaptive, synchronized, trajectory tracking control of a P-R-R type planar parallel manipulator, comprised of three constrained sub-manipulators, is proposed in [45]. The position errors are reduced by controlling motions of all sub-manipulators synchronously and regulating their synchronization error. The synchronization control objective in [13] is to synchronize the movements of a set fully actuated manipulators when they are tracking a common prescribed trajectory. In [57] the authors propose a synchronized trajectory tracking control law that makes each mobile robot track its assigned trajectory and synchronize the motions of other robots simultaneously. Another effective approach to solve position synchronization problems is using adaptive control [56, 55, 13]. Considering parameter uncertainties, the synchronized trajectory tracking control law in [13] is extended to an adaptive synchronization control and robustly synchronizes robots to track a desired trajectory with time delay issues in the network communication channels. The synchronization in the task space for both non-redundant and redundant robotic manipulators in [35] can be accomplished by communicating with each manipulator over balanced graphs while tracking a given trajectory. The work in [45, 57, 13, 35] shows that trajectory tracking does not guarantee output invariance.

1.3 Notation

In this thesis, $\mathbb{R}$ denotes the set of real numbers and $\mathbb{Z}$ denotes the integers. We let $\mathrm{col}(x_1, \ldots, x_k) = \begin{bmatrix} x_1 & \cdots & x_k \end{bmatrix}^\top$ where $^\top$ denotes transpose. Given a positive real number L, the notation $\mathbb{R} \bmod L$ represents the real numbers modulo L. Given vectors $x, y \in \mathbb{R}^n$, we let $\langle x, y \rangle$ denote the Euclidean inner product and denote the associated Euclidean norm by $\|x\|$. We let I_n represent the $n \times n$ identity matrix and $0_{m\times n}$ represent the $m \times n$ zero matrix.

Given a function $\sigma : A \to B$, we let $\mathrm{Im}(\sigma)$ or $\sigma(A)$ denote its image, i.e., $\mathrm{Im}(\sigma) = \sigma(A) = \{b \in B : b = \sigma(a), \text{ for some } a \in A\}$. If M and N are two smooth manifolds and $F : M \to N$ is a map, we denote by $\mathrm{d}F_p$ the differential of F at $p \in M$. If M and N are open subsets of Euclidean spaces $\mathbb{R}^m$ and $\mathbb{R}^n$, respectively, then $\mathrm{d}F_p$ is the familiar derivative of F at p, whose matrix representation is the $n \times m$ Jacobian of F. In particular, if $\sigma : \mathbb{R}^n \to \mathbb{R}$ is a real-valued function then, depending on the context, $\mathrm{d}\sigma_x$ may represent

the differential map $\mathbb{R}^n \to \mathbb{R}$ or the row vector $[\partial_{x_1}\sigma \ \cdots \ \partial_{x_n}\sigma]$. On the other hand, we will denote by $\nabla\sigma(x)$ the column vector $\mathrm{d}\sigma_x^\top$.

If f, $g : \mathbb{R}^n \to \mathbb{R}^n$ are smooth vector fields and $\phi : \mathbb{R}^n \to \mathbb{R}^m$ is a smooth map, we use the following standard notation for iterated Lie derivatives $L_f^0\phi \coloneqq \phi$, $L_f^k\phi \coloneqq L_f(L_f^{k-1}\phi) = \langle \mathrm{d}L_f^{k-1}\phi_x, f(x)\rangle$, $L_gL_f\phi \coloneqq L_g(L_f\phi) = \langle \mathrm{d}L_f\phi_x, g(x)\rangle$. If $\phi : \mathbb{D} \to M$ is a smooth map between manifolds, with either $\mathbb{D} = \mathbb{R}$ or $\mathbb{D} = \mathbb{R} \bmod L$, and $\mathrm{d}/\mathrm{d}\lambda$ is the tangent vector to $\mathbb{D}$ at λ, we denote by $\phi'(\lambda) \coloneqq \mathrm{d}\phi_\lambda(\mathrm{d}/\mathrm{d}\lambda)$ the tangent vector at $\phi(\lambda)$, which we identify with a vector in $\mathbb{R}^m$, where $m = \dim M$, so that the function $\lambda \mapsto \phi'(\lambda)$ maps $\mathbb{D} \to \mathbb{R}^m$. Then, the second derivative $\phi''(\lambda)$ is also a function $\mathbb{D} \to \mathbb{R}^m$.

1.4 Problem Formulation

In this thesis we consider the problem of synchronizing two control systems. The first of these systems is of the form

$$\begin{aligned} \dot{x} &= f(x) + \sum_{i=1}^{m} g_i(x)u_i \\ y &= h(x) \end{aligned} \tag{1.1}$$

where $x \in \mathbb{R}^n$ is the state, $u \coloneqq \mathrm{col}\,(u_1, \ldots, u_m)$ is the control input, the vector fields $f, g_1, \ldots, g_m : \mathbb{R}^n \to \mathbb{R}^n$ are smooth. The smooth function $h : \mathbb{R}^n \to \mathbb{R}^p$ models the output, i.e., the variables we are interested in controlling.

The second system is an Euler-Lagrange system with an N-dimensional configuration space $\mathcal{Q}$ and m control inputs $\tau \in \mathbb{R}^m$. The model is given by

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial q} - \frac{\partial L}{\partial \dot{q}} = B(q)\tau$$

where $L(q, \dot{q})$ is the Lagrangian function. We assume that L is smooth and has the form $L(q, \dot{q}) = K(q, \dot{q}) - P(q)$ where $K(q, \dot{q}) = (1/2)\dot{q}^\top M(q)\dot{q}$ is the system's kinetic energy and $P : \mathcal{Q} \to \mathbb{R}$ is the system's potential energy. The inertia matrix $M(q)$ is positive definite for all q. Furthermore, $B : \mathcal{Q} \to \mathbb{R}^{N\times m}$ is smooth and full rank for all q. The system can be rewritten in the standard vector form

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = B(q)\tau \tag{1.2}$$

where $C(q, \dot{q}) \in \mathbb{R}^{N\times N}$ represents the centripetal and Coriolis terms and $G(q) = \nabla P(q) \in \mathbb{R}^N$ represents the gravitation effects [53].

The main ideas presented in this thesis are local and so henceforth we assume that $\mathcal{Q} \simeq \mathbb{R}^N$. Motivated by the notion of a task space in robotics, we assume that system (1.2) has a smooth output modelled as

$$Y = H(q), \qquad H : \mathcal{Q} \to \mathbb{R}^P. \tag{1.3}$$

1.4.1 Admissible Class of Paths

Consider a regular Jordan curve[1] $\mathcal{C}_u$ of length L_u in the output space of (1.1) with smooth parameterization $\sigma_u : \mathbb{R} \to \mathbb{R}^p$, $\sigma_u(\mathbb{R}) = \mathcal{C}_u$. Since σ_u is regular we assume, without loss of generality, that σ_u is a unit-speed parameterization, i.e., $\|\sigma_u'(\cdot)\| \equiv 1$. Under this assumption, σ_u is parameterized by its arc-length L_u, i.e., $\forall \lambda_u \in \mathbb{R}, \sigma_u(\lambda_u + L_u) = \sigma_u(\lambda_u)$. Similarly, consider a second regular Jordan curve $\mathcal{C}_r$ of length L_r in the output space of (1.2) with unit-speed parameterization $\sigma_r : \mathbb{R} \to \mathbb{R}^P$, $\sigma_r(\mathbb{R}) = \mathcal{C}_r$. The above discussion is summarized in the following assumption.

Assumption 1 (unit-speed parameterization)**.** The regular Jordan curves $\mathcal{C}_u \subset \mathbb{R}^p$, $\mathcal{C}_r \subset \mathbb{R}^P$ have smooth unit-speed parameterizations σ_u and σ_r respectively.

We also assume that the curves $\mathcal{C}_u$, $\mathcal{C}_r$ have implicit representations.

Assumption 2 (implicit representation)**.** The regular Jordan curves $\mathcal{C}_u \subset \mathbb{R}^p$, $\mathcal{C}_r \subset \mathbb{R}^P$ have implicit representations

$$\mathcal{C}_u = \{y \in W_u : s_u(y) = 0\}$$
$$\mathcal{C}_r = \{Y \in W_r : s_r(Y) = 0\}$$

where $s_u : W_u \subseteq \mathbb{R}^p \to \mathbb{R}^{p-1}$, $s_r : W_r \subseteq \mathbb{R}^P \to \mathbb{R}^{P-1}$, are smooth functions such that $\mathrm{d}s_u \neq 0$ on $\mathcal{C}_u$ and $\mathrm{d}s_r \neq 0$ on $\mathcal{C}_r$ and W_u, W_r are open sets containing $\mathcal{C}_u$ and $\mathcal{C}_r$ respectively.

Assumption 2 automatically holds if the parameterizations are regular of order p[2]. We show how to construct these functions in this special case in Chapter 4.

[1] We use the subscript u because in the subsequent discussion, system (1.1) will be a unicycle (differential drive) mobile robot.

[2] A parameterized curve $\sigma(\cdot)$ is regular of order p if the vectors $\sigma'(\lambda)$, $\sigma''(\lambda) \ldots$, $\sigma^{(p)}(\lambda)$ are linearly independent at each point $\sigma(\lambda)$ on the curve.

1.4.2 Design Objective

Our control objective is to design smooth state feedback control laws for the nonlinear system (1.1) and the Euler-Lagrange system (1.2) that make the closed-loop output of (1.1) approach the curve $\mathcal{C}_u$ and the closed-loop output (1.3) of (1.2) approach the curve $\mathcal{C}_r$. The motion of the two systems, when restricted to their assigned curves, must be synchronized. To make our notion of synchronization precise, we introduce a synchronization constraint.

Definition 1.4.1. A **path following synchronization constraint** for two curves $\mathcal{C}_u \subset \mathbb{R}^p$, and $\mathcal{C}_r \subset \mathbb{R}^P$ is a relation $F(y, Y) = 0$ where $F : \mathcal{C}_u \times \mathcal{C}_r \to \mathbb{R}^k$ is smooth, $\operatorname{rank}(\mathrm{d}F_{(y,Y)}) = k$ for each $(y, Y) \in F^{-1}(0)$.

We now introduce the synchronized path following problem considered in this thesis.

Synchronized path following problem : Given two regular Jordan curves $\mathcal{C}_u$ and $\mathcal{C}_r$ that satisfy Assumptions 1 and 2, and a path following synchronization constraint $F : \mathcal{C}_u \times \mathcal{C}_r \to \mathbb{R}^k$, find state feedback controllers $u : \mathbb{R}^n \to \mathbb{R}^m$, $\tau : \mathbb{R}^{2N} \to \mathbb{R}^N$ and two open sets of initial conditions $U \subseteq \mathbb{R}^n$, $U_c \times U_v \subseteq \mathbb{R}^N \times \mathbb{R}^N$ such that $\mathcal{C}_u \subset h(U)$, $\mathcal{C}_r \subset H(U_c)$, and the closed-loop systems satisfies

- **A** For each initial condition in U, and each initial condition in $U_c \times U_v$, the solution $x(t)$ to (1.1) and the solution $(q(t), \dot{q}(t))$ to (1.2) exist for all $t \geq 0$ and all solutions are such that $\|h(x(t))\|_{\mathcal{C}_u} \to 0$ and $\|H(q(t))\|_{\mathcal{C}_r} \to 0$ as $t \to \infty$.
- **I** The curves $\mathcal{C}_u$ and $\mathcal{C}_r$ are output invariant for the respective closed-loop systems.
- **S** For each initial condition in U, and each initial condition in $U_c \times U_v$, such that the solution $x(t)$ to (1.1) and the solution $(q(t), \dot{q}(t))$ to (1.2) satisfy $h(x(t)) \in \mathcal{C}_u$, $H(q(t)) \in \mathcal{C}_r$ for all $t \geq 0$, $F(h(x(t)), H(q(t))) \to 0$ as $t \to \infty$. Furthremore, if $F(h(x(0)), H(q(0))) = 0$ at $t = 0$, then $F(h(x(t)), H(q(t))) = 0$ for all $t \geq 0$.

•

Objective **A** in our synchronized path following problem asks that the assigned paths be attractive for the closed-loop systems. Objective **I** asks that the assigned paths be invariant in the sense that, if the closed-loop systems are properly initialized, the corresponding solutions evolve on their assigned paths. Finally, objective **S** asks that the systems synchronize their motion along the path. In practice, condition **S** should also hold for solutions that do not strictly evolve on the assigned curves, but rather, only asymptotically approach the curves. In this thesis we do not solve the synchronized path following problem in its full

generality. Specifically, we look at the special case when (i) system (1.1) is a differential drive robot (ii) system (1.2) is fully actuated ($m = N$) and non-redundant ($P = N$).

Our approach to solving this problem is to first stabilize the path following manifold [39] associated to each system. This achieves objectives **A** and **I** and is done using feedback linearization. It does not require any inter-system communication. Objective **S** is achieved by controlling the dynamics restricted to their respective path following manifolds. This aspect of the control law requires inter-system communication.

1.5 Organization and Contributions

In Chapter 2 transverse feedback linearization for a differential drive ground vehicle modeled as a unicycle is presented. A switching scheme is introduced for the unicycle to avoid singularities on its path. Necessary and sufficient conditions for applying the transverse feedback linearization approach for path following to Euler-Lagrange systems are summarized in Chapter 3. Chapter 4 presents a procedure to construct an implicit representation of a given parameterized curve. After fully feedback linearizing the unicycle and mechanical system, path following controllers are designed for each system in Chapter 5. Chapter 5 also investigates velocity and position synchronization for the unicycle and Euler-Lagrange systems when they are following their paths. An application of synchronized path following for a unicycle and a fully actuated 4 degree-of-freedom robotic manipulator are shown in Chapter 6 and experimental set up for implementing our control laws to laboratory equipment are discussed in Chapter 6. Simulations are also provided to illustrate the effectiveness of our control laws. The main contribution of this thesis is that it provides a solution to the synchronized path following problem that is applicable to a large class of Jordan curves. It also introduces a switching scheme that allows the unicycle to avoid controller singularities while synchronizing with other systems. Although this thesis does not contain experimental results, the framework for running these experiments has been realized. In particular, the software needed to communicate between robots has been created and tested.

Chapter 2

Dynamic Unicycle

In this chapter, we propose two coordinate and feedback transformations for a differential drive ground vehicle with a given Jordan curve in it's output space. These transformations will be used to simplify the design of synchronized path following controllers.

The first coordinate and feedback transformation is based on that used in [5] and [40]. In those papers the authors present transverse feedback linearization of a kinematic model of the unicycle. A limitation of that work is that the speed of the unicycle is fixed along its path. This means that synchronization along a path is impossible which motivates us to use a dynamic system model. This modification allows us to vary the motion of the robot while restricted to the desired path and hence achieve synchronization.

The author of [5] shows that the proposed feedback transformation is singular whenever the unicycle's forward velocity is zero. However, when synchronizing with other systems, it is possible that the unicycle needs to stop and change directions. To avoid singularities in these situations, we introduce a second coordinate and feedback transformation. A switching rule is used to switch between the two sets of coordinates.

2.1 Differential Drive Robot as a Dynamic Unicycle

We are motivated to study the control of a differential drive mobile robot by the availability the Chameleon R200 built by Clearpath Robotics, see Figure 2.1, at the University of Waterloo. A schematic illustration of a differential drive robot on the plane is shown in Figure 2.2.

Figure 2.1: Photograph of the differential drive robot Chameleon R200.

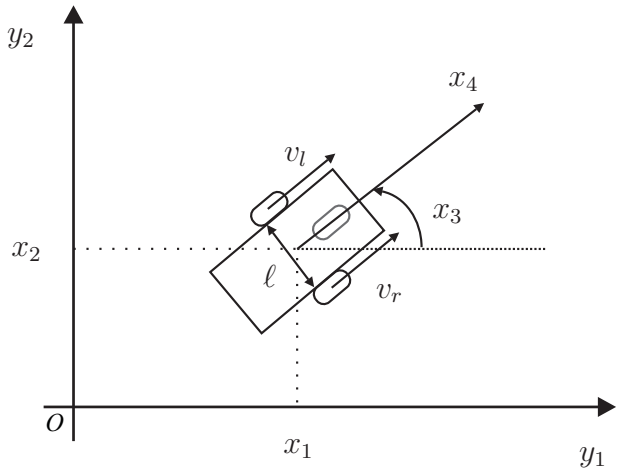


Figure 2.2: Schematic diagram illustrating the states of the dynamic model of unicycle.

The movement of a differential drive ground vehicle is determined by the two sperate wheels placed on either side of the vehicle. Let v_l and v_r denote the left-wheel velocity and right-wheel velocity of a differential drive robot and v and ω represent the translational velocity and rotational velocity of a unicycle model. Without loss of generality, we assume ω is positive when the unicycle is rotating in the counter-clockwise direction, then the relationship between v_l, v_r, v and ω is

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{\ell} & -\frac{1}{\ell} \end{bmatrix} \begin{bmatrix} v_l \\ v_r \end{bmatrix}.$$

Here ℓ is the effective track of the vehicle shown in Figure 2.2. Next, we introduce the *dynamic* model of the unicycle. In Figure 2.2, the variables x_1, x_2, represent the positions of the unicycle on the plane, x_3 is the unicycle's orientation while x_4 is its translational ve-

locity. Define $x \coloneqq \mathrm{col}(x_1, x_2, x_3, x_4)$ and take the translational acceleration and rotational velocity as the control inputs. Then the dynamic unicycle takes the form (1.1) with

$$\dot{x} = \begin{bmatrix} x_4 \cos x_3 \\ x_4 \sin x_3 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} u_2. \tag{2.1}$$

Remark 2.1.1. The system (2.1) is referred as a dynamic model of the unicycle robot because we take the translational acceleration and rotational velocity as the control inputs. The kinematic model of the unicycle can be found in [5] in which the control input is taken as the translational and rotational velocities. In the lab, we are able to command the left and right velocities, i.e., v_l and v_r, of the robot Chameleon R200. To implement our control design of u_1 and u_2 experimentally, we will have to generate the state x_4 in software by integration.

Let

$$f_u(x) \coloneqq \begin{bmatrix} x_4 \cos x_3 \\ x_4 \sin x_3 \\ 0 \\ 0 \end{bmatrix}, \qquad g_u^1(x) \coloneqq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \qquad g_u^2(x) \coloneqq \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

so that system (2.1) can be written compactly in the form of (1.1) as

$$\dot{x} = f_u(x) + g_u^1(x)u_1 + g_u^2(x)u_2.$$

We take the position of the unicycle in the plane as its output (2.1)

$$y_u = h_u(x) = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}. \tag{2.2}$$

2.2 Coordinate Transformation: Unicycle

Consider a regular Jordan curve $\mathcal{C}_u$ of length L_u in the output space of the unicycle (2.1) that satisfies Assumptions 1 and 2. Denote the smooth parameterization of $\mathcal{C}_u$ by

$$\sigma_u : \mathbb{R} \to \mathbb{R}^2, \qquad \lambda_u \mapsto \begin{bmatrix} \sigma_u^1(\lambda_u) \\ \sigma_u^2(\lambda_u) \end{bmatrix}.$$

By Assumption 2 there exists a smooth function $s_u : W_u \subseteq \mathbb{R}^2 \to \mathbb{R}$ with W_u an open set such that $\mathcal{C}_u \subset W_u$ and such that

$$\mathcal{C}_u = \{y_u \in W_u : s_u(y_u) = 0\}.$$

The *lift* of $\mathcal{C}_u$ to the state space of the unicycle is the set of $x \in \mathbb{R}^4$ such that the associated output $h(x)$ lies on the desired curve $\mathcal{C}_u$. The lift is given by

$$\Gamma_u \coloneqq (s_u \circ h_u)^{-1}(0) = \left\{x \in U \subseteq \mathbb{R}^4 : s_u\left(h_u(x)\right) = 0\right\}$$

where $U = h_u^{-1}(W_u)$. The set Γ_u is a two-dimensional embedded submanifold of $\mathbb{R}^4$ if $\operatorname{rank}\left(\mathrm{d}(s_u \circ h_u)_x\right) = 1$ at each $x \in \Gamma_u$. Using the chain rule

$$\mathrm{d}(s_u \circ h_u)_x = \mathrm{d}s_u|_{y_u = h_u(x)}\, \mathrm{d}h_u|_x = \left[\begin{array}{cccc} \frac{\partial s_u}{\partial y_u^1} & \frac{\partial s_u}{\partial y_u^2} & 0 & 0 \end{array}\right]_{y_u = h_u(x)}.$$

By Assumption 2, this differential is non-zero on $\mathcal{C}_u$ which shows that Γ_u is a submanifold of $\mathbb{R}^4$. Intuitively, the position of the unicycle approaches to its path $\mathcal{C}_u$ if the state x approaches Γ_u. Define $\alpha_u : U \subseteq \mathbb{R}^4 \to \mathbb{R}$ as $\alpha_u(x) = s_u \circ h_u(x)$. The path following manifold for the unicycle with respect to $\mathcal{C}_u$ is the largest controlled invariant subset of Γ_u [39]. This set is denoted by $\Gamma_u^\star$ and is given by

$$\begin{aligned} \Gamma_u^\star \coloneqq & \{x \in U : \alpha_u(x) = \dot{\alpha}_u(x) = 0\} \\ = & \{x \in U : \alpha_u(x) = \langle \mathrm{d}\alpha_u(x),\ f_u(x)\rangle = 0\} \\ = & \{x \in U : \alpha_u(x) = L_{f_u}\alpha_u(x) = 0\}. \end{aligned}$$

The function $\alpha_u(x)$ yields a well-defined relative degree of 2 at each point on $\Gamma_u^\star$ so long as the unicycle's forward velocity is non-zero, i.e., $x_4 \neq 0$. Hence, in a neighbourhood of such points, we can use the function α_u to feedback linearize that portion of the unicycle dynamics that determines whether or not the system is approaching $\Gamma_u^\star$. We seek a second function so that this new function, together with α_u, yields a relative degree of $\{2, 2\}$ on $\Gamma_u^\star$. In this way we can also feedback linearize the dynamics that determine the motion on the set $\Gamma_u^\star$ and hence on the path $\mathcal{C}_u$.

To this end, let $\mathcal{C}_u^\varepsilon$ denote a tubular neighbourhood of path $\mathcal{C}_u$. The tubular neighbourhood has the property that if $y_u \in \mathcal{C}_u^\varepsilon$ then there is a unique $y_u^\star \in \mathcal{C}_u$ that minimizes the distance form y_u to $\mathcal{C}_u$. Define the map

$$\begin{aligned} \varpi_u : \mathcal{C}_u^\varepsilon & \to \mathbb{R} \\ y_u & \mapsto \arg \min_{\lambda_u \in [0, L_u)} \left\| y_u - \sigma_u(\lambda_u) \right\|. \end{aligned} \tag{2.3}$$

Let $\lambda_u^\star \in \mathbb{R}$ be such that $y_u^\star = \sigma_u(\lambda_u^\star)$. Intuitively, the map (2.3) returns the distance, *along the path*, to the closest point $\sigma_u(\lambda_u^\star) = \sigma_u(\varpi_u(y_u))$, see Figure 2.3.

Using (2.3) let

$$\begin{aligned} \pi_u : & V \subseteq \mathbb{R}^4 \to [0, L_u) \subset \mathbb{R} \\ & x \mapsto \varpi_u \circ h_u(x) \end{aligned} \tag{2.4}$$

where $V \coloneqq h_u^{-1}(\mathcal{C}_u^\varepsilon)$.

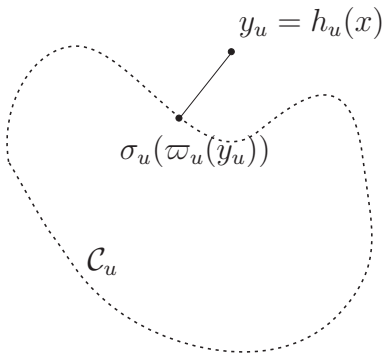


Figure 2.3: The map (2.3) returns the path parameter that, when σ_u is evaluated there, minimizes the distance from the unicycle's position to the desired path.

Using the functions α_u and π_u, we define a "path following output" for the unicycle as

$$\hat{y}_u \coloneqq \begin{bmatrix} \pi_u(x) \\ \alpha_u(x) \end{bmatrix} = \begin{bmatrix} \varpi_u \circ h_u \\ s_u \circ h_u \end{bmatrix}. \tag{2.5}$$

Roughly speaking, $\alpha_u(x)$ quantifies the distance of unicycle from the path and $\pi_u(x)$ represents the arc-length from an arbitrarily chosen starting point (i.e. $\lambda_u = 0$) to the closest point (i.e. $\lambda_u = \lambda_u^\star$) along the path to y_u.

Lemma 2.2.1. The dynamic unicycle (2.1) with output (2.5) yields a well defined vector relative degree of $\{2, 2\}$ at each point on $\Gamma_u^\star$ where $x_4 \neq 0$.

Proof. By the definition of relative degree [28], system (2.1) with output (2.5) has relative degree $\{2, 2\}$ at $x^\star \in \Gamma_u^\star$ if and only if

1. $L_{g_1}\pi_u(x)$, $L_{g_2}\pi_u(x)$, $L_{g_1}\alpha_u(x)$ and $L_{g_2}\alpha_u(x)$ are identically equal to zero for all x in a neighbourhood of $x^\star$.

2. The decoupling matrix associated to the output (2.5)

$$D_u(x) \coloneqq \begin{bmatrix} L_{g_1} L_{f_u} \pi_u(x) & L_{g_2} L_{f_u} \pi_u(x) \\ L_{g_1} L_{f_u} \alpha_u(x) & L_{g_2} L_{f_u} \alpha_u(x) \end{bmatrix}$$

is non-singular at $x^\star$.

By direct calculations, it easy to check that, for all x in a neighbourhood of $x^\star$,

$$L_{g_1}\pi_u(x) = L_{g_2}\pi_u(x) = L_{g_1}\alpha_u(x) = L_{g_2}\alpha_u(x) \equiv 0$$

and that the entries in of the decoupling matrix are

$$\begin{aligned}
L_{g_1} L_{f_u} \pi_u(x) &= \tfrac{1}{\|\sigma_u'(\lambda_u^\star)\|} \left(\sigma_u^{1\prime}(\lambda_u^\star) \cos(x_3) + \sigma_u^2(\lambda_u^\star) \sin(x_3) \right) \\
L_{g_2} L_{f_u} \pi_u(x) &= \tfrac{1}{\|\sigma_u'(\lambda_u^\star)\|} \left(\sigma_u^{2\prime}(\lambda_u^\star) (x_4 \cos(x_3)) - \sigma_u^{1\prime}(\lambda_u^\star) (x_4 \sin(x_3)) \right) \\
L_{g_1} L_{f_u} \alpha_u(x) &= \tfrac{\partial \alpha_u}{\partial x_1} \cos(x_3) + \tfrac{\partial \alpha_u}{\partial x_2} \sin(x_3) \\
L_{g_2} L_{f_u} \alpha_u(x) &= -\tfrac{\partial \alpha_u}{\partial x_1} x_4 \sin(x_3) + \tfrac{\partial \alpha_u}{\partial x_2} x_4 \cos(x_3)
\end{aligned}$$

where $\lambda_u^\star = \pi_u(x)$. The determinant of $D_u(x)$ is

$$\det(D_u(x)) = \frac{x_4}{\|\sigma_u'(\lambda_u^\star)\|} \left(\sigma_u^{1\prime}(\lambda_u^\star) \frac{\partial \alpha_u}{\partial x_1} - \sigma_u^{2\prime}(\lambda_u^\star) \frac{\partial \alpha_u}{\partial x_2} \right).$$

The determinant goes to 0 if either (i) $x_4 = 0$ (ii) $\sigma_u^{1\prime}(\lambda_u^\star)\frac{\partial \alpha_u}{\partial x_1} - \sigma_u^{2\prime}(\lambda_u^\star)\frac{\partial \alpha_u}{\partial x_2} = 0$. The first condition is ruled out by assumption. We now show that the second condition does not hold because the vectors $\operatorname{col}\left(\frac{\partial \alpha_u}{\partial x_1}, \frac{\partial \alpha_u}{\partial x_1}\right)$ and $\sigma_u'(\lambda_u^\star)$ are orthogonal at $x^\star \in \Gamma_u^\star$. To begin with, using the chain rule the vector $\operatorname{col}\left(\frac{\partial \alpha_u}{\partial x_1}, \frac{\partial \alpha_u}{\partial x_1}\right)$ can be written

$$\left[\begin{smallmatrix} \frac{\partial \alpha_u}{\partial x_1} \\ \frac{\partial \alpha_u}{\partial x_2} \end{smallmatrix} \right]\Bigg|_{x=x^\star} = \left[\begin{smallmatrix} \frac{\partial s_u}{\partial y_u^1} \\ \frac{\partial s_u}{\partial y_u^2} \end{smallmatrix} \right]\Bigg|_{y_u = h_u(x^\star)} = \nabla s_u(h_u(x^\star)).$$

The tangent vector $\sigma_u'(\lambda_u^\star)$ is unit-length since σ_u is unit-speed parameterized. By the Assumption 2, the Jacobian $\mathrm{d}s_u$ is full rank, i.e., non-zero, at $\sigma_u(\lambda_u^\star)$. Hence the gradient, $\nabla s_u = \mathrm{d}s_u^\top$ is orthogonal to $s_u^{-1}(0)$ as shown in Figure 2.4. Therefore, the vectors $\operatorname{col}\left(\frac{\partial \alpha_u}{\partial x_1}, \frac{\partial \alpha_u}{\partial x_1}\right)$ and $\sigma_u'(\lambda_u^\star)$ are orthogonal as claimed. □

A consequence of Lemma 2.2.1 and [28, Propostion 5.1.2] is that we can use the virtual output (2.5) and its derivatives to define a local diffeomorphism in a neighbourhood of any point on $\Gamma_u^\star$ on which $x_4 \neq 0$.

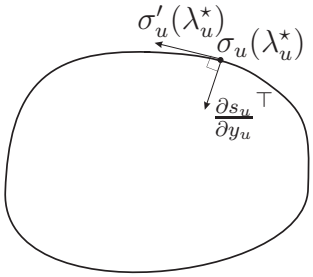


Figure 2.4: Geometric illustration of $\frac{\partial s_u}{\partial y_u}^\top$ and $\sigma_u'(\lambda_u^\star)$.

Corollary 2.2.2. Let $x^\star \in \Gamma_u^\star \setminus \{x \in \mathbb{R}^4 : x_4 = 0\}$, there exists a neighbourhood $U \in \mathbb{R}^4$ containing $x^\star$ such that the mapping $T_u : U \subset \mathbb{R}^4 \to T_u(U) \subset \mathbb{R}^4$,

$$\begin{bmatrix} \eta_1^u \\ \eta_2^u \\ \xi_1^u \\ \xi_2^u \end{bmatrix} = T_u(x) = \begin{bmatrix} \pi_u(x) \\ L_{f_u}\pi_u(x) \\ \alpha_u(x) \\ L_{f_u}\alpha_u(x) \end{bmatrix} \tag{2.6}$$

is a diffeomorphism onto its image.

Using the coordinate transformation (2.6), in a neighbourhood of any point $x^\star \in \Gamma_u^\star$ the system (2.1) in (η^u, ξ^u)-coordinates reads

$$\begin{aligned} \dot\eta_1^u &= \eta_2^u \\ \dot\eta_2^u &= L_{f_u}^2\pi + L_{g1}L_{f_u}\pi_u u_1 + L_{g2}L_{f_u}\pi_u u_2 \,\big|_{x=T^{-1}(\eta^u,\xi^u)} \\ \dot\xi_1^u &= \xi_2^u \\ \dot\xi_2^u &= L_{f_u}^2\alpha_u + L_{g1}L_{f_u}\alpha_u u_1 + L_{g2}L_{f_u}\alpha_u u_2 \,\big|_{x=T^{-1}(\eta^u,\xi^u)} . \end{aligned}$$

Consider the regular feedback transformation

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = D_u(x)^{-1}\left(\begin{bmatrix} -L_{f_u}^2\pi_u \\ -L_{f_u}^2\alpha_u \end{bmatrix} + \begin{bmatrix} v_u^{\parallel} \\ v_u^{\pitchfork} \end{bmatrix}\right) \tag{2.7}$$

where $(v_u^{\parallel}, v_u^{\pitchfork}) \in \mathbb{R}^2$ are auxiliary control inputs. By Lemma 2.2.1 this controller is well defined in a neighbourhood of $x^\star \in \Gamma_u^\star \setminus \{x \in \mathbb{R}^4 : x_4 = 0\}$. Thus in a neighbourhood of $x^\star$,

the unicycle system is feedback equivalent to

$$\begin{aligned}\dot{\eta}_1^u &= \eta_2^u \\ \dot{\eta}_2^u &= v_u^{\parallel} \\ \dot{\xi}_1^u &= \xi_2^u \\ \dot{\xi}_2^u &= v_u^{\pitchfork}.\end{aligned} \tag{2.8}$$

The ξ^u-subsystem is called transversal subsystem and the states ξ^u the transversal states. We call the η^u-subsystem the tangential subsystem and the states η^u the tangential states. The normal form (2.8) is useful for synchronized path following control design. This is because the transversal and tangential subsystems are decoupled. Hence we can design the transversal controller $v_u^{\pitchfork}$ to stabilize the transversal subsystem and drive the unicycle to the path. The tangential control input $v_u^{\parallel}$ can be used to achieve synchronization while the motion of the system is restricted to the path. The overall control scheme is presented in Figure 2.5. The design of the blocks labelled "Transversal controller" and "Tangential controller" is discussed in Chapter 5.

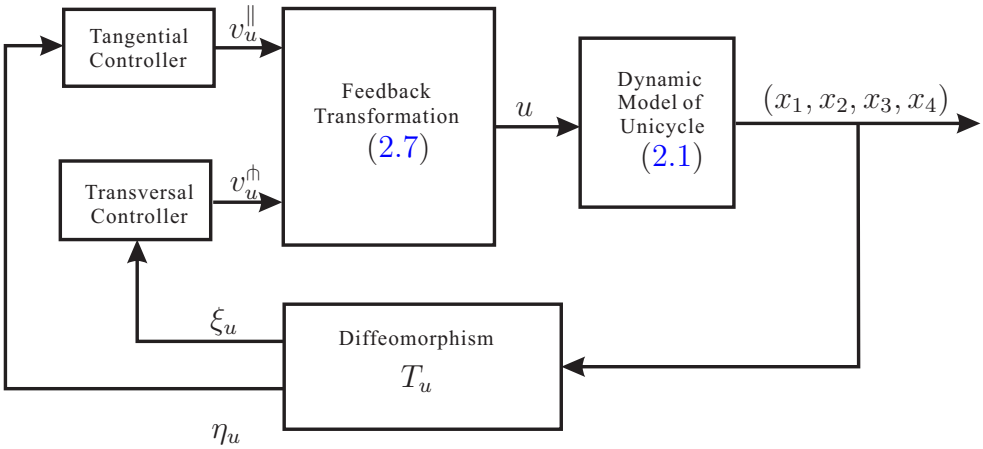


Figure 2.5: Diagram of the transverse feedback linearization of the dynamic unicycle.

2.2.1 Switching Through Singularities

Lemma 2.2.1 shows that the relative degree of unicycle system is not well-defined when $x_4 = 0$, meaning that the feedback transformation (2.7) is not applicable when the unicycle

stops. Since our synchronization is not restricted to leader-follower type of synchronization, the unicycle has to stop and even turn around in some cases to synchronize with the Euler-Lagrange system. We introduce a switching scheme applied in a neighbourhood of $x_4 = 0$ to avoid the singularities caused by our synchronization control laws. To do this, we partition the state space of the unicycle into two regions [61]. Let $\delta > 0$ be a fixed constant and define $M_0 := \{x \in \mathbb{R}^4 : |x_4| \leq \delta\}$ and $M := \{x \in \mathbb{R}^4 : |x_4| > \delta\}$. In region M, we can apply the control law (2.7). In region M_0, we introduce a new coordinate transformation $T^{M_0} : U \subset \mathbb{R}^4 \to T^{M_0}(U) \subset \mathbb{R}^4$ here U is a neighbourhood of $x^\star \in \Gamma_u^\star$, defined by

$$\begin{bmatrix} \eta_1^u \\ \eta_2^u \\ l \\ \tilde{\theta} \end{bmatrix} = T^{M_0}(x) = \begin{bmatrix} \pi_u(x) \\ L_{f_u}\pi_u(x) \\ \left[R_{\frac{\pi}{2}} \frac{\sigma'(\lambda_u^\star)}{\|\sigma'(\lambda_u^\star)\|}\right]^\top (h_u(x) - \sigma_u(\lambda_u^\star)) \\ x_3 - \operatorname{atan}\left\{\frac{\sigma_u^{2\prime}(\lambda_u^\star)}{\sigma_u^{1\prime}(\lambda_u^\star)}\right\} \end{bmatrix}$$

where $R_{\frac{\pi}{2}}$ is the $\frac{\pi}{2}$ counter-clockwise constant rotation matrix

$$R_{\frac{\pi}{2}} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}.$$

The states l and $\tilde{\theta}$ physically model the signed distance from the unicycle to its path and orientation error respectively, see Figure 2.6. Here $\sigma_u(\lambda_u^\star)$ denotes the projection of the

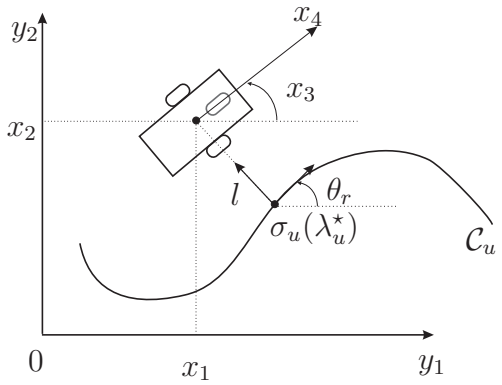


Figure 2.6: Physical illustration of states l and $\tilde{\theta}$.

unicycle to the closest point on the path $\mathcal{C}_u$ and θ_r denotes the reference orientation at point $\sigma_u(\lambda_u^\star)$. In other words $\tilde{\theta}$ is the error between the unicycle's heading x_3 and the angle θ_r that $\sigma_u'(\lambda_u^\star)$ makes with the y_1-axis.

Proposition 2.2.3. For every $x^\star \in \Gamma_u^\star$, there exists a neighbourhood U containing $x^\star$ such that T^{M_0} is a local diffeomorphism.

Proof. The Jacobian of T^{M_0} evaluated at $x = x^\star$ is

$$\mathrm{d}T^{M_0}_{x^\star} = \begin{bmatrix} \frac{\partial \pi_u}{\partial x_1} & \frac{\partial \pi_u}{\partial x_2} & 0 & 0 \\ \frac{\partial L_{f_u}\pi_u}{\partial x_1} & \frac{\partial L_{f_u}\pi_u}{\partial x_2} & \frac{\partial L_{f_u}\pi_u}{\partial x_3} & \frac{\partial L_{f_u}\pi_u}{\partial x_4} \\ \frac{\partial l}{\partial x_1} & \frac{\partial l}{\partial x_2} & 0 & 0 \\ \frac{\partial \theta}{\partial x_1} & \frac{\partial \theta}{\partial x_2} & 1 & 0 \end{bmatrix}.$$

To show that this is non-singular, we argue that rows one and three are linearly independent and that the (2, 4)-entry is non-zero.

We start by showing that $\frac{\partial L_f \pi}{\partial x_4} \neq 0$ on $\Gamma_u^\star$. To this end, note that

$$\frac{\partial L_{f_u}\pi_u}{\partial x_4} = L_{g_1}L_{f_u}\pi_u = \frac{1}{\|\sigma_u'(\lambda_u^\star)\|}\left(\sigma_u^{1\prime}(\lambda_u^\star)\cos(x_3) + \sigma_u^{2\prime}(\lambda_u^\star)\sin(x_3)\right)$$

where $\lambda_u^\star = \varpi_u(y_u)$ and ϖ_u was defined in (2.3) . On the $\Gamma_u^\star$, since $\xi_1^u = \xi_2^u = 0$, we have that $\sigma_u^1(\lambda_u^\star)$ and $\sigma_u^2(\lambda_u^\star)$ represent the position of unicycle and the orientation of unicycle is tangent to the path. These facts imply

$$\frac{\sigma_u^{2\prime}(\lambda_u^\star)}{\sigma_u^{1\prime}(\lambda_u^\star)} = \tan(x_3).$$

On the other hand, $L_{g_1}L_{f_u}\pi_u = 0$ if and only if

$$\frac{\sigma_u^{2\prime}(\lambda_u^\star)}{\sigma_u^{1\prime}(\lambda_u^\star)} = -\cot(x_3).$$

Since the solution set of $\tan(x_3) = -\cot(x_3)$ for x_3 is empty, $L_{g_1}L_{f_u}\pi_u \neq 0$ for all $x^\star \in \Gamma_u^\star$. Next, we show that rows one and three are linearly independent. To begin with, the functions $\pi_u(x)$ and $l(x)$ can be written as compositions

$$\pi_u(x) = \varpi_u \circ h_u(x), \qquad l(x) = \varphi \circ h_u(x)$$

where $\varphi : \mathbb{R}^2 \to \mathbb{R}$ is given by

$$\varphi(y_u) = \left[R_{\frac{\pi}{2}}\frac{\sigma'(\lambda_u^\star)}{\|\sigma'(\lambda_u^\star)\|}\right]^\top (y_u - \sigma_u(\lambda_u^\star)).$$

Applying the chain rule

$$\mathrm{d}(\pi_u)_x = \mathrm{d}(\varpi_u)_{h(x)}\mathrm{d}h_x = \left[\begin{array}{cccc} \frac{\partial \varpi_u}{\partial y_u^1} & \frac{\partial \varpi_u}{\partial y_u^2} & 0 & 0 \end{array}\right]_{y_u = h_u(x)},$$
$$\mathrm{d}l_x = \mathrm{d}\varphi_{h(x)}\mathrm{d}h_x = \left[\begin{array}{cccc} \frac{\partial \varphi_u}{\partial y_u^1} & \frac{\partial \varphi_u}{\partial y_u^2} & 0 & 0 \end{array}\right]_{y_u = h_u(x)}$$

and so it is sufficient to show that $\mathrm{d}(\varpi_u)_{y_u}$ and $\mathrm{d}\varphi_{y_u}$ are linearly independent.

At the point $\sigma_u(\lambda_u^\star)$, the vector $\nabla\varpi_u = \mathrm{d}(\varpi_u)_{y_u}^\top$ is tangent to the curve $\mathcal{C}_u$. Meanwhile, the vector $\nabla\varphi = \mathrm{d}\varphi_{y_u}^\top$ is normal to the curve. Following [16, Lemma 3.2] and calculations on Page 14 from [16], we have

$$\mathrm{d}(\varpi_u)_{y_u}^\top = \sigma_u'(\lambda_u^\star)$$
$$\mathrm{d}\varphi_{y_u}^\top = R_{\frac{\pi}{2}}\frac{\sigma'(\lambda_u^\star)}{\|\sigma'(\lambda_u^\star)\|}$$

for all $y_u \in \mathcal{C}_u$. Therefore, since σ_u is a regular curve, $\sigma_u'(\lambda_u^\star)$ is non-zero and hence $\mathrm{d}(\varpi_u)_{y_u}^\top$ is non-zero and so too is $\mathrm{d}\varphi_{y_u}^\top$. □

The unicycle in $\left(\eta_1^u, \eta_2^u, l, \tilde{\theta}\right)$-coordinates takes the form [18]

$$\begin{aligned}
\dot{\eta}_1^u &= \eta_2^u \\
\dot{\eta}_2^u &= L_{f_u}^2\pi_u + L_{g_1}L_{f_u}\pi_u u_1 + L_{g_2}L_{f_u}\pi_u u_2\big|_{x=T^{M_0}{}^{-1}\left(\eta^u,l,\tilde{\theta}\right)} \\
\dot{l} &= x_4\sin\left(\tilde{\theta}\right)\Big|_{x=T^{M_0}{}^{-1}\left(\eta^u,l,\tilde{\theta}\right)} \\
\dot{\tilde{\theta}} &= u_2 - x_4\cos\left(\tilde{\theta}\right)\frac{\kappa\left(\eta_1^u\right)}{1-\kappa\left(\eta_1^u\right)l}\Big|_{x=T^{M_0}{}^{-1}\left(\eta^u,l,\tilde{\theta}\right)}
\end{aligned}$$

where $\kappa(\eta_u)$ is the curvature function of the path σ_u. Consider the feedback transformation

$$\left[\begin{array}{c} u_1 \\ u_2 \end{array}\right] = \left[\begin{array}{cc} L_{g_1}L_{f_u}\pi_u & L_{g_2}L_{f_u}\pi_u \\ 0 & 1 \end{array}\right]^{-1}\left(\left[\begin{array}{c} -L_{f_u}^2\pi_u \\ x_4\cos\left(\tilde{\theta}\right)\frac{\kappa\left(\eta_1^u\right)}{1-\kappa\left(\eta_1^u\right)l} \end{array}\right] + \left[\begin{array}{c} v_u^{\|} \\ u_0 \end{array}\right]\right) \tag{2.9}$$

Remark 2.2.4. In this chapter, we assume $y_u \in \mathcal{C}_u^\varepsilon$. If this assumption holds, then $\kappa(\eta_1^u) \neq \frac{1}{l}$. For instance, consider the simplest case that a planar unit circle centered at the origin of the plane is the desired path for the unicycle, then $\mathcal{C}_u^\varepsilon$ is defined as $\mathcal{C}_u^\varepsilon = \{y_u \in \mathbb{R}^2 : y_u \neq \left[\begin{array}{cc} 0 & 0 \end{array}\right]^\top\}$. In this case, $\kappa(\eta_1^u) \equiv 1$ and $\kappa(\eta_1^u) = \frac{1}{l}$ only happens when the unicycle is located at the center of the circle, i.e., $y_u = \left[\begin{array}{cc} 0 & 0 \end{array}\right]^\top$, which contradicts the assumption that $y_u \in \mathcal{C}_u^\varepsilon$.

Proposition 2.2.5. The feedback transformation (2.9) is well defined at every $x^\star \in \Gamma_u^\star$.

Proof. Let

$$D^{M_0}(x) \coloneqq \begin{bmatrix} L_{g_1} L_{f_u} \pi_u(x) & L_{g_2} L_{f_u} \pi_u(x) \\ 0 & 1 \end{bmatrix}.$$

The determinant of $D^{M_0}(x)$ is $L_{g_1} L_{f_u} \pi_u(x)$ which, according to the proof of Proposition 2.2.3, $L_{g_1} L_{f_u} \pi_u$ is non-zero for all $x^\star \in \Gamma_u^\star$. □

Under the feedback transformation (2.9), the unicycle reads as

$$\begin{aligned} \dot{\eta}_1^u &= \eta_2^u \\ \dot{\eta}_2^u &= v_u^{\parallel} \\ \dot{l} &= x_4 \sin\left(\tilde{\theta}\right)\Big|_{x = T^{M_0}{}^{-1}\left(\eta^u, l, \tilde{\theta}\right)} \\ \dot{\tilde{\theta}} &= u_0 \end{aligned} \tag{2.10}$$

By the physical definition of states $l, \tilde{\theta}$, we have $l|_{\Gamma_u^\star} = \tilde{\theta}\Big|_{\Gamma_u^\star} = 0$. Thus, stabilizing the ξ_u-subsystem in (2.8) is equivalent to stabilizing the $(l, \tilde{\theta})$-subsystem in (2.10). The discussion of path following controller design for $v_u^{\pitchfork}$ and u_0 and the switching scheme between feedback transformations (2.7) and (2.9) are presented in Section 5.1. Note that there is no change in the tangential dynamics in the new coordinates $(\eta_1^u, \eta_2^u, l, \tilde{\theta})$. The overall diagram of the feedback linearization of the unicycle (2.1) in the singular region M_0 is shown in Figure 2.7.

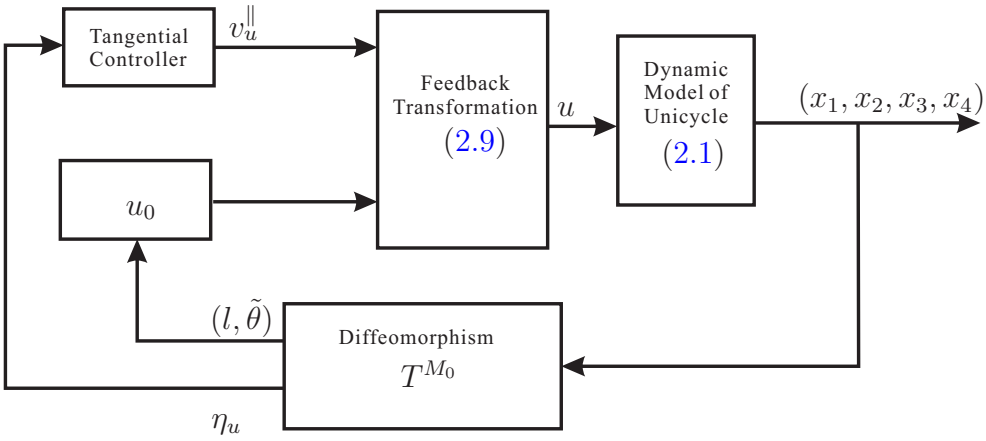


Figure 2.7: Feedback linearization of the unicycle in its singular region M_0.

Chapter 3

Euler-Lagrange Systems

In this chapter, we aim to use the transverse feedback linearization technique used in Chapter 2 to feedback linearize a fully actuated N degree-of-freedom mechanical system. With sufficient actuation and a regularity assumptions, the dynamics along the path can be represented in a normal form that simplifies the control design to solve the synchronized path following problem. This chapter is based on the work in [25].

3.1 Modeling

Recall the equations of motion (1.2) for a fully-actuated N degree-of-freedom mechanical system discussed in Section 1.4. Following [26], let

$$\begin{aligned} x_c &\coloneqq \operatorname{col}(x_{c1}, \cdots, x_{cN}) \coloneqq q \\ x_v &\coloneqq \operatorname{col}(x_{v1}, \cdots, x_{vN}) \coloneqq \dot{q} \end{aligned}$$

and $x = (x_c, x_v) \in \mathbb{R}^N \times \mathbb{R}^N$ so that the Euler-Lagrange equations can be locally expressed in the form (1.1)

$$\dot{x} = \begin{bmatrix} x_v \\ f_v(x) \end{bmatrix} + \begin{bmatrix} 0_{N\times m} \\ g_v(x_c) \end{bmatrix} u \tag{3.1}$$

where $f_v(x)$ equals $-M^{-1}(x_c)(C(x_c, x_v) + G(x_c))$ and $g_v(x_c)u$ equals $M^{-1}(x_c)B(x_c)\tau$. Define

$$f_r(x) \coloneqq \begin{bmatrix} x_v \\ f_v(x) \end{bmatrix}, \qquad g_r(x_c) \coloneqq \begin{bmatrix} 0 \\ g_v(x_c) \end{bmatrix}$$

so that system (3.1) can be written as

$$\dot{x} = f_r(x) + g_r(x_c)\tau.$$

Definition 3.1.1 ([25])**.** An Euler-Lagrange system (1.2) with m inputs and N degrees of freedom is said to be

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

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

As discussed in Section 1.4, we restrict the output of (3.1) to be a function of the configuration variables $x_c = q$. For notational convenience, we re-write the output (1.3) as

$$y_r = h_r(x_c) \tag{3.2}$$

where $h_r(x_c) \coloneqq H(q)$ and $y_r \coloneqq Y \in \mathbb{R}^p$.

Definition 3.1.2 ([51])**.** An Euler-Lagrange system (1.2) with output (1.3) is **kinematically redundant** if $N > p$.

For instance, a 3-link planar manipulator has 3 degrees of freedom, i.e., $N = 3$. It is kinematically redundant if its endpoint is restricted to moving on a plane, i.e., $p = 2$. If the endpoint is allowed to move in $\mathbb{R}^3$ space, then it is not kinematically redundant since $N = p = 3$.

3.2 Path Following Normal Form

Let $\mathcal{C}_r$ be a given regular Jordan curve that satisfies Assumptions 1 and 2. By Assumption 2 there exists a smooth function $s_r : W_r \subseteq \mathbb{R}^p \to \mathbb{R}^{p-1}$ with W_r an open set such that $\mathcal{C}_r \subset W_r$ and such that

$$\mathcal{C}_r = \{y_r \in W_r : s_r(y_r) = 0\}.$$

The *lift* of $\mathcal{C}_r$ to the state space of the mechanical system is the set of $x \in \mathbb{R}^{2N}$ such that the associated output $h_r(x_c)$ lies on the desired curve $\mathcal{C}_r$. The lift is given by

$$\Gamma_r \coloneqq (s_r \circ h_r)^{-1}(0) = \left\{(x_c, x_v) \in U \times \mathbb{R}^N : s_r\left(h_r(x_c)\right) = 0\right\}$$

where $U = h_r^{-1}(W_r)$. Define $\alpha_r : U \times \mathbb{R}^N \to \mathbb{R}^{p-1}$ as $\alpha_r(x_c) = \mathrm{col}\left(\alpha_r^1(x_c), \cdots, \alpha_r^{p-1}(x_c)\right) \coloneqq s_r \circ h_r(x_c) = \mathrm{col}\left(s_r^1 \circ h_r(x_c), \cdots, s_r^{p-1} \circ h_r(x_c)\right)$. The path following manifold for the mechanical system with respect to $\mathcal{C}_u$ is the largest controlled invariant subset of Γ_r [39]. This set is denoted by $\Gamma_r^\star$ and is given by

$$\begin{aligned}\Gamma_r^\star &\coloneqq \left\{(x_c, x_v) \in U \times \mathbb{R}^N : \alpha_r = \dot{\alpha}_r = 0\right\} \\ &= \left\{(x_c, x_v) \in U \times \mathbb{R}^N : \alpha_r(x_c) = \langle \mathrm{d}\alpha_r(x_c),\ x_v\rangle = 0\right\} \\ &= \left\{(x_c, x_v) \in U \times \mathbb{R}^N : \alpha_r(x_c) = L_{f_r}\alpha_r(x_c) = 0\right\}.\end{aligned}$$

The function $\alpha_r(x_c)$ yields a well-defined vector of relative degree $\{r_1, \cdots, r_{p-1}\} = \{2, \cdots, 2\}$ at each point on $\Gamma_r^\star$, as long as the curve $\mathcal{C}_r$ does not pass through any kinematic singularities of the output (3.2). Similarly to the unicycle, we seek a second function so that this new function, together with α_r, yields a relative degree of $\{r_1, \cdots, r_p\} = \{2, \cdots, 2\}$ on $\Gamma_r^\star$. Furthermore, this function should be physically meaningful from the point of view of the path following problem. Let $\mathcal{C}_r^\varepsilon$ denote a tubular neighbourhood of path $\mathcal{C}_r$ and define the map

$$\begin{aligned}\varpi_r : \mathcal{C}_r^\varepsilon &\to \mathbb{R} \\ y_r &\mapsto \arg \min_{\lambda_r \in [0, L_r)} \left\| y_r - \sigma_r(\lambda_r) \right\|.\end{aligned} \tag{3.3}$$

Following (3.3) let

$$\begin{aligned}\pi_1^r : V \subseteq \mathbb{R}^N &\to [0, L_r) \subset \mathbb{R} \\ x_c &\mapsto \varpi_r \circ h_r(x_c)\end{aligned} \tag{3.4}$$

where $V \coloneqq h_r^{-1}(\mathcal{C}_r^\varepsilon)$

We choose a "virtual output" for system (3.1), following [26], as

$$\hat{y}_r = \begin{bmatrix} \pi_1^r(x_c) \\ \alpha_r(x_c) \end{bmatrix} = \begin{bmatrix} \varpi_r \circ h_r(x_c) \\ s_r \circ h_r(x_c) \end{bmatrix}. \tag{3.5}$$

The next result is from [26, Lemma 3.1].

Lemma 3.2.1 ([26])**.** For each $y_r \in \mathcal{C}_r$, the matrix

$$\begin{bmatrix} \mathrm{d}\pi_{1_{y_r}}^r \\ \mathrm{d}s_{r_{y_r}} \end{bmatrix}$$

is orthogonal.

The next result is from [26, Theorem 3.2].

Lemma 3.2.2 ([26])**.** The mechanical system (3.1) with output (3.5) yields a well defined vector relative degree of $\{r_1, \ldots, r_p\} = \{2, \cdots, 2\}$ at each point $x_c^\star$ satisfying $h_r(x_c^\star) \in \mathcal{C}_r$ if and only if

$$\operatorname{Im}\left(\left.\frac{\partial h_r}{\partial x_c}\right|_{x_c=x_c^\star} g_v\left(x_c^\star\right)\right) + \ker\left(\left.\frac{\partial s_r}{\partial y_r}\right|_{y_r=h_r(x_c^\star)}\right) \simeq \mathbb{R}^p \tag{3.6}$$

and

$$\dim\left(\operatorname{Im}\left(\left.\frac{\partial h_r}{\partial x_c}\right|_{x_c=x_c^\star} g_v\left(x_c^\star\right)\right) \cap \ker\left(\left.\frac{\partial s_r}{\partial y_r}\right|_{y_r=h_r(x_c^\star)}\right)\right) = 1 \tag{3.7}$$

hold at $x_c^\star$.

Lemma 3.2.2 allows us to define a local diffeomorphism using the output (3.5) and its derivatives [28, Propostion 5.1.2]. The conditions of Lemma 3.2.2 automatically hold if the system (3.1) has $m = p$ inputs and the curve $\mathcal{C}_r$ does not pass through any kinematic singularities.

Corollary 3.2.3. If system (3.1) is

(i) fully actuated, i.e., $m = N$,

(ii) not kinematically redundant, i.e., $p = N$,

(iii) the Jacobian $\mathrm{d}h_r$ is full rank on $\mathcal{C}_r$,

then (3.1) with output (3.5) is feedback equivalent to a controllable, linear, time-invariant system.

A consequence of Corollary 3.2.3 is that there exists a neighbourhood $U_c \times U_v \in \mathbb{R}^N \times \mathbb{R}^N$ containing $(x_c^\star, x_v^\star)$ such that the mapping $T_r : U_c \times U_v \to T_r(U_c \times U_v) \subset \mathbb{R}^{2N}$, defined by

$$\begin{bmatrix} \eta_1^1 \\ \eta_1^2 \\ \xi_1^1 \\ \xi_1^2 \\ \xi_2^1 \\ \xi_2^2 \\ \vdots \\ \xi_{N-1}^1 \\ \xi_{N-1}^2 \end{bmatrix} = T_r(x) = \begin{bmatrix} \pi_1^r(x_c) \\ L_{f_r}\pi_1^r(x_c) \\ \alpha_r^1(x_c) \\ L_{f_r}\alpha_r^1(x_c) \\ \alpha_r^2(x_c) \\ L_{f_r}\alpha_r^2(x_c) \\ \vdots \\ \alpha_r^{N-1}(x_c) \\ L_{f_r}\alpha_r^{N-1}(x_c) \end{bmatrix} \tag{3.8}$$

is a diffeomorphism onto its image. In (η, ξ)-coordinates the system is modelled as

$$\begin{aligned}
\dot{\eta}_1^1 &= \eta_1^2 \\
\dot{\eta}_1^2 &= L_{f_r}^2 \pi_1^r(x_c) \Big|_{x=T_r^{-1}(\eta,\xi)} + L_g L_{f_r} \pi_1^r(x_c) \Big|_{x=T_r^{-1}(\eta,\xi)} \tau \\
\dot{\xi}_1^1 &= \xi_1^2 \\
\dot{\xi}_1^2 &= L_{f_r}^2 \alpha_r^1(x_c) \Big|_{x=T_r^{-1}(\eta,\xi)} + L_g L_{f_r} \alpha_r^1(x_c) \Big|_{x=T_r^{-1}(\eta,\xi)} \tau \\
&\vdots \\
\dot{\xi}_{N-1}^1 &= \xi_{N-1}^2 \\
\dot{\xi}_{N-1}^2 &= L_{f_r}^2 \alpha_r^{N-1}(x_c) \Big|_{x=T_r^{-1}(\eta,\xi)} + L_g L_{f_r} \alpha_r^{N-1}(x_c) \Big|_{x=T_r^{-1}(\eta,\xi)} \tau.
\end{aligned}$$

Corollary 3.2.3 implies that the feedback transformation

$$\tau = D^{-1}(x) \left(\begin{bmatrix} -L_{f_r}^2 \pi_1^r(x_c) \\ -L_{f_r}^2 \alpha_r^1(x_c) \\ \vdots \\ -L_{f_r}^2 \alpha_r^{N-1}(x_c) \end{bmatrix} + \begin{bmatrix} v_r^{\parallel} \\ v_1^{\pitchfork} \\ \vdots \\ v_{N-1}^{\pitchfork} \end{bmatrix} \right), \tag{3.9}$$

where $D(x)$ is the decoupling matrix and $(v_r^{\parallel}, v_1^{\pitchfork}, \cdots, v_{N-1}^{\pitchfork}) \in \mathbb{R}^N$ are auxiliary control inputs, is well-defined in a neighbourhood of each point on $\Gamma_r^\star$. Thus, in a neighbourhood of each point on $\Gamma_r^\star$, the system dynamics in (η, ξ)-coordinates are modeled by

$$\begin{aligned}
\dot{\eta}_1^1 &= \eta_1^2 \\
\dot{\eta}_1^2 &= v_r^{\parallel} \\
\dot{\xi}_1^1 &= \xi_1^2 \\
\dot{\xi}_1^2 &= v_1^{\pitchfork} \\
&\vdots \\
\dot{\xi}_{N-1}^1 &= \xi_{N-1}^2 \\
\dot{\xi}_{N-1}^2 &= v_{N-1}^{\pitchfork}
\end{aligned} \tag{3.10}$$

Remark 3.2.4. In (η, ξ)-coordinates, the dynamics of η_1^1 and η_1^2 determine the motion of the Euler-Lagrange system along the path while the ξ-subsystem determines the motion of the system off the path. The control objectives in this thesis impose specifications on the

tangential controller $v_r^{\parallel}$ and transversal controllers $(v_1^{\pitchfork}, \cdots, v_{N-1}^{\pitchfork})$. If (3.1) is kinematically redundant and the conditions of Lemma 3.2.2 hold, then the system is feedback equivalent to a partially linearized system. The remaining, possibly nonlinear, dynamics correspond to $(N-p)$ degrees-of-freedom that have no control specifications. If the remaining degrees-of-freedom are left uncontrolled, the system (3.1) may have unstable internal dynamics. To ensure that the internal dynamics are stable, we may impose restrictions on the extra freedoms. One application of this idea to a kinematically redundant 4-DOF robotic manipulator is illustrated in Section 3.3.

The diagram of the transverse feedback linearization of the mechanical system (3.1) is shown in Figure 3.1.

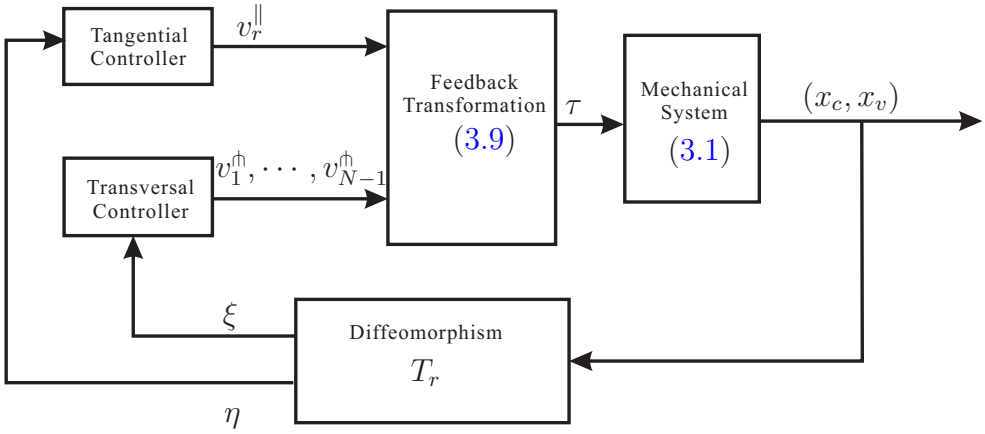


Figure 3.1: Diagram of the transverse feedback linearization of the mechanical system.

3.3 Application to a 4-DOF Manipulator

In this section, we apply the results discussed in Section 3.2 to a fully actuated 4-DOF robotic manipulator. This application is motivated by the availability of a 4-DOF robotic manipulator J10-WAT07 built by Clearpath Robotics at the University of Waterloo, see Figure 3.2.

The simplified model of a revolute 4-DOF manipulator is illustrated in Figure 3.3. Following the notation in Section 3.1, let $x_c = \mathrm{col}(x_{c_1}, x_{c_2}, x_{c_3}, x_{c_4}) \coloneqq \mathrm{col}(q_1, q_2, q_3, q_4)$ and $x_v = \mathrm{col}(x_{v_1}, x_{v_2}, x_{v_3}, x_{v_4}) \coloneqq \mathrm{col}(\dot{q}_1, \dot{q}_2, \dot{q}_3, \dot{q}_4)$. Let $y_r = \mathrm{col}(y_r^1, y_r^2, y_r^3)$ represent the position

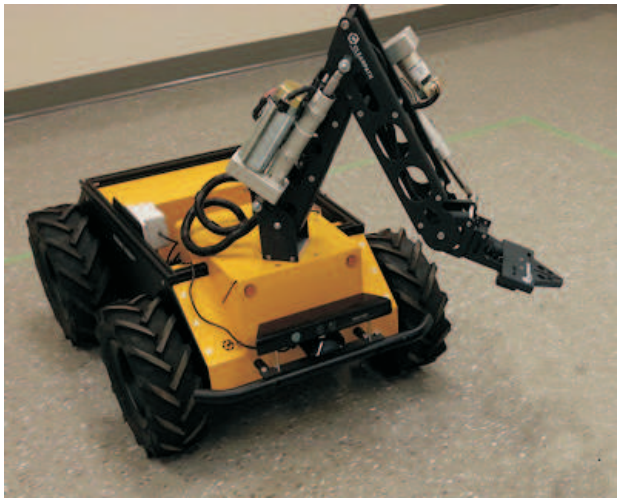


Figure 3.2: Photograph of 4-DOF robotic manipulator J10-WAT07.

of the end-effector

$$y_r = h_r(x_c) \coloneqq \begin{bmatrix} c_1 \left(\ell_2 c_2 + \ell_3 c_{23} + \ell_4 c_{234}\right) \\ s_1 \left(\ell_2 c_2 + \ell_3 c_{23} + \ell_4 c_{234}\right) \\ d_1 + \ell_2 s_2 + \ell_3 s_{23} + \ell_4 s_{234} \end{bmatrix}. \tag{3.11}$$

Here $d_1 > 0$ denotes the link offset of joint 1 and ℓ_2, ℓ_3, $\ell_4 > 0$ denote the link lengths of joints 2, 3, 4 respectively. We use shorthand notation for trigonometric functions, $c_1 \coloneqq \cos(x_{c_1})$, $s_1 \coloneqq \sin(x_{c_1})$, $c_{23} \coloneqq \cos(x_{c_2} + x_{c_3})$ and $c_{234} \coloneqq \cos(x_{c_2} + x_{c_3} + x_{c_4})$. Suppose that we are given a regular Jordan curve $\mathcal{C}_r \subset \mathbb{R}^3$ that satisfies Assumptions 1 and 2 and does not pass through any kinematic singularities of the output (3.11). Let $\alpha_r(x) = s_r \circ h_r(x)$ where h_r is given by (3.11). Let $\Gamma_r^\star$ denote the associated path following manifold.

Note that the 4-DOF manipulator with output (3.11) is kinematically redundant, see Remark 3.2.4. This leaves $N - p = 4 - 3 = 1$ degree-of-freedom with no obvious control specification with respect to the synchronized path following problem. In this thesis we eliminate this redundancy by enforcing a desired pose for the manipulator's wrist position x_{c_4}. To do this, define

$$\pi_2^r(x) \coloneqq x_{c_4}$$

and, following Section 3.2, define the "virtual output" of the 4-DOF manipulator as

$$\hat{y}_r = \begin{bmatrix} \pi_1^r\left(x_c\right) \\ \pi_2^r\left(x_c\right) \\ \alpha_r\left(x_c\right) \end{bmatrix} \tag{3.12}$$

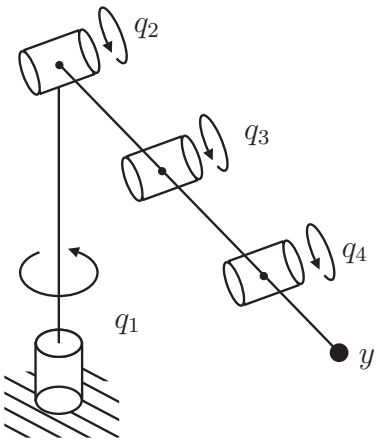


Figure 3.3: 4-DOF robotic manipulator with 4 revolute joints.

Lemma 3.3.1. The system (3.1) with $N = 4$, $p = 3$, $m = 4$ and output (3.5) yields a well defined vector relative degree of $\{2, 2, 2, 2\}$ at each point on $\Gamma_r^\star$.

The proof of Lemma 3.3.1 is omitted because, conceptually, it is virtually the same as the proof of Lemma 2.2.1 and [26, Theorem 3.2].

Lemma 3.3.1 implies that the manipulator is feedback linearizable in a neighbourhood of each point on $\Gamma_r^\star$ using the feedback transformation

$$u = \begin{bmatrix} L_{g_r} L_{f_r} \pi_1^r(x_c) \\ L_{g_r} L_{f_r} \pi_2^r(x_c) \\ L_{g_r} L_{f_r} \alpha_r^1(x_c) \\ L_{g_r} L_{f_r} \alpha_r^2(x_c) \end{bmatrix}^{-1} \left(\begin{bmatrix} -L_{f_r}^2 \pi_1^r(x_c) \\ -L_{f_r}^2 \pi_2^r(x_c) \\ -L_{f_r}^2 \alpha_r^1(x_c) \\ -L_{f_r}^2 \alpha_r^2(x_c) \end{bmatrix} + \begin{bmatrix} v_r^{\parallel} \\ v_r \\ v_1^{\pitchfork} \\ v_2^{\pitchfork} \end{bmatrix} \right), \tag{3.13}$$

where $(v_u^{\parallel}, v_r, v_1^{\pitchfork}, v_2^{\pitchfork}) \in \mathbb{R}^4$ are outer-loop control inputs. Thus, in a neighbourhood of each

point on $\Gamma_r^\star$, the manipulator is feedback equivalent to

$$\begin{aligned}
\dot{\eta}_1^1 &= \eta_1^2 \\
\dot{\eta}_1^2 &= v_r^{\parallel} \\
\dot{\eta}_2^1 &= \eta_2^2 \\
\dot{\eta}_2^2 &= v_r \\
\dot{\xi}_1^1 &= \xi_1^2 \\
\dot{\xi}_1^2 &= v_1^{\pitchfork} \\
\dot{\xi}_2^1 &= \xi_2^2 \\
\dot{\xi}_2^2 &= v_2^{\pitchfork}
\end{aligned} \tag{3.14}$$

Here v_r is the control input related to the remaining degree of freedom, the wrist position x_{c_4}, and we design $v_r = -K_1^r x_{c_4} - K_2^r x_{v_4}$ with $K_1^r, K_2^r > 0$ to stabilize $x_{c_4} = 0$ to zero, physically meaning that the wrist is parallel to the ground, so that the internal dynamics of the manipulator are stable.

3.3.1 Relative Degree and Coordinate Transformations

	β [deg]	h [mm]	ℓ [mm]
Joint 2	65.38	54	335
Joint 3	152.45	54	335
Joint 4	45	54	335

Table 3.1: 4-DOF manipulator model parameters

In this section, we have modeled the 4-DOF robotic manipulator using the joint angles as the configuration variables. However, the experimental platform of the 4-DOF robotic manipulator J10-WAT07 has one rotational and three linear configuration variables, i.e., $x_c = \mathrm{col}(q_1, d_2, d_3, d_4)$. Let $T_0 : \mathbb{R}^4 \to \mathbb{R}^4$ denote the mapping from $\mathrm{col}(q_1, q_2, q_3, q_4)$ to $\mathrm{col}(q_1, d_2, d_3, d_4)$, which is given by [15]

$$\begin{bmatrix} q_1 \\ d_2 \\ d_3 \\ d_4 \end{bmatrix} = T_0(q) = \begin{bmatrix} q_1 \\ \sqrt{\ell_2^2 + h_2^2 - 2\ell_2 h_2 \cos(\beta_2 - q_2)} \\ \sqrt{\ell_2^2 + h_3^2 - 2\ell_2 h_3 \cos(\beta_3 - q_3)} \\ \sqrt{\ell_3^2 + h_4^2 - 2\ell_3 h_4 \cos(\beta_4 - q_4)} \end{bmatrix}. \tag{3.15}$$

Here the angle β_i, distance h_i and link length ℓ_i for $i = 2, 3, 4$ are given in Table 3.1 [15].

We would like to prove that the 4-DOF manipulator with its "virtual output" (3.12) in new coordinates $T_0(q)$ still yields a well-defined relative degree of $\{2, 2, 2, 2\}$ at each point on $\Gamma_r^\star$.

The definition of relative degree is coordinate invariant [28]. Therefore, if the system in $(q, \dot{q})$ yields a well-defined relative degree, and if the map from q to (q_1, d_2, d_3, d_4) is a valid diffeomorphism, then the manipulator will still yield the same relative degree in the new coordinates.

Checking the Jacobian of the map T_0, we have

$$\mathrm{d}T_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \mathrm{d}(d_2)_{q_2} & 0 & 0 \\ 0 & 0 & \mathrm{d}(d_3)_{q_3} & 0 \\ 0 & 0 & 0 & \mathrm{d}(d_4)_{q_4} \end{bmatrix}.$$

with

$$\begin{aligned} \mathrm{d}(d_2)_{q_2} &= -\left(\ell_2^2 + h_2^2 - 2\ell_2 h_2 \cos(\beta_2 - q_2)\right)^{-\frac{1}{2}} \ell_2 h_2 \sin(\beta_2 - q_2) \\ \mathrm{d}(d_3)_{q_3} &= -\left(\ell_2^2 + h_3^2 - 2\ell_2 h_3 \cos(\beta_3 - q_3)\right)^{-\frac{1}{2}} \ell_2 h_3 \sin(\beta_3 - q_3) \\ \mathrm{d}(d_4)_{q_4} &= -\left(\ell_3^2 + h_4^2 - 2\ell_3 h_4 \cos(\beta_4 - q_4)\right)^{-\frac{1}{2}} \ell_3 h_4 \sin(\beta_4 - q_4). \end{aligned}$$

The Jacobian $\mathrm{d}T_0$ is full rank if $\beta_i \neq q_i$, for $i = 2, 3, 4$. Therefore, the map T_0 is a valid diffeomorphism as long as the configurations $\beta_i = q_i$, for $i = 2, 3, 4$, are avoided.

Chapter 4

Implicit representations of parameterized Jordan curves

There are two ways of representing curves: parametric representation and implicit representation. The implicit representation of a curve is defined as a set of points satisfying one or more constraints. The parametrization of a curve is viewed as the curve traced out by a moving point, which turns out to be more useful than its implicit representation in many situations [42]. In the case of simple common curves, such as circles, ellipses and cassini ovals, both the parameterizations and implicit forms of these curves are easy to find. However, for a general parametric curve, for instance the limacon, the implicit form of it is not known.

The transversal feedback linearization approach discussed in this thesis depends on having both a parameterized and implicit representation of the desired paths to complete the proposed coordinate transformations. In this chapter, given a parameterized Jordan curve in $\mathbb{R}^p$, $(p \geq 2)$, we present a procedure to construct an implicit representation of the curve. The procedure involves finding a Serret-Frenet frame for each curve and then expressing the output of the system in terms of its signed distance to the curve along each direction of the frame. Implicit representations of the curve can be constructed by solving the associated equations.

4.1 Planar Curves

Given a planar Jordan curve $\mathcal{C}_u$, let

$$\begin{aligned} \sigma_u :& \mathbb{R} \to \mathbb{R}^2 \\ \lambda_u \mapsto& \begin{bmatrix} \sigma_u^1(\lambda_u) \\ \sigma_u^2(\lambda_u) \end{bmatrix} \end{aligned}$$

be a unit-speed parametrization. Define $T_u(\lambda_u) = \sigma_u'(\lambda_u)$, $N_u(\lambda_u) = R_{\frac{\pi}{2}} T_u(\lambda_u)$ as the unit tangent vector and unit normal vector to curve σ_u, shown in Figure 4.1. Let $y_u \in \mathcal{C}_u^\varepsilon \subset \mathbb{R}^2$

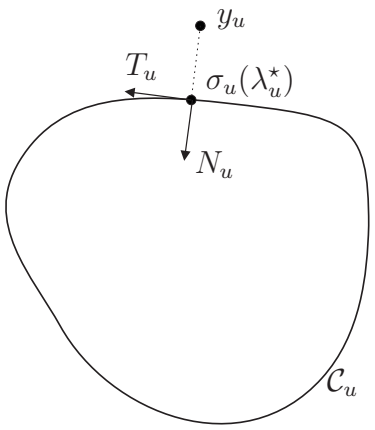


Figure 4.1: Serret-Frenet frame for $\mathcal{C}_u$.

be the position of the unicycle's output and let $\lambda_u^\star = \varpi_u(y_u)$ where ϖ_u is given by (2.3). The value y_u in Figure 4.1 can be expressed as

$$y_u = \sigma_u(\lambda_u^\star) + N_u(\lambda_u^\star)\xi_u^1 \tag{4.1}$$

We get the implicit representation of $\sigma_u(\lambda_u)$ by computing ξ_u^1 from (4.1):

$$s_u(y_u) \coloneqq \langle N_u(\lambda_u^\star), y_u - \sigma_u(\lambda_u^\star) \rangle|_{\lambda_u^\star = \varpi_u(y_u)} . \tag{4.2}$$

Proposition 4.1.1. If $\sigma_u : \mathbb{R} \to \mathbb{R}^2$ is a regular parameterization of a Jordan curve, then (4.2) satisfies Assumption 2.

Proof. We must show that

1. The image of σ_u is $s_u^{-1}(0)$.
2. $\mathrm{d}s_{uy_u}$ is a non-zero vector for each point on the curve.

The equation (4.2) equals to 0 if either

(i) the vectors $N_u(\lambda_u^\star)$ and $y_u - \sigma_u(\lambda_u^\star)$ are orthogonal.

(ii) $y_u - \sigma_u(\lambda_u^\star) = 0$.

By the definition of the normal vector $N_u(\lambda_u^\star)$, condition (i) never occurs. Thus, the function $s_u = 0$ only if $y_u = \sigma_u(\lambda_u^\star)$. Following the calculations in [16], we have

$$\mathrm{d}s_{uy_u} = N_u(\lambda_u^\star)^\top.$$

Since σ_u is a regular curve, $N_u(\lambda_u^\star)$ is non-zero. Thus, $\mathrm{d}s_{uy_u}$ is a non-zero vector for all $y_u \in s_u(y_u)$. □

4.2 Curves in Higher Dimensions

We assigned a curve in $\mathbb{R}^p$ for the Euler-Lagrange system to follow in Section 1.4. To compute the implicit representations of a given parameterized curve in $\mathbb{R}^p$, we would like to determine the Serret-Frenet frame of the curve. To begin with, the definition of a Frenet curve is given.

Definition 4.2.1 ([31])**.** Consider a unit-speed parameterized, simple closed curve $\sigma : \mathbb{R} \to \mathbb{R}^p$ which is p-times continuously differentiable. Then σ is called a **Frenet curve** if at every point the vectors $\sigma', \cdots, \sigma^{p-1}$ are linearly independent. The Serret-Frenet p-frame $v_1, v_2, \cdots, v_p$ is then uniquely determined by the following conditions:

(i) $v_1, v_2, \cdots, v_p$ are orthogonal and positively oriented.

(ii) For every $k \in \{1, \cdots, p-1\}$ one has $\mathrm{span}(v_1, \cdots, v_k) = \mathrm{span}(\sigma', \cdots, \sigma^k)$.

(iii) For every $k \in \{1, \cdots, p-1\}$ one has $\langle \sigma^k,\ v_k \rangle > 0$.

Remark 4.2.2. In the case discussed most often, $p = 3$, the only restrictive condition on a Frenet curve is $\sigma'' \neq 0$. This excludes inflection points. For planar curves ($p = 2$) discussed in above section, there are no restrictive conditions, besides regularity.

Let $\sigma_r(\lambda_r) = \operatorname{col}(\sigma_r^1(\lambda_r), \ldots, \sigma_r^p(\lambda_r))$ be the parameterized representation of the curve for the mechanical system to follow. The following assumption can be used to replace Assumption 2.

Assumption 3. The curve $\sigma_r : \mathbb{R} \to \mathbb{R}^p$ is a Frenet curve.

Under Assumption 3 we can construct the Serret-Frenet frame $v_1, v_2, \cdots, v_p$ of σ_r by means of the *Gram-Schmidt orthogonalization procedure* as follows [31]:

$$
\begin{aligned}
v_1 &:= \sigma_r'(\lambda_r) / \left\|\sigma_r'(\lambda_r)\right\| \\
v_2 &:= \sigma_r''(\lambda_r) / \left\|\sigma_r''(\lambda_r)\right\| \\
v_3 &:= (\sigma_r'''(\lambda_r) - \langle \sigma_r'''(\lambda_r),\ v_1 \rangle\, v_1 - \langle \sigma_r'''(\lambda_r),\ v_2 \rangle\, v_2) / \left\|\cdots\right\| \\
&\vdots \\
v_j &:= (\sigma_r^{(j)}(\lambda_r) - \sum_{i=1}^{j-1} \left\langle \sigma_r^{(j)}(\lambda_r),\ v_i \right\rangle v_i) / \left\|\cdots\right\| \\
&\vdots \\
v_{p-1} &:= (\sigma_r^{(p-1)}(\lambda_r) - \sum_{i=1}^{p-2} \left\langle \sigma_r^{(p-1)}(\lambda_r),\ v_i \right\rangle v_i) / \left\|\cdots\right\|.
\end{aligned}
$$

The last vector v_p is uniquely determined by the condition (i) in Definition 4.2.1. For instance, in the case $p = 3$, the 3-dimensional frame is defined as $v_1 = \sigma_r'(\lambda_r) / \left\|\sigma_r'(\lambda_r)\right\|$, $v_2 = \sigma_r''(\lambda_r) / \left\|\sigma_r''(\lambda_r)\right\|$ and $v_3 = v_1 \times v_2$ and here v_1, v_2 and v_3 represent the unit tangent vector, unit normal vector and unit binormal vector respectively of a 3-dimensional curve.

Remark 4.2.3. In this thesis, we discuss curves that are parameterized by their arc-length, i.e., unit-speed parameterized. For curves that are not unit-speed parameterized, the procedure presented above to find the implicit representations still works.

Let $\mathcal{C}_r$ be such that Assumption 3 holds. Let $y_r \in \mathcal{C}_r^\varepsilon \subset \mathbb{R}^p$ be the position of the mechanical system's output and let $\lambda_r^\star = \varpi_r(y_r)$ where ϖ_r is given by (3.3). The value of y_r can be written as

$$
y_r = \sigma_r(\lambda_r^\star) + v_2(\lambda_r^\star)\xi_1^1 + v_3(\lambda_r^\star)\xi_2^1 + \cdots + v_p(\lambda_r^\star)\xi_{p-1}^1.
$$

Let $y_r^{v_2}, y_r^{v_3}, \cdots, y_r^{v_p}$ denote, respectively, the projection of y_r onto the plane $\{v_1, v_2\}$,

$\{v_1, v_3\}, \cdots, \{v_1, v_p\}$. Then

$$\begin{aligned} y_r^{v_2} &= \sigma_r(\lambda_r^\star) + v_2(\lambda_r^\star)\xi_1^1 \\ y_r^{v_3} &= \sigma_r(\lambda_r^\star) + v_3(\lambda_r^\star)\xi_2^1 \\ &\vdots \\ y_r^{v_p} &= \sigma_r(\lambda_r^\star) + v_p(\lambda_r^\star)\xi_{p-1}^1 \end{aligned} \tag{4.3}$$

By solving for $\xi_1^1, \cdots, \xi_{p-1}^1$ in equations (4.3), the implicit representation of σ_r is

$$s_r(y_r) = \begin{bmatrix} s_r^1(y_r) \\ s_r^2(y_r) \\ \vdots \\ s_r^{p-1}(y_r) \end{bmatrix} \coloneqq \begin{bmatrix} \langle v_2(\lambda_r^\star), y_r^{v_2} - \sigma_r(\lambda_r^\star) \rangle \\ \langle v_3(\lambda_r^\star), y_r^{v_3} - \sigma_r(\lambda_r^\star) \rangle \\ \vdots \\ \langle v_p(\lambda_r^\star), y_r^{v_p} - \sigma_r(\lambda_r^\star) \rangle \end{bmatrix} \Bigg|_{\lambda_r^\star = \varpi_r(y_r)} . \tag{4.4}$$

Proposition 4.2.4. If $\sigma_r : \mathbb{R} \to \mathbb{R}^p, \lambda_r \mapsto \sigma_r(\lambda_r)$ is a regular and smooth simple closed Frenet curve, then (4.4) satisfies Assumption 2.

Proof. Using a similar argument to that used in the proof of Proposition 4.1.1, we will show the following conditions hold :

1. The image of σ_r is $s_r^{-1}(0)$.
2. The Jacobian $\mathrm{d}s_{ry_r}$ is full rank for each point on the curve.

The equation (4.4) equals to 0 if either

(i) the vectors $v_2(\lambda_r^\star)$ and $y_r^{v_2} - \sigma_r(\lambda_r^\star)$, $v_3(\lambda_r^\star)$ and $y_r^{v_3} - \sigma_r(\lambda_r^\star)$, $\cdots$, $v_p(\lambda_r^\star)$ and $y_r^{v_p} - \sigma_r(\lambda_r^\star)$ are all orthogonal.

(ii) $y_r^{v_2} - \sigma_r(\lambda_r^\star) = y_r^{v_3} - \sigma_r(\lambda_r^\star) = \cdots = y_r^{v_p} - \sigma_r(\lambda_r^\star) = 0$.

By the definition of the vectors $v_2, v_3, \cdots, v_p$, condition (i) never occurs. Thus, the function $s_r = 0$ only if $y_r^{v_2} - \sigma_r(\lambda_r^\star) = y_r^{v_3} - \sigma_r(\lambda_r^\star) = \cdots = y_r^{v_p} - \sigma_r(\lambda_r^\star) = 0$, which implies that $y_r = \sigma_r(\lambda_r^\star)$. Therefore, condition 1 holds. The Jacobian of s_r is

$$\mathrm{d}s_{ry_r} = \begin{bmatrix} v_2(\lambda_r^\star)^\top \frac{\partial y_r^{v_2}}{\partial y_r} \\ v_3(\lambda_r^\star)^\top \frac{\partial y_r^{v_3}}{\partial y_r} \\ \vdots \\ v_p(\lambda_r^\star)^\top \frac{\partial y_r^{v_p}}{\partial y_r} \end{bmatrix}$$

For all $y_r \in s_r(y_r)$, we have $y_r = y_r^{v_2} = y_r^{v_3} = \cdots = y_r^{v_p}$. Thus,

$$\frac{\partial y_r^{v_2}}{\partial y_r} = \cdots = \frac{\partial y_r^{v_p}}{\partial y_r} = I_p$$

and

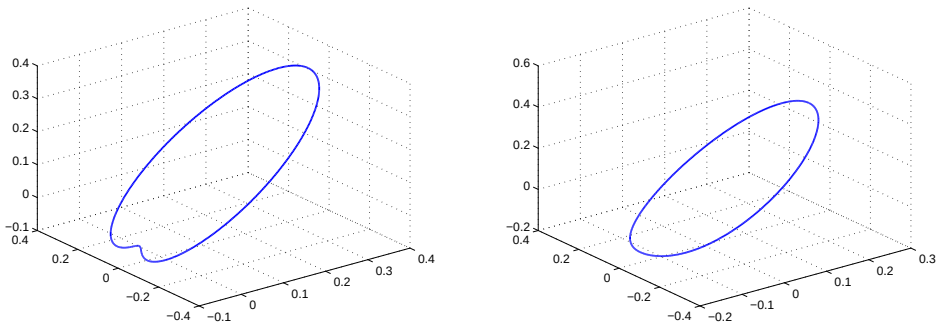
$$\mathrm{d}s_{r\,y_r} = \begin{bmatrix} v_2(\lambda_r^\star) & v_3(\lambda_r^\star) & \cdots & v_p(\lambda_r^\star) \end{bmatrix}^\top.$$

By assumption, since σ_r is a Frenet curve, the vectors $\sigma_r'(\lambda_r), \cdots, \sigma_r^{p-1}(\lambda_r)$ are linearly independent and hence the Serret-Frenet vectors $v_1, \cdots, v_p$ generated by the Gram-Schmidt orthogonalization procedure are also linearly independent. □

Example 4.2.1. Consider a limacon curve in $\mathbb{R}^3$. A parameterized representation is

$$\sigma_r : \mathbb{R} \to \mathbb{R}^3, \qquad \lambda_r \mapsto \begin{bmatrix} (a_0 + a\cos(\lambda_r))\cos(\lambda_r) \\ (b_0 + b\cos(\lambda_r))\sin(\lambda_r) \\ (c_0 + c\cos(\lambda_r))\cos(\lambda_r) \end{bmatrix}.$$

The limacon curves with two different sets of parameters are shown in Figure 4.2. Following



(a) A limacon curve with $a = b = c = 0.15$ and $a_0 = b_0 = c_0 = 0.2$.

(b) A limacon curve with $a = 0.05$, $b = 0.05$, $c = 0.1$, $a_0 = 0.2$, $b_0 = 0.2$ and $c_0 = 0.25$.

Figure 4.2: Limacon curves

the procedure proposed above, we obtain the Serret-Frenet frame $\{v_1, v_2, v_3\}$ of σ_r as

$$\begin{aligned} v_1 &:= \sigma_r'(\lambda_r) / \left\|\sigma_r'(\lambda_r)\right\| \\ v_2 &:= \sigma_r''(\lambda_r) / \left\|\sigma_r''(\lambda_r)\right\| \\ v_3 &:= v_1 \times v_2. \end{aligned}$$

The position $y_r \in \mathbb{R}^3$ can be written in terms of the Serret-Frenet frame $\{v_1, v_2, v_3\}$ as

$$y_r = \sigma_r(\lambda_r^\star) + v_2(\lambda_r^\star)\xi_1^1 + v_3(\lambda_r^\star)\xi_2^1.$$

Let $y_r^{v_2}$ and $y_r^{v_3}$ denote the projections of y_r to the plane $\{v_1, v_2\}$ and $\{v_1, v_3\}$. By the definition of vectors v_1, v_2 and v_3, v_2 and v_3 are the unit normal vectors to plane $\{v_1, v_3\}$ and $\{v_1, v_2\}$ respectively. Thus we have

$$\begin{aligned} y_r^{v_2} &= y_r + k_{v_2} v_3 = \sigma_r(\lambda_r^\star) + v_2(\lambda_r^\star)\xi_1^1 \\ y_r^{v_3} &= y_r + k_{v_3} v_2 = \sigma_r(\lambda_r^\star) + v_3(\lambda_r^\star)\xi_2^1 \end{aligned} \tag{4.5}$$

with

$$\begin{aligned} k_{v_2} &= -(v_3^\top (y_r - \sigma_r(\lambda_r^\star)))/\left\|v_3\right\|^2 \\ k_{v_3} &= -(v_2^\top (y_r - \sigma_r(\lambda_r^\star)))/\left\|v_2\right\|^2. \end{aligned}$$

By solving for ξ_1^1 and ξ_2^1 in equations (4.5), the implicit representations of σ_r is

$$s_r(y_r) = \begin{bmatrix} s_r^1(y_r) \\ s_r^2(y_r) \end{bmatrix} := \left[\begin{array}{c} \langle v_2(\lambda_r^\star), y_r^{v_2} - \sigma_r(\lambda_r^\star) \rangle \\ \langle v_3(\lambda_r^\star), y_r^{v_3} - \sigma_r(\lambda_r^\star) \rangle \end{array} \right] \Bigg|_{\lambda_r^\star = \varpi_r(y_r)}. \tag{4.6}$$

Substituting the expressions of σ_r, $y_r^{v_2}$ and $y_r^{v_3}$ from (4.5) and $\lambda_r^\star$, one can get the implicit representation s_r as functions of y_r.

△

Chapter 5

Control Design

Having feedback linearized the dynamic unicycle (2.1) and, under the hypotheses of Corollary 3.2.3, the Euler-Lagrange system (3.1), in this chapter, we design decentralized control laws that solve the path following portion and centralized controllers that solve the synchronization portion of our problem.

5.1 Path Following Controller Design

Recall that, in (η, ξ)-coordinates, the transversal subsystems are linear and controllable. Thus objectives **A** and **I** of the synchronized path following problem can be solved easily. Specifically, **A** and **I** can be achieved by exponentially stabilizing the transversal subsystems. The simplest choice of transversal controllers are

$$\begin{aligned} v_u^{\pitchfork} &= K_u^1 \xi_1^u + K_u^2 \xi_2^u \\ v_1^{\pitchfork} &= K_1^1 \xi_1^1 + K_1^2 \xi_1^2 \\ &\vdots \\ v_{N-1}^{\pitchfork} &= K_{N-1}^1 \xi_{N-1}^1 + K_{N-1}^2 \xi_{N-1}^2 \end{aligned} \tag{5.1}$$

where $K_u^1, K_u^{21}, K_1^1, K_1^2 \cdots, K_{N-1}^1, K_{N-1}^2 < 0$. The PD controllers (5.1) make the equilibrium point $\xi_1^u = \xi_1^1 = \cdots = \xi_{N-1}^1 = 0$ of the closed-loop transversal subsystems exponentially stable. Physically this implies that if the system is initialized on its path with a velocity

[1] We use the superscript 1 and 2 to distinguish the two gains of each transversal controller.

tangent to the path, then it will remain on the path for all future time. Therefore, path invariance (**I**) is achieved. Furthermore, as long as the closed-loop trajectories are bounded, then $\xi \to 0$ implies that $x \to \Gamma_u^\star$ and $(x_c, x_v) \to \Gamma_r^\star$ which means that the path is attractive (objective **A**).

In the case of the unicycle, the feedback transformation (2.7) is not well defined when $x_4 = 0$. In the singularity set M_0, defined in Section 2.2.1, we apply the feedback transformation (2.9) which yields the unicycle model (2.10). We now design the input u_0 such that, if a singularity is encountered while on the path following manifold, the unicycle does not leave it's path.

Proposition 5.1.1. Consider system (2.10). If the unicycle velocity x_4 is bounded and

$$u_0 = -ax_4 l - b\tilde{\theta} \tag{5.2}$$

with a, $b > 0$, then the equilibrium $(l, \tilde{\theta}) = (0, 0)$ of the $(l, \tilde{\theta})$-subsystem is stable. The control (5.2) renders $\Gamma_u^\star$ controlled invariant. If $x_4 \neq 0$ then $(l, \tilde{\theta}) \to (0, 0)$.

Proof. Consider the candidate Lyapunov-like function $V : \mathbb{R}^2 \to \mathbb{R}$ given by $V = a\frac{l^2}{2} + \frac{\tilde{\theta}^2}{2}$. Taking the Lie derivative of V along the $(l, \tilde{\theta})$-subsystem gives

$$\dot{V} = alx_4 \sin(\tilde{\theta}) - ax_4 l\tilde{\theta} - b\tilde{\theta}^2.$$

In a neighbourhood of $(l, \tilde{\theta} = 0)$, $\sin\tilde{\theta} \approx \tilde{\theta} \Rightarrow \dot{V} \approx -b\tilde{\theta}^2$. Thus the equilibrium point $(l = 0, \tilde{\theta} = 0)$ is stable.

For the second part of the proposition, note that if the unicycle is on the path following manifold, then $l = \tilde{\theta} = 0$. If the unicycle enters the singularity region $M_0 \cap \Gamma_u^\star$, then, since x_4 is bounded by assumption, $u_0 = 0$ and hence the unicycle remains on its desired path.

Finally, suppose that $x_4 \neq 0$. Let

$$E \coloneqq \left\{ \left(l, \tilde{\theta}\right) \in \mathbb{R}^2 : \dot{V} = 0 \right\} = \left\{ \left(l, \tilde{\theta}\right) \in \mathbb{R}^2 : \tilde{\theta} = 0 \right\}.$$

We now characterize the largest invariant set contained in E. For a solution to remain in E we must have that $\tilde{\theta} = 0$ and $\dot{\tilde{\theta}} = 0$. By the choice of control law (5.2), this means that $x_4 l = 0$. Therefore the largest invariant set contained in E is given by

$$M = \left\{ \left(l, \tilde{\theta}\right) \in \mathbb{R}^2 : \tilde{\theta} = x_4 l = 0 \right\}.$$

Since, by assumption $x_4 \neq 0$, the invariance principle yields that $(l, \tilde{\theta}) \to 0$.

□

In M, we apply the feedback transformation (2.7). In M_0, we switch to feedback transformation (2.9) with u_0 defined in (5.2). If this switch occurs while on the path following manifold, the control u_0 maintains controlled invariance of the path following manifold. Moreover, the switching has no effect on the tangential dynamics of the unicycle. In a conclusion, the switching scheme does not spoil the continuity of the control signals on the path following manifold.

Note that Proposition 5.1.1 does not imply asymptotic stability since there are no guarantees that x_4 will remain non-zero. It does show that when the unicycle has a non-zero translational velocity, it will move towards the path following manifold, when the unicycle enters the set $M_0 \backslash \Gamma_u^\star$.

5.2 Comparison of Path Following Controllers

In this section we compare the unicycle controllers (5.1) and (5.2) via simulation. The reason for comparing the controllers is that one could, in principle, simply use (5.2) to achieve **A** and **I** and completely avoid the issue of singularities on $\Gamma_u^\star$. Instead we have decided to use (5.1) under most operating conditions and switch to (5.2) when the state enters the singularity set. The following comparison illustrates the reason for this design decision.

Consider the case when $\mathcal{C}_u$ is a unit circle centred at the origin. Then

$$\sigma_u(\lambda_u) = \begin{bmatrix} \cos(\lambda_u) \\ \sin(\lambda_u) \end{bmatrix}, \qquad s_u(y_u) = {y_u^1}^2 + {y_u^2}^2 - 1.$$

We simulate two copies of the closed-loop dynamic unicycle. One with transversal controller (5.1) and the other with transversal controller (5.2). Both systems have the same tangential controller

$$v_u^{\parallel} = -(\eta_2^u - 0.2)$$

to traverse the path with constant velocity 0.2 unit per second. The comparison simulation results are presented in Figure 5.1. In all the simulations, the unicycle is initialized at $x(0) = (1.2, 0, \frac{\pi}{4}, 0.1)$ and the gains of path following controllers (5.1) and (5.2) are chosen as $a = 1, b = \sqrt{2}$ and $K_u^1 = K_u^2 = -1$ respectively. Figure 5.1(a) shows the closed-loop unicycle output under transversal controller (5.1). Figure 5.1(b) shows the closed-loop unicycle output under controller (5.2). Recall that ξ_1^u and l quantify the path error while ξ_2^u and $\tilde{\theta}$ model, in some sense, the orientation error under coordinate transformation T_u and T^{M_0} respectively. The comparisons of path errors and orientation errors between

the two sets of path following controllers are shown in Figure 5.1(c) and 5.1(d). The path error and orientation error under control law (5.1) converge to zero in less time. Figures 5.1(e) and 5.1(f) show a zoomed in version of the steady-state path and orientation errors. The steady-state path and orientation error are smaller using (5.1) than those obtained from (5.2). The control efforts in these simulations are comparable as shown in Figures 5.2 and 5.3. The comparison of 2% settling time and steady state error of the path error and orientation error under different path following control laws are shown in Table 5.1 and Table 5.2 with respect to five sets of simulations.

Initial condition	2% Settling time		Steady-state error	
	$v_u^{\Uparrow}$	u_0	$v_u^{\Uparrow}$	u_0
$(1.1, 0, \frac{\pi}{2}, 0.1)$	8.79s	19.12s	1.87×10^{-5}	-1.94×10^{-4}
$(0.8, 0, \frac{\pi}{4}, 0.2)$	8.42s	15.19s	1.85×10^{-5}	-2.06×10^{-4}
$(0, 1.2, \pi, 0.2)$	8.80s	18.20s	1.86×10^{-5}	-3.12×10^{-4}
$(-1.2, 0, -\frac{\pi}{2}, 0.1)$	8.85s	17.43s	1.88×10^{-5}	-3.87×10^{-4}
$(0, -0.9, 0, 0.1)$	9.05s	21.49s	1.87×10^{-5}	-3.24×10^{-4}

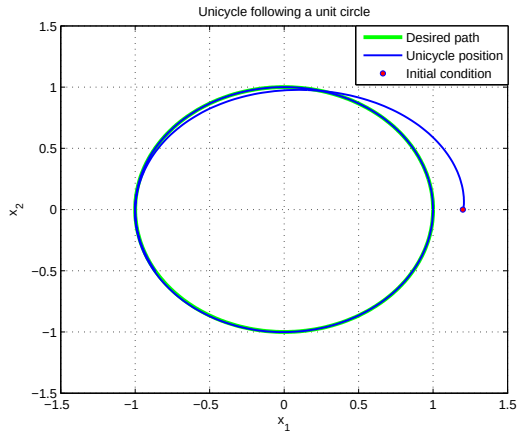
Table 5.1: Simulation results of path error under different path following control laws.

Initial condition	2% Settling time		Steady-state error	
	$v_u^{\Uparrow}$	u_0	$v_u^{\Uparrow}$	u_0
$(1.1, 0, \frac{\pi}{2}, 0.1)$	7.87s	20.66s	-2.66×10^{-6}	8.02×10^{-5}
$(0.8, 0, \frac{\pi}{4}, 0.2)$	9.18s	16.39s	-2.39×10^{-6}	8.54×10^{-5}
$(0, 1.2, \pi, 0.2)$	9.95s	19.18s	-2.78×10^{-6}	1.29×10^{-5}
$(-1.2, 0, -\frac{\pi}{2}, 0.1)$	10.01s	18.36s	-2.89×10^{-6}	1.60×10^{-5}
$(0, -0.9, 0, 0.1)$	10.13s	23.60s	-1.73×10^{-6}	-1.34×10^{-5}

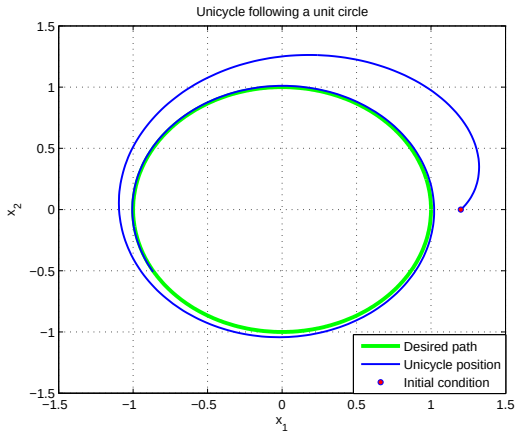
Table 5.2: Simulation results of orientation error under different path following control laws.

5.3 Synchronization Control Design

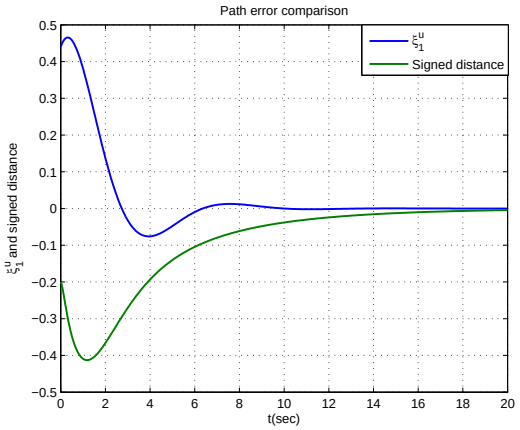
Path following controllers are proposed in Section 5.1 to stabilize the transversal subsystems to make each system approach and get on its path following manifold. We are left to design controllers that achieve **S**. We now present a solution for velocity and position synchronization in this section for the dynamic unicycle (2.1) and an Euler-Lagrange system (3.1) that satisfies the hypotheses of Corollary 3.2.3.



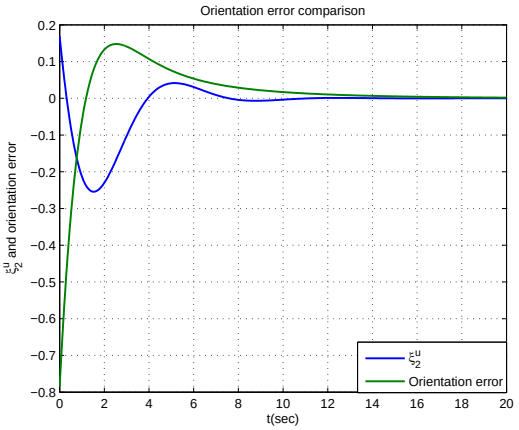
(a) The unicycle following a unit circle.



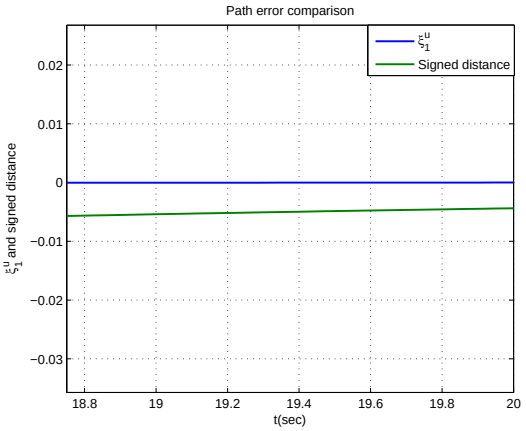
(b) The unicycle following a unit circle.



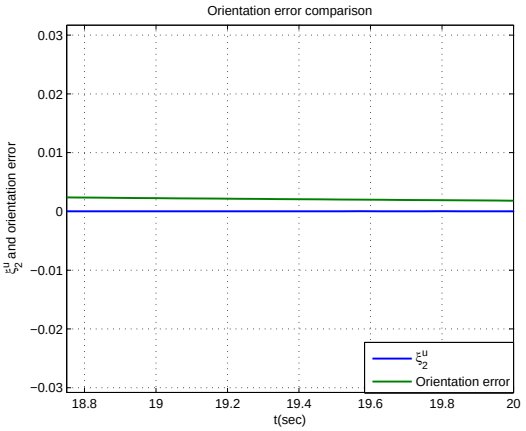
(c) Comparison between path errors.



(d) Comparison between orientation errors.



(e) Zoomed in path errors.



(f) Zoomed in orientation errors.

Figure 5.1: Comparison between path following control laws (5.1) and (5.2).

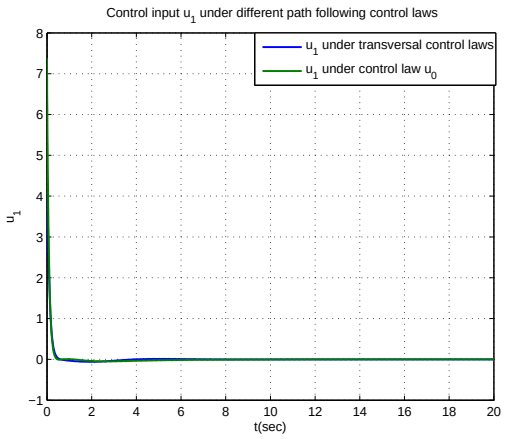


Figure 5.2: Control signal u_1 under different path following control laws.

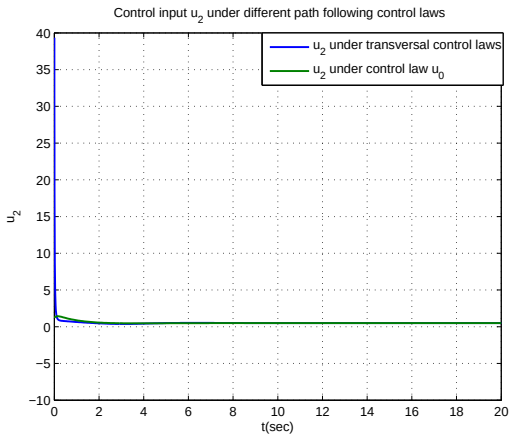


Figure 5.3: Control signal u_2 under different path following control laws.

For position synchronization, we generate a path following synchronization constraint (see Definition 1.4.1) of the form $F(y_u, y_r) = G(\varpi_u(y_u), \varpi_r(y_u)) = G(\eta_1^u, \eta_1^1)$. We characterize a notion of feasibility for such a constraint. To be able to synchronize, the systems must communicate with each other and exchange state information. Since it is not always possible to exchange all state information, we study feasibility under different communication constraints.

5.3.1 Tangential Dynamics

On the path following manifolds $\Gamma_u^\star$ and $\Gamma_r^\star$, the dynamics of unicycle and an Euler-Lagrange system that satisfies the hypotheses of Corollary 3.2.3 are

$$\begin{aligned} \dot{\eta}_1^u &= \eta_2^u \\ \dot{\eta}_2^u &= v_u^{\|} \\ \dot{\eta}_1^1 &= \eta_1^2 \\ \dot{\eta}_1^2 &= v_r^{\|}. \end{aligned} \tag{5.3}$$

Physically, η_1^u and η_1^1 represent the arc-length along the respective paths of the unicycle and the mechanical system from an arbitrarily chosen starting point ($\lambda_u = \lambda_r = 0$) to the location of the system output along the path, while η_2^u and η_2^1 govern the velocities along the paths.

Remark 5.3.1. In the special case we are considering, namely $p = m = N$, the tangential subsystems are completely decoupled from the transversal subsystems. This implies that the synchronization control objectives can be achieved even if the systems are initialized off their path following manifolds.

5.3.2 Velocity Synchronization

In order to achieve velocity synchronization the systems must exchange their velocity information. We select the tangential controllers

$$\begin{aligned} v_u^{\parallel} &= K_{\text{vs}}^1(\eta_2^u - \eta_1^2) \\ v_r^{\parallel} &= K_{\text{vs}}^2(\eta_1^2 - \eta_2^u) \end{aligned} \tag{5.4}$$

with K_{vs}^1, $K_{\text{vs}}^2 < 0$.

Lemma 5.3.2. The solution to the closed-loop tangential subsystem (5.3) with feedback controller (5.4) is such that $\eta_2^u(t) - \eta_1^2(t) \to 0$ as $t \to \infty$. Furthermore, $\eta_2^u(t)$ and $\eta_1^2(t)$ are bounded.

Proof. The closed-loop tangential subsystem is

$$\begin{aligned} \dot{\eta}_1^u &= \eta_2^u \\ \dot{\eta}_2^u &= K_{\text{vs}}^1(\eta_2^u - \eta_1^2) \\ \dot{\eta}_1^1 &= \eta_1^2 \\ \dot{\eta}_1^2 &= K_{\text{vs}}^2(\eta_1^2 - \eta_2^u). \end{aligned}$$

Let the velocity error variable be $e_v := \eta_2^u - \eta_1^2$. The velocity error satisfies

$$\begin{aligned} \dot{e}_v = \dot{\eta}_2^u - \dot{\eta}_1^2 &= K_{\text{vs}}^1(\eta_2^u - \eta_1^2) - K_{\text{vs}}^2(\eta_1^2 - \eta_2^u) \\ &= (K_{\text{vs}}^1 + K_{\text{vs}}^2)e_v. \end{aligned}$$

Since $K_{\text{vs}}^1, K_{\text{vs}}^2 < 0$, the error e_v approaches 0 as $t \to +\infty$, in other words, η_2^u approaches η_1^2 exponentially. In order to show the boundness of η_2^u and η_1^2, let

$$\eta_2 := \begin{bmatrix} \eta_2^u \\ \eta_1^2 \end{bmatrix}$$

and consider the Lyapunov function

$$V(\eta_2) = -\frac{K_{\text{vs}}^2}{2}(\eta_2^u)^2 - \frac{K_{\text{vs}}^1}{2}(\eta_1^2)^2.$$

The time-derivative of V is

$$\frac{\mathrm{d}V}{\mathrm{d}t} = \begin{bmatrix} -K_{\mathrm{vs}}^2 \eta_2^u & -K_{\mathrm{vs}}^1 \eta_1^2 \end{bmatrix} \begin{bmatrix} K_{\mathrm{vs}}^1(\eta_2^u - \eta_1^2) \\ K_{\mathrm{vs}}^2(\eta_1^2 - \eta_2^u) \end{bmatrix} = -K_{\mathrm{vs}}^1 K_{\mathrm{vs}}^2 (\eta_2^u - \eta_1^2)^2 \leq 0.$$

Therefore, η_2^u and η_1^2 are bounded as claimed. □

5.3.3 Position Synchronization

In this section we design control laws that synchronize the positions of unicycle and the Euler-Lagrange system while the systems follow their paths. Recall that η_1^u and η_1^1 denote the location of the system output along the path, see Figure 5.4. This motivates us to

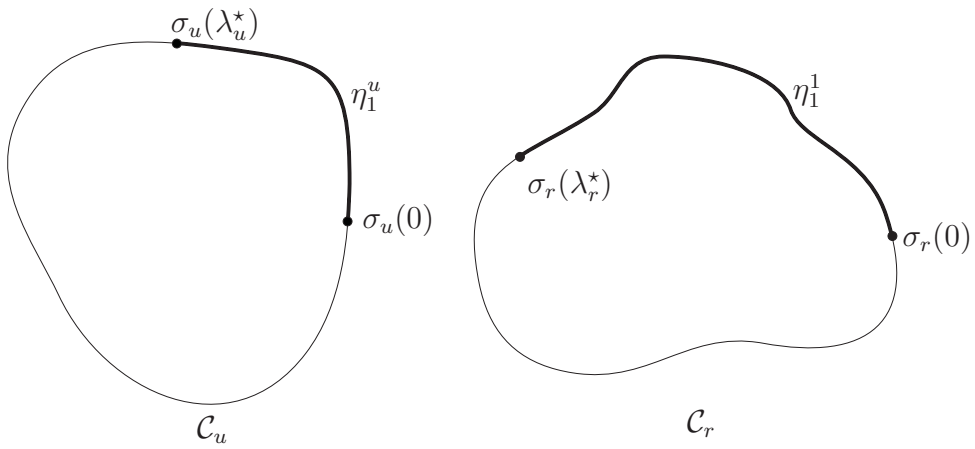


Figure 5.4: Physical illustration of η_1^u and η_1^1. The bold black lines on $\mathcal{C}_u$ and $\mathcal{C}_r$ represent η_1^u and η_1^1 respectively.

generate a path following synchronization constraint (see Definition 1.4.1) of the form

$$F(y_u, y_r) = G(\varpi_u(y_u), \varpi_r(y_u)) = G(\eta_1^u, \eta_1^1) = \eta_1^1 - f(\eta_1^u) \tag{5.5}$$

where $f : [0, L_u) \to [0, L_r)$. With this choice, synchronization is achieved at time t if $\eta_1^1(t) = f(\eta_1^u(t))$. Without loss of generality we assume that $f(0) = 0$. Informally, we refer to the function $f : [0, L_u) \to [0, L_r)$ as a *position constraint*. In Figure 5.5, three different position constraints are illustrated. Each curve from $(0, 0)$ to (L_u, L_r) represents a different position constraint. Next we give conditions on f such that the position constraint is a feasible constraint.

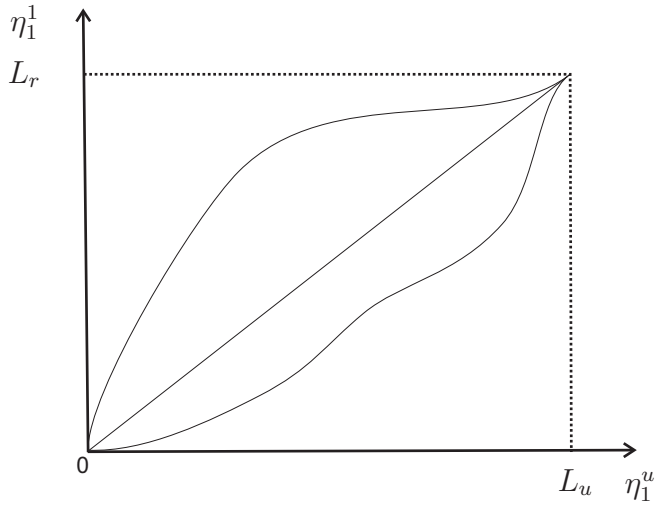


Figure 5.5: Three examples of position constraints.

Definition 5.3.3. A path following synchronization constraint (5.5) is **feasible** if the largest controlled invariant set contained in

$$\mathcal{S} \coloneqq \left\{\eta \in \Gamma_u^\star \times \Gamma_r^\star : \eta_1^1 = f(\eta_1^u)\right\} \tag{5.6}$$

is non-empty.

Definition 5.3.3 means that if the synchronization constraint is feasible, then the unicycle and the Euler-Lagrange system can be initialized such that they maintain synchronization indefinitely.

Proposition 5.3.4. Assume that there are no communication constraints between the systems. If $f : \mathbb{R} \bmod L_u \to \mathbb{R} \bmod L_r$ is twice differentiable then (5.5) is a feasible path following synchronization constraint.

Proof. If the position constraint $\eta_1^1 = f(\eta_1^u)$ is a twice differentiable function, then the first and second derivatives with respect to η_1^u exist. The first derivative of the constraint, in light of the tangential dynamics (5.3), is

$$\eta_1^2 = f'(\eta_1^u)\eta_2^u \tag{5.7}$$

and the second derivative of the constraint function is

$$v_r^{\parallel} = f''(\eta_1^u)\left(\eta_2^u\right)^2 + f'(\eta_1^u)v_u^{\parallel}. \tag{5.8}$$

If we initialize $\eta = (\eta_1^u, \eta_1^1, \eta_2^u, \eta_1^2) = 0$ then $\eta \in \mathcal{S}$ and (5.7) holds. Finally, pick $v_u^{\parallel} = v_r^{\parallel} = 0$ so that (5.8) holds. Thus the largest controlled invariant set in (5.6) is

$$\mathcal{S}^\star = \left\{\eta \in \mathcal{S} : \eta_1^2 = f'(\eta_1^u)\eta_2^u\right\} \tag{5.9}$$

which is non-empty since it contains $\eta = 0$. □

Remark 5.3.5. In the full communication case, the selection of $v_u^{\parallel}$ and $v_r^{\parallel}$ that make (5.8) hold on $\mathcal{S}^\star$ are not unique.

The characterization of feasible constraints in Proposition 5.3.4 shows that if the unicycle and the Euler-Lagrange system are initially synchronized, and remain stationary, they remain synchronized. Clearly, such a characterization is not useful in cases where the systems should traverse their respective paths. In these cases, the feasibility of the constraint is highly dependent on the information being exchanged between the unicycle and the Euler-Lagrange system.

Proposition 5.3.6. Let $f : \mathbb{R} \bmod L_u \to \mathbb{R} \bmod L_r$ be a twice differentiable function. Suppose that one of the following hypotheses hold.

(i) There is no communication and $f'(\eta_1^u) \equiv 0$.

(ii) The Euler-Lagrange system has $v_u^{\parallel}$ available for feedback and $f'(\eta_1^u)$ is a non-zero constant.

(iii) The Euler-Lagrange system has $(v_u^{\parallel}, \eta_1^u, \eta_2^u)$ available for feedback.

Then the largest controlled invariant subset of (5.6) is non-empty, given by

$$\mathcal{S}^\star = \left\{\eta \in \mathcal{S} : \eta_1^2 = f'(\eta_1^u)\eta_2^u\right\} \tag{5.10}$$

with $\eta_u^2 \not\equiv 0$ on $\mathcal{S}^\star$.

Proof. In the above three hypotheses, we can always initialize $\eta = (\eta_1^u, \eta_1^1, \eta_2^u, \eta_1^2)$ with non-zero η_2^u such that $\eta \in \mathcal{S}$ and equation (5.7) holds. In order to prove that the set $\mathcal{S}^\star \setminus \{\eta \in \mathcal{S}^\star : \eta_2^u \equiv 0\}$ is non-empty, it is sufficient to show that there exist selections of $v_u^{\parallel}$ and $v_r^{\parallel}$ such that equation (5.8) holds in those cases. In hypothesis (i), since $f'(\eta_1^u) \equiv 0$, the equation (5.8) always holds with any $v_u^{\parallel}$ if we pick $v_r^{\parallel} \equiv 0$. In hypothesis (ii), the equation (5.8) is reduced to $v_r^{\parallel} = f'(\eta_1^u)v_u^{\parallel}$ and it is possible to pick $v_r^{\parallel} = f'(\eta_1^u)v_u^{\parallel}$ with

arbitrary selection of $v_u^{\parallel}$ such that (5.8) holds if the Euler-Lagrange system has $v_u^{\parallel}$ available for feedback and $f'(\eta_1^u)$ is a non-zero constant. Given any twice differentiable constraint function, we can select $v_u^{\parallel}$ freely and then let $v_r^{\parallel} = f''(\eta_1^u)(\eta_2^u)^2 + f'(\eta_1^u)v_u^{\parallel}$, which is always possible if the Euler-Lagrange system has $(v_u^{\parallel}, \eta_1^u, \eta_2^u)$ available for feedback, such that (5.8) holds. In conclusion, the set $\mathcal{S}^\star \setminus \{\eta \in \mathcal{S}^\star : \eta_2^u \equiv 0\}$ is non-empty under any of the above hypotheses. □

The hypotheses in Proposition 5.3.6 characterize the feasibility of constraints functions associated with communication constraints in terms of non-stationary synchronization. Specifically, this characterization illustrates that the systems can remain synchronized in non-stationary case if they are initialized on the non-stationary synchronization set, i.e., $\mathcal{S}^\star \setminus \{\eta \in \mathcal{S}^\star : \eta_2^u \equiv 0\}$. Next, our objective is to design synchronization control laws stabilizing the motions of systems to the set $\mathcal{S}^\star$ if they are not initially synchronized. To begin with, synchronization error coordinates are introduced.

Remark 5.3.7. In this thesis we write the path following synchronization constraint in the form (5.5) with $f : \mathbb{R} \bmod L_u \to \mathbb{R} \bmod L_r$. If we instead consider position constraints of the form $f : \mathbb{R} \bmod L_r \to \mathbb{R} \bmod L_u$ then the discussion above holds, *mutatis mutandis*, with the role of the unicycle and the Euler-Lagrange system swapping.

Position Synchronization Error

Given a feasible path following synchronization constraint of the form (5.5) with $f : \mathbb{R} \bmod L_u \to \mathbb{R} \bmod L_r$, define the synchronization error as

$$e_1 \coloneqq \eta_1^1 - f(\eta_1^u)$$

and let

$$e_2 \coloneqq \eta_1^2 - f'(\eta_1^u)\eta_2^u.$$

The (e_1, e_2)-dynamics evolve according to

$$\begin{aligned} \dot{e}_1 &= e_2 \\ \dot{e}_2 &= v_r^{\parallel} - f''(\eta_1^u)\,(\eta_2^u)^2 - f'(\eta_1^u)v_u^{\parallel}. \end{aligned} \tag{5.11}$$

Define the fictitious input $u_e \coloneqq v_r^{\parallel} - f''(\eta_1^u)\,(\eta_2^u)^2 - f'(\eta_1^u)v_u^{\parallel}$. To meet our control objective, we seek a control law such that the synchronization error $e_1 \to 0$ as $t \to \infty$ and so that if

$(e_1(0), e_2(0)) = 0$, then, for all $t \geq 0$, $(e_1(t), e_2(t)) = 0$. To this end we select the simplest controller that exponentially stabilizes the origin of (5.11)

$$u_e = k_{e_1} e_1 + k_{e_2} e_2, \tag{5.12}$$

with k_{e_1}, $k_{e_2} < 0$. The division of u_e into $v_u^{\|}$ and $v_r^{\|}$ is not unique and the available choices depend on the communication constraints. For instance, if f is a non-zero linear function, the synchronization control law (5.12) can be implemented by only exchanging position information, i.e., η_1^u and η_1^1. For instance, let the division of u_e into $v_u^{\|}$ and $v_r^{\|}$ be

$$\begin{aligned} v_u^{\|} &= \frac{-\frac{k_{e_1}}{2}(\eta_1^1 - f(\eta_1^u)) + k_{e_2}(f'(\eta_1^u) - c)}{f'(\eta_1^u)} \\ v_r^{\|} &= \frac{k_{e_1}}{2}(\eta_1^1 - f(\eta_1^u)) + k_{e_2}(\eta_1^2 - c) \end{aligned}$$

here c is a mutually known, non-zero, constant, then we have $e_1, e_2 \to 0$ as $t \to \infty$. Moreover, the velocities of systems $\eta_2^u \to \frac{c}{f'(\eta_u)}$ and $\eta_1^2 \to c$ as $t \to \infty$, indicating the motions of systems satisfy the synchronization constraint while moving along their paths.

There is a small technical issue associated with the control law (5.12). The state space of the (e_1, e_2)-system is not $\mathbb{R}^2$. This can be seen by the fact that if we pick $\eta_1^u = 0, \eta_1^1 = L_r$ and $\eta_1^2 = f(\eta_u^2)$, satisfying the desired synchronization constraint, then the error state $(L_r, 0)$ is physically the same state with $(e_1, e_2) = (0, 0)$. Thus e_1 is periodic with period L_r and the state space of system (5.11) is a cylinder $\mathbb{R}\text{mod}L_r \times \mathbb{R}$ rather than a plane $\mathbb{R}^2$. In this case, all the physically identical states $(nL_R, 0)$, $n \in \mathbb{Z}$, should be locally stable. However, the feedback control law (5.12) only stabilizes the origin of system (5.11). For example, if we initialize system (5.11) at $(L_r, 0)$, then the systems are synchronized and no further control action is needed. However, the controller (5.12) drives the state (e_1, e_2) from $(L_r, 0)$ to $(0, 0)$ causing unnecessary motions [11]. The cause of this problem is the periodicity of e_1 and to solve this, we use a control law from [11] that stabilizes all the states that are physically identical to $(e_1, e_2) = (0, 0)$.

The feedback control law from [11] that achieves this is

$$u_e = -\text{sign}\,(e_2)\,|e_2|^{\alpha} - \text{sign}\left(\sin\left(\frac{\varphi_\alpha}{L_r} 2\pi\right)\right)\left|\sin\left(\frac{\varphi_\alpha}{L_r} 2\pi\right)\right|^{\frac{\alpha}{2-\alpha}} \tag{5.13}$$

with $\alpha \in (0, 1)$ and $\varphi_\alpha \coloneqq e_1 + \frac{1}{2-\alpha}\text{sign}\,(e_2)\,|e_2|^{2-\alpha}$.

The phase portrait of (5.11) with feedback controller (5.13) is shown in Figure 5.6 with $\alpha = \frac{1}{3}$ and $L_r = 4$. In Figure 5.6, the three solid dots represent three identical states

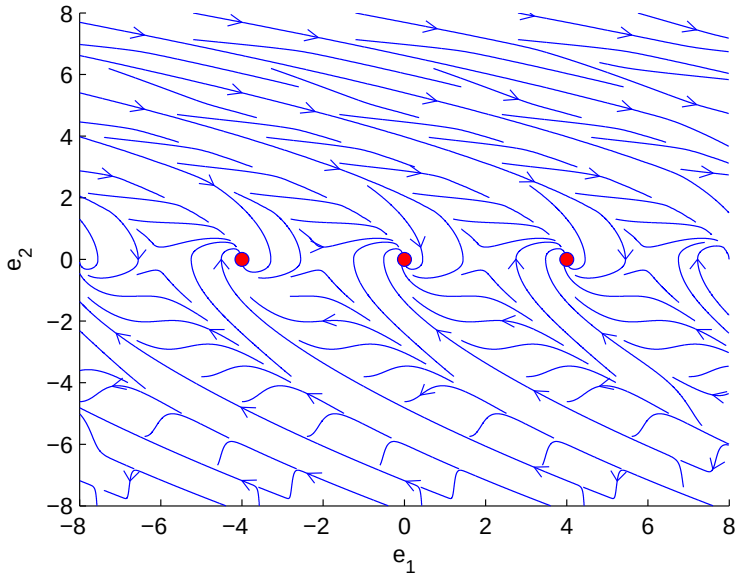


Figure 5.6: Rotational double integrator (5.11) with controller (5.13)

$(\pm 4, 0)$ and $(0, 0)$ which are all the local stable equilibrium. The phase portrait shows that the closed-loop system (5.11) with control law (5.13) has locally stable equilibrium at $(nL_r, 0)$, $n \in \mathbb{Z}$ [11].

When there are no communication constraints we can accomplish position synchronization as well as guarantee boundness of the velocities of the unicycle and the Euler-Lagrange system, i.e., η_1^u and η_1^2. Under the control law (5.13), $e_2 \to 0$ as $t \to \infty$ so that $\eta_1^2 \to f'(\eta_1^u)\eta_2^u$ as $t \to \infty$. Since $f'(\eta_1^u)$ is bounded, η_1^2 bounded implies η_2^u is bounded. Thus, one can freely choose $v_u^{\parallel}$ to guarantee the boundness of η_2^u, and fix $v_r = u_e + f''(\eta_1^u) + f'(\eta_1^u)v_u^{\parallel}$ to achieve the position synchronization and the boundness of η_1^2. The diagram illustrating the relations between u_e and tangential dynamics (5.3) is shown in Figure 5.7.

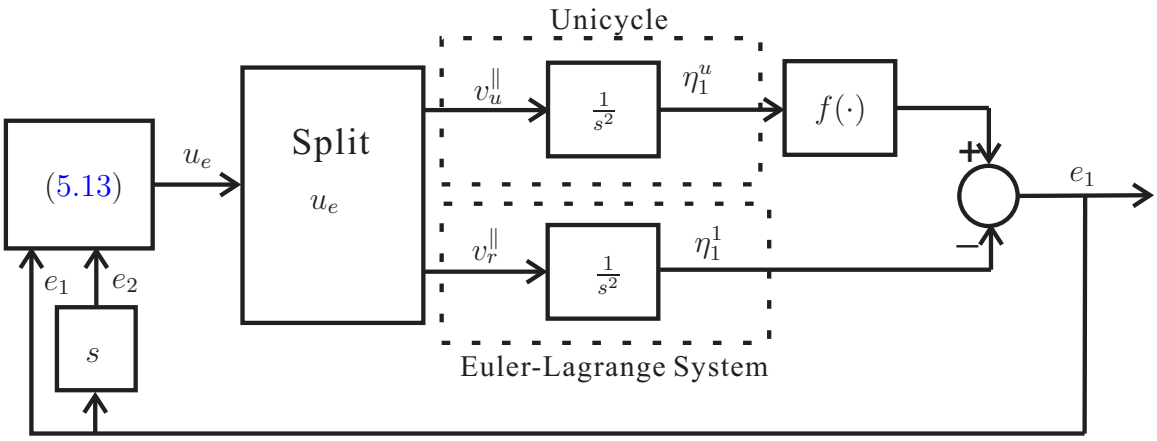


Figure 5.7: The relationship between u_e and tangential dynamics (5.3)

Chapter 6

Application to a Differential Drive Vehicle and a 4-DOF Manipulator

We apply the controllers proposed in the previous chapters to a differential drive vehicle and a fully actuated four degree-of-freedom robotic manipulator with four revolute joints. Simulation results are shown to illustrate the effectiveness of our control algorithms.

6.1 Implementation Issues

Before presenting simulation results, we first address some issues that could make the implementation of the proposed controllers difficult.

6.1.1 Computation of the Tangential States

The computations of η_1^u and η_1^1 are presented in Chapter 2 and 3 as

$$\eta_1^u = \varpi_u \circ h_u(x)$$
$$\eta_1^1 = \varpi_r \circ h_r(x_c).$$

In general, the maps ϖ_u, ϖ_r do not have closed-form expressions. The above values are calculated by doing a line search minimization algorithm. Hence, the states η_1^u and η_1^1 are available for feedback. However, since ϖ_u, ϖ_r do not have closed-form expressions, it is not immediately clear how to compute the other tangential states. Furthermore, in most

cases there is no closed-form expression for the unit-speed parameterization of the regular curves σ_u, σ_u. We address both of these issues below.

Consider two regular parameterizations σ_u and σ_r, not necessarily parameterized by their arc-length. Introduce functions that return the arc-length of the curves $\mathcal{C}_u$ and $\mathcal{C}_r$, respectively, at λ_u and λ_r

$$\begin{aligned} p_u(\lambda_u) &\coloneqq \int_0^{\lambda_u} \left\| \frac{\mathrm{d}\sigma_u}{\mathrm{d}\lambda_u} \right\| \mathrm{d}u \\ p_r(\lambda_r) &\coloneqq \int_0^{\lambda_r} \left\| \frac{\mathrm{d}\sigma_r}{d\lambda_r} \right\| \mathrm{d}u. \end{aligned} \tag{6.1}$$

The tangential states η_1^u and η_1^1 can be written as

$$\begin{aligned} \eta_1^u = p_u \circ \varpi_u \circ h_u(x) &= \int_0^{\varpi_u(h_u(x))} \left\| \frac{\mathrm{d}\sigma_u}{\mathrm{d}\lambda_u} \right\| \mathrm{d}u \\ \eta_1^1 = p_r \circ \hat{\varpi}_r \circ h_r(x_c) &= \int_0^{\varpi_r(h_r(x))} \left\| \frac{\mathrm{d}\sigma_r}{\mathrm{d}\lambda_r} \right\| \mathrm{d}u \end{aligned}$$

To simplify notation, let $\lambda_u^\star = \varpi_u(h_u(x))$ and $\lambda_r^\star = \varpi_r(h_r(x))$. To calculate η_2^u and η_1^2 note that

$$\begin{aligned} \eta_2^u &= \frac{\partial \left(p_u \circ \varpi_u \circ h_u \right)}{\partial x} \frac{\mathrm{d}x}{\mathrm{d}t} \\ &= \left(\frac{\partial p_u}{\partial \lambda_u} \right) \Big|_{\lambda_u = \lambda_u^\star} \left(\frac{\partial \varpi_u}{\partial y_u} \right) \Big|_{y_u = h_u(x)} \begin{bmatrix} x_4 \cos(x_3) \\ x_4 \sin(x_3) \end{bmatrix} \\ &= \| \sigma_u'(\lambda_u^\star) \| \left(\frac{\partial \varpi_u}{\partial y_u} \right) \Big|_{y_u = h_u(x)} \begin{bmatrix} x_4 \cos(x_3) \\ x_4 \sin(x_3) \end{bmatrix} \end{aligned}$$

Following the arguments in [16], we have

$$\frac{\partial \varpi_u}{\partial y_u} \Big|_{y_u = h_u(x)} = \frac{\sigma_u'(\lambda_u^\star)^\top}{\| \sigma_u'(\lambda_u^\star) \|^2}$$

and therefore

$$\eta_2^u = \frac{\sigma_u'(\lambda_u^\star)^\top}{\| \sigma_u'(\lambda_u^\star) \|} \begin{bmatrix} x_4 \cos(x_3) \\ x_4 \sin(x_3) \end{bmatrix}. \tag{6.2}$$

The expression (6.2) can be effectively calculated since $\lambda_u^\star$ is available from the calculations needed to find η_u^1. To simplify the notation, let

$$\Delta \coloneqq \frac{\sigma_u'(\lambda_u^\star)^\top}{\| \sigma_u'(\lambda_u^\star) \|}, \qquad \Omega \coloneqq \begin{bmatrix} x_4 \cos(x_3) \\ x_4 \sin(x_3) \end{bmatrix}$$

To compute $L^2_{f_u}\pi_u$ we differentiate η^u_2,

$$L^2_{f_u}\pi_u = \dot{\Delta}\Omega + \Delta\dot{\Omega}.$$

The term $\dot{\Delta} = \mathrm{d}\Delta_{\lambda_u}\dot{\lambda}_u$ can be found by noting that

$$\mathrm{d}\Delta_{\lambda_u} = \frac{(\sigma_u{}'')^\top \|\sigma_u{}'\|^2 - (\sigma_u{}')^\top \sum_{i=1}^2 \sigma_u^{i\,\prime}\sigma_u^i}{\|\sigma_u{}'\|^3}$$

and using the chain rule for η^u_1,

$$\dot{\lambda}^\star_u = \frac{1}{\|\sigma'_u\|^2}\eta^u_2.$$

The term $\dot{\Omega}$ is easy to get using the system dynamics. Putting these calculations together we obtain

$$L^2_{f_u}\pi_u(x) = \frac{(\sigma_u{}'')^\top \|\sigma_u{}'\|^2 - (\sigma_u{}')^\top \sum_{i=1}^2 \sigma_u^{i\,\prime}\sigma_u^i}{\|\sigma_u{}'\|^5}\eta^u_2 \begin{bmatrix} x_4\cos(x_3) \\ x_4\sin(x_3) \end{bmatrix}. \tag{6.3}$$

Since all the terms in (6.3) are available, $L^2_{f_u}\pi_u$ can be effectively computed. Similarly, for the mechanical system, we get

$$\eta^2_1 = \frac{\partial\left(p_r \circ \varpi_r \circ h_r\right)}{\partial x_c}\frac{\mathrm{d}x_c}{\mathrm{d}t} = \frac{\sigma'_r(\lambda^\star_r)^\top}{\|\sigma'_r(\lambda^\star_r)\|}\mathrm{d}h_{rx_c}x_v$$

and

$$\begin{aligned} L^2_{f_r}\pi^1_r =& \frac{(\sigma_r{}'')^\top \|\sigma_r{}'\|^2 - (\sigma_r{}')^\top \sum_{i=1}^3 \sigma_r^{i\,\prime}\sigma_r^i}{\|\sigma_r{}'\|^5}\eta^2_1\mathrm{d}h_{rx_c}x_v \\ &+ \frac{\sigma'_r(\lambda^\star_r)^\top}{\|\sigma'_r(\lambda^\star_r)\|}\left(\mathrm{d}h_{rx_c}\left(-M^{-1}(x_c)B(x_c)\left(C(x_c,x_v)+G(x_c)\right)\right) + \mathrm{d}(\mathrm{d}h_{rx_c}x_v)_{x_c}x_v\right). \end{aligned}$$

6.1.2 Computation of the Transversal States

In Chapter 4 we showed how to construct the zero level set representations of $\mathcal{C}_u$ and $\mathcal{C}_r$. To compute the rest of the transversal states we need to differentiate these expressions. We now discuss how this can be achieved.

Suppose that we have the implicit constructions (4.2) and (4.4). Then $\xi_1^u = s_u \circ h_u(x)$, $\xi_1^1 = s_r^1 \circ h_r(x)$ and $\xi_2^1 = s_r^2 \circ h_r(x)$. To calculate derivatives of these expressions we have (4.4)

$$\begin{aligned}
\xi_2^u &= \dot{\xi}_1^u \\
&= \langle N_u(\lambda_u^\star),\ \dot{y}_u \rangle \\
&= N_u(\lambda_u^\star)^\top \begin{bmatrix} x_4 \cos(x_3) \\ x_4 \sin(x_3) \end{bmatrix} \\
\xi_1^2 &= \dot{\xi}_1^1 \\
&= \langle v_2(\lambda_r^\star),\ \dot{y}_r^{v_2} \rangle \\
&= v_2(\lambda_r^\star)^\top \mathrm{d}(y_r^{v_2})_{y_r} \mathrm{d}h_{rx_c} x_v \\
\xi_2^2 &= \dot{\xi}_2^1 \\
&= \langle v_3(\lambda_r^\star),\ \dot{y}_r^{v_3} \rangle \\
&= v_3(\lambda_r^\star)^\top \mathrm{d}(y_r^{v_3})_{y_r} \mathrm{d}h_{rx_c} x_v.
\end{aligned} \tag{6.4}$$

Differentiating equations (6.4) and using expressions of $\dot{\lambda}_u^\star$ and $\dot{\lambda}_r^\star$ from Section 6.1.1

$$\begin{aligned}
L_{f_u}^2 \alpha_u =& \frac{\partial N_u(\lambda_u^\star)^\top}{\partial \lambda_u^\star} \frac{1}{\|\sigma_u'(\lambda_u^\star)\|^2} \eta_u^2 \begin{bmatrix} x_4 \cos(x_3) \\ x_4 \sin(x_3) \end{bmatrix} \\
L_{f_r}^2 \alpha_r^1 =& \frac{\partial v_2(\lambda_r^\star)^\top}{\partial \lambda_r^\star} \frac{1}{\|\sigma_r'(\lambda_r^\star)\|^2} \eta_1^2 \mathrm{d}(y_r^{v_2})_{y_r} \mathrm{d}h_{rx_c} x_v \\
&+ v_2(\lambda_r^\star)^\top \mathrm{d}(y_r^{v_2})_{y_r} \left(\mathrm{d}h_{rx_c} \left(-M^{-1}(x_c) B(x_c) \left(C(x_c, x_v) + G(x_c) \right) \right) + \mathrm{d}(\mathrm{d}h_{rx_c} x_v)_{x_c} x_v \right) \\
L_{f_r}^2 \alpha_r^2 =& \frac{\partial v_3(\lambda_r^\star)^\top}{\partial \lambda_r^\star} \frac{1}{\|\sigma_r'(\lambda_r^\star)\|^2} \eta_1^2 \mathrm{d}(y_r^{v_3})_{y_r} \mathrm{d}h_{rx_c} x_v \\
&+ v_3(\lambda_r^\star)^\top \mathrm{d}(y_r^{v_3})_{y_r} \left(\mathrm{d}h_{rx_c} \left(-M^{-1}(x_c) B(x_c) \left(C(x_c, x_v) + G(x_c) \right) \right) + \mathrm{d}(\mathrm{d}h_{rx_c} x_v)_{x_c} x_v \right)
\end{aligned}$$

The partial derivatives in the above expressions are easy to implement using symbolic algebra software.

6.1.3 Experimental Set up and Inter-robot Communication

We now discuss how these control laws can be implemented on the laboratory equipment at the University of Waterloo, see Figure 2.1 and 3.2. The differential drive vehicle (Chameleon R200) and the 4-DOF robotic manipulator (J10-WAT07) and both built by Clearpath Robotics and we implement the synchronized path following controllers via the Robot Operating System (ROS). Precisely, the control algorithms developed for the

Chameleon R200 are interpreted into C++ running on a PC as a ROS node to control its left and right wheel velocities and read encoder outputs for feedback.

The software control for the manipulator is programmed completely by Clearpath Robotics via LabView ®. Each joint is driven by an actuator controlled by a PWM input generated by an FPGA [15]. The interfacing between LabView, the FPGA and the actuators is designed by Clearpath Robotics. The controller runs on LabView Real-Time, communicating with the actuators of the manipulator through the FPGA.

In order to let the manipulator communicate with the vehicle, we would like to control the manipulator via ROS instead of LabView. This is because the differential drive robot is controlled by ROS, running in Linux, and inter-robot communication is relatively easy to implement in ROS. However, LabView runs under the Microsoft Windows ®operating system and ROS cannot run on Windows.

If the LabView code is downloaded to the National Instruments single-board RIO card ahead of time, the manipulator can be controlled via ROS independently of LabView by running our control algorithms as a ROS node. The ROS interface uses the `rosbridge` node in conjunction with a toolkit developed by Clearpath Robotics to provide interoperability between ROS and LabVIEW. Once a connection has been opened, the sbRIO can provide feedback to ROS and receive commands. The architecture of control of the manipulator is shown in Figure 6.1. Here the `roscore` is a collection of nodes and programs that are pre-requisites of a ROS-based system and must be running in order for ROS nodes to communicate.

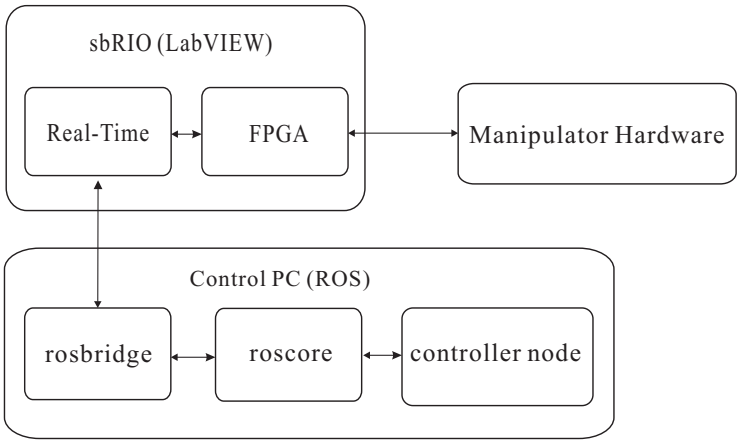


Figure 6.1: The architecture of the control system of the manipulator J10-WAT07.

After the ROS control for each robot is achieved, we now are able to build the commu-

nication among the robots by following the tutorials on the ROS wiki website such that each of them has the tangential information of the other robot available for feedback.

6.2 Simulation Results

We present simulations of our controllers using realistic model parameters. The link offset and link lengths of the robotic manipulator are $d_1 = 70mm$, $\ell_2 = 335mm$, $\ell_3 = 335mm$ and $\ell_4 = 335mm$. In these simulations, we assume there are no communication constraints.

6.2.1 Velocity Synchronized Path Following

The simulation results of the velocity synchronized path following controllers (5.4) and (5.1) are presented in Figure 6.2. In this velocity synchronization simulation, the desired paths are chosen as elliptic curves which are given as

$$\sigma_u : \mathbb{R} \to \mathbb{R}^2, \qquad \lambda_u \mapsto \begin{bmatrix} 0.6\cos(\lambda_u) \\ 0.3\sin(\lambda_u) \end{bmatrix}$$
$$\sigma_r : \mathbb{R} \to \mathbb{R}^3, \qquad \lambda_r \mapsto \begin{bmatrix} 0.6\cos(\lambda_r) \\ 0.3\sin(\lambda_r) \\ 0.3\cos(\lambda_r) + 0.07 \end{bmatrix}.$$

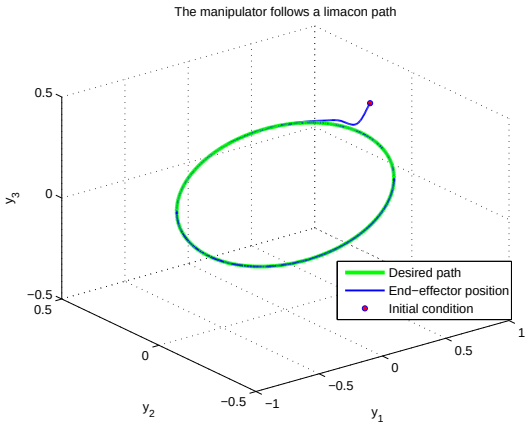
The unicycle and the manipulator are following their curves in Figure 6.2(a) and 6.2(b). Figure 6.2(c) shows that the velocities η_2^u and η_1^2 are approaching the same and the velocity error e_v

$$e_v = \eta_2^u - \eta_1^2.$$

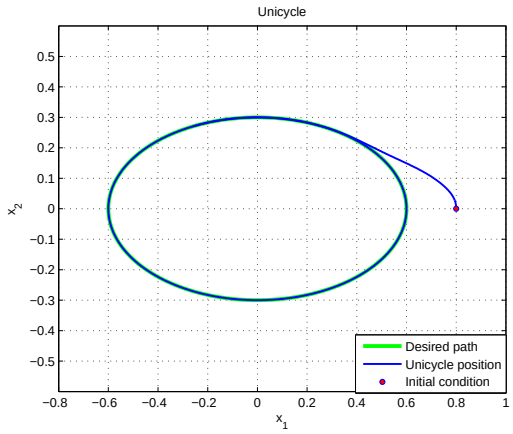
converging to 0 in Figure 6.2(d). Note that since the curves σ_u and σ_r are not unit-speed parameterized, the velocities approach to same but varying.

6.2.2 Position Synchronized Path Following

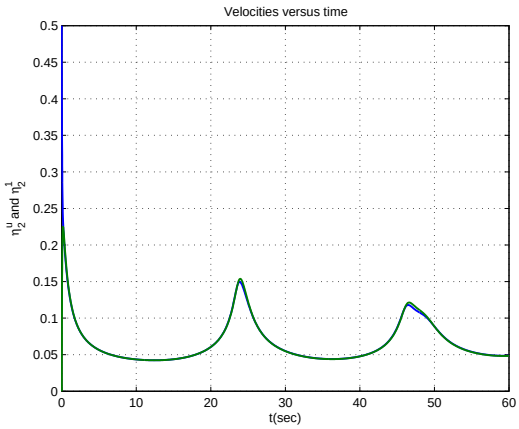
In this section, we present two sets of simulation results corresponding to two different position synchronization constraints.



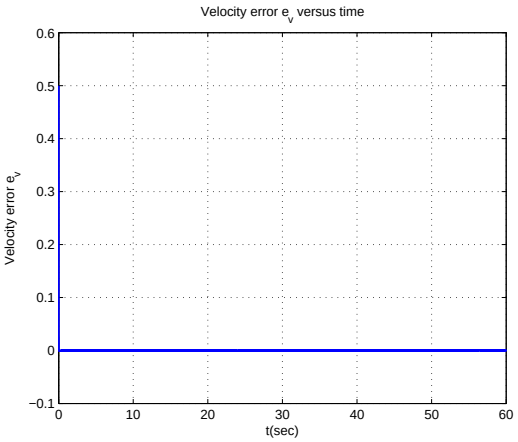
(a) The manipulator following an elliptic path.



(b) The unicycle following an elliptic path.



(c) The velocities are approaching the same.



(d) The velocity error e_v approaching zero.

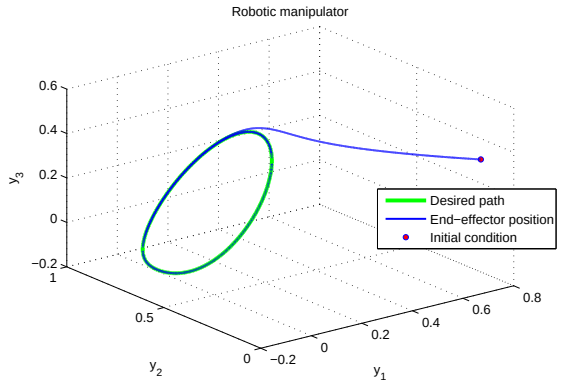
Figure 6.2: Velocity synchronized path follwing simulation results.

Linear position synchronization constraint

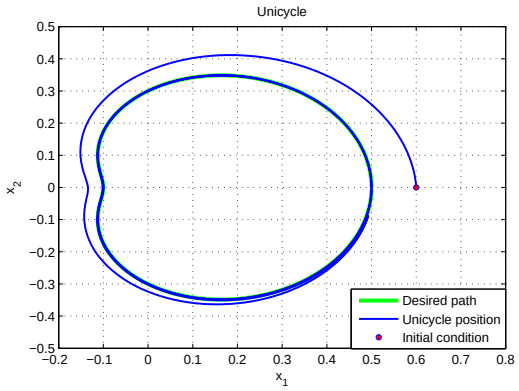
We choose $f : \mathbb{R} \bmod L_u \times \mathbb{R} \bmod L_r \to \mathbb{R}, f(\eta_1^u, \eta_1^1) = \eta_1^1 - \frac{L_r}{L_u}\eta_1^u$ with $L_u = 2.1010$ and $L_r = 1.6891$ as our synchronization constraint. The paths are selected as limacon curves:

$$\sigma_u : \mathbb{R} \to \mathbb{R}^2, \qquad \lambda_u \mapsto \begin{bmatrix} (0.2 + 0.3\cos(\lambda_u))\cos(\lambda_u) \\ (0.2 + 0.3\cos(\lambda_u))\sin(\lambda_u) \end{bmatrix}$$

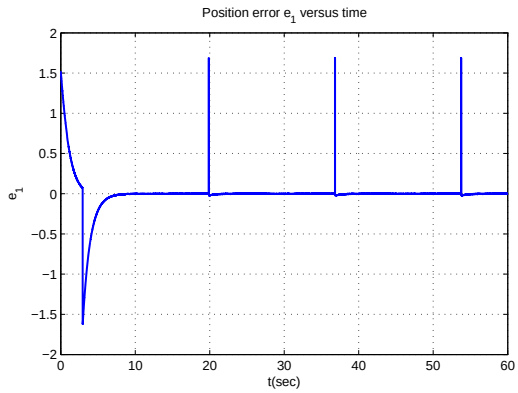
$$\sigma_r : \mathbb{R} \to \mathbb{R}^3, \qquad \lambda_r \mapsto \begin{bmatrix} (0.2 + 0.05\cos(\lambda_r))\cos(\lambda_r) \\ (0.2 + 0.05\cos(\lambda_r))\sin(\lambda_r) + 0.6 \\ (0.25 + 0.1\cos(\lambda_r))\cos(\lambda_r) \end{bmatrix}.$$



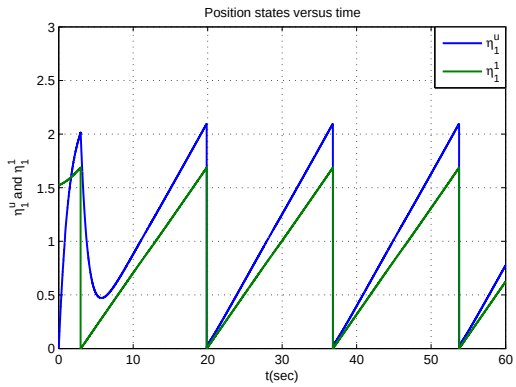
(a) The manipulator following a limacon path.



(b) The unicycle following a limacon path.



(c) Synchronization error.



(d) The revolution of η_1^u and η_1^1.

Figure 6.3: Position synchronization simulation results.

Figure 6.3(a) and 6.3(b) shows that the unicycle and robotic manipulator get on and follow their desired paths. The initial positions of the unicycle and the joints of the robotic manipulator are represented by solid dots. The simulation result shows the synchronization error converging to 0 in Figure 6.3(c) with tangential controllers chosen as

$$v_r^{\|} = -\left(\eta_1^2 - 0.1\right) - e_1$$
$$v_u^{\|} = \frac{L_u}{L_r}\left(v_r^{\|} - u_e\right)$$

The evolution of the tangential states η_1^u and η_1^1 versus time are shown in Figure 6.3(c). The jumps of e_1 in Figure 6.3(c) coincide with robots passing through their starting points ($\lambda_u = \lambda_r = 0$), thus the jumps do not spoil the continuity of e_1. Figure 6.3(c) and 6.3(d) show that the motions of robots satisfy the synchronization constraint.

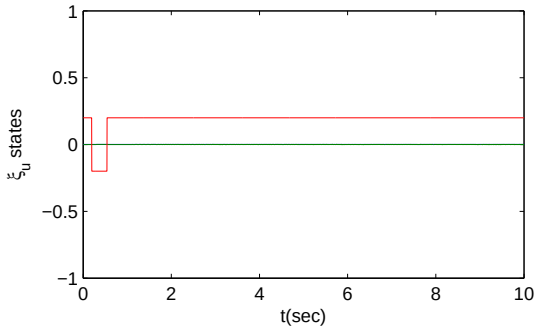


Figure 6.4: The path following manifold $\Gamma_u^\star$ is invariant under switching scheme.

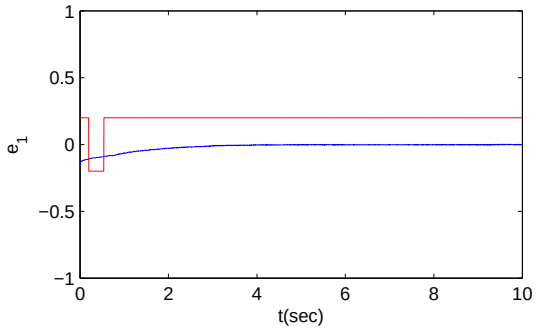


Figure 6.5: The synchronization error e_1 is continuous under the switching scheme.

In the second simulation, the systems are initialized on their path following manifolds but out of synchronization. We see the effect of switching control laws between (2.7) and (2.9) on the path following manifold $\Gamma_u^\star$ with singularity boundary $\delta = 0.01$ are shown in Figure 6.4 and Figure 6.5. The switching signal in Figure 6.4 and Figure 6.5 indicate when the control laws switch. Figure 6.4 shows that ξ_1^u and ξ_2^u remain 0 under the control law (2.9), indicating the path following manifold $\Gamma_u^\star$ is invariant under the switching scheme. In Figure 6.5 shows the convergence of the synchronization error to zero. These results show that the systems synchronize without ever leaving their respective paths.

Nonlinear position synchronization constraint

The simulation results in this part shows the positions of the systems satisfying a nonlinear constraint while they moving along their paths. The assigned paths are elliptic curves:

$$\begin{aligned}
\sigma_u :&\mathbb{R} \to \mathbb{R}^2, \qquad \lambda_u \mapsto \begin{bmatrix} 0.6\cos(\lambda_u) \\ 0.3\sin(\lambda_u) \end{bmatrix} \\
\sigma_r :&\mathbb{R} \to \mathbb{R}^3, \qquad \lambda_r \mapsto \begin{bmatrix} 0.4\cos(\lambda_r) \\ 0.4\sin(\lambda_r) \\ 0.2\cos(\lambda_r) + 0.07 \end{bmatrix}.
\end{aligned}$$

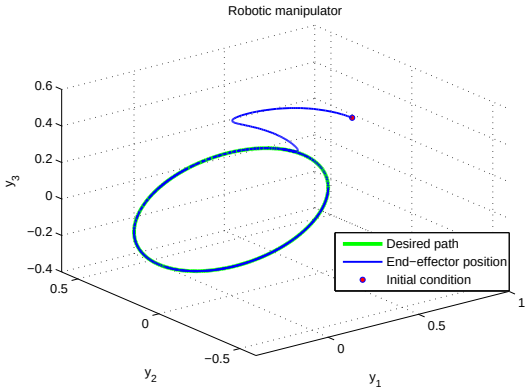
We choose

$$\begin{aligned}
f :&\mathbb{R} \bmod L_u \to \mathbb{R} \bmod L_r, \\
&\eta_1^u \mapsto \frac{L_r}{{L_u}^2}\eta_1^{u2}
\end{aligned}$$

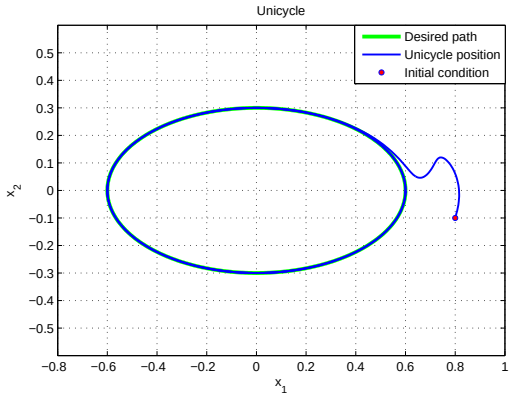
with $L_u = 2.9065$ and $L_r = 3.1622$ as our synchronization constraint. The synchronization simulation results of the nonlinear constraint is shown in Figure 6.6. The synchronization error converging to 0 in Figure 6.6(c) with tangential controllers chosen as

$$\begin{aligned}
v_u^{\parallel} &= -\left(\eta_2^u - 0.2\right) - \left(\eta_1^1 - \frac{L_r}{{L_u}^2}\eta_1^{u2}\right) \\
v_r^{\parallel} &= u_e + 2\frac{L_r}{{L_u}^2}\eta_2^{u2} + \frac{L_r}{{L_u}^2}\eta_1^u v_u^{\parallel}.
\end{aligned}$$

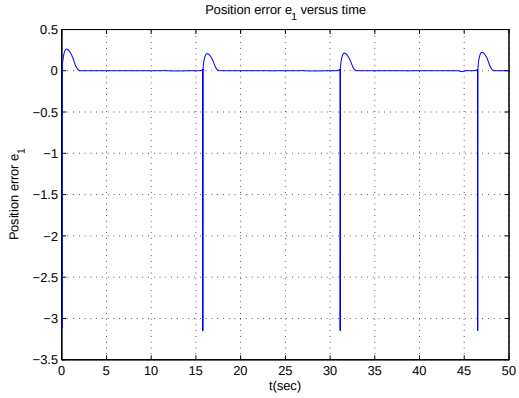
The jumps of e_1 in Figure 6.6(c) coincide with the η_1 states passing through the starting points ($\lambda_u = \lambda_r = 0$), see Figure 6.6(d). Similarly to the linear constraint case, we present the simulation results of the switching scheme occurs on the path following manifolds in Figure 6.7 and Figure 6.8 by initializing the systems on their path following manifolds but out of synchronization. We use the same singular region $M_0 = \{x \in \mathbb{R}^4 : |x_4| \leq 0.01\}$ and the simulation results show that the system remain on their path following manifolds synchronizing their positions when the switching occurs.



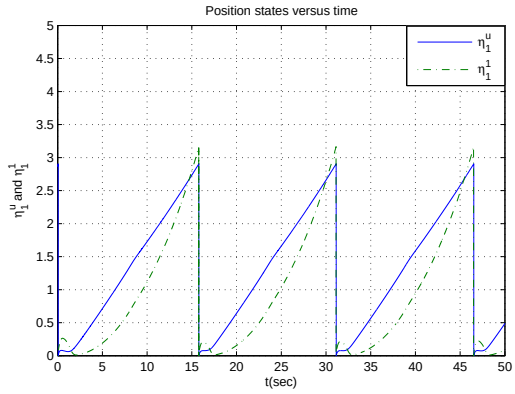
(a) The manipulator following an elliptic path.



(b) The unicycle following an elliptic path.



(c) Synchronization error.



(d) The revolution of η_1^u and η_1^1.

Figure 6.6: Position synchronization simulation results.

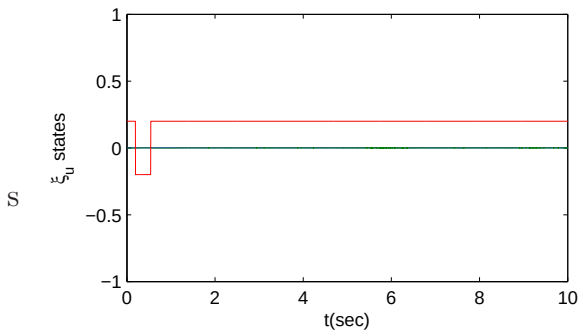


Figure 6.7: The path following manifold $\Gamma_u^\star$ is invariant under switching scheme.

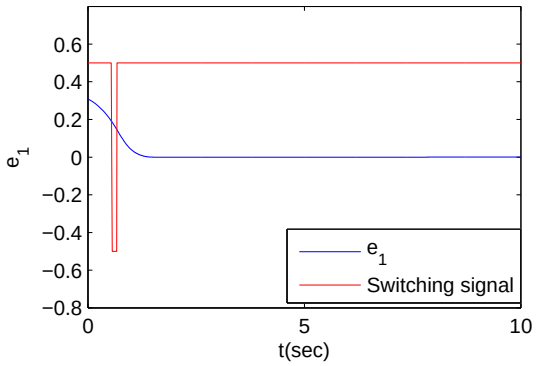


Figure 6.8: The synchronization error e_1 is continuous under the switching scheme.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

In this thesis we introduced a synchronized path following problem for a general nonlinear control system and an Euler-Lagrange system. We proposed a control design method for the special case when the nonlinear system is a differential drive ground vehicle modeled as a dynamic unicycle and the Euler-Lagrange system is fully actuated and not kinematically redundant. Transverse feedback linearization was applied to transform these nonlinear systems into linear and controllable systems, which facilitated the control design.

The nonlinear system of the dynamic unicycle was feedback linearized in Chapter 2. However, this procedure creates singularities when the translational velocity of the unicycle is zero. We introduced an alternate set of coordinates in the unicycle's singular region where the primary transformation is not well defined. In Chapter 3 we reviewed the conditions under which transverse feedback linearization is applicable to a general Euler-Lagrange system. In the case of fully actuated systems that are not kinematically redundant, the system can be fully feedback linearized. In order to implement our controllers we require both parametric and implicit forms of a given curve. An approach for constructing implicit representations of a given parametric Jordan curves were proposed in Chapter 4 using Serret-Frenet frame.

Path following controller design was investigated in Chapter 5 for the unicycle and the Euler-Lagrange system. For the unicycle, we discussed the path following design both in its singular and nonsingular region and a comparison with respect to the two different path following control laws was shown via simulation. After the systems get on their path

following manifolds, the tangential dynamics model the motions along the paths. The control design of synchronization problems including velocity and position synchronization was analysed in Chapter 5. The main results in this thesis were applied to a unicycle and a 4-DOF robotic manipulator and the effectiveness of the proposed control laws was shown via simulation in Chapter 6.

7.2 Future Work

Future work includes extending the synchronized path following design in this thesis to the cases where there are communication constraints. We assumed full communication between systems in this thesis. It would be interesting to investigate the synchronization problems in the situations in which the inter-system communication is limited. We would like to use state estimation or build observers to estimate the necessary tangential states that are not exchanged via communication. The general synchronized path following problem allows for a very general control of an affine nonlinear system. We focused on the case where this system is a differential drive robot. It would be useful to obtain a characterization of the class systems for which the approach of this thesis is feasible and apply the approach to a group of N systems, for $N \geq 2$. For instance, non-holonomic systems. Furthermore, in this thesis, we restrict our paths to simple closed curves. Extending this work to non-closed and self-intersecting paths would be a contribution.

A practical contribution of future experimental work is to apply the synchronized path following results to laboratory equipment at the University of Waterloo, i.e., a differential drive ground vehicle (Chameleon R200) and a 4-DOF robotic manipulator (J10-WAT07). The path following controllers have been implemented using the Robot Operating System and tested in simulation. In other words, the software is already in place for the Chameleon R200 following a unit circle as its desired path and the experimental preparation for the manipulator J10-WAT07 has been set up. However, we were unable to carry out our experiments due to some technical issues with the manipulator hardware. Once those issues are solved, it would be interesting, and straight forward, to complete our experimental work.

References

[1] A. P. Aguiar, D. B. Dačić, J. P. Hespanha, and P. Kokotović. Path-following or reference-tracking? An answer relaxing the limits to performance. *International Federation of Automatic Control*, 2004.

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

[3] A. P. Aguiar, J. P. Hespanha, and P. V. Kokotović. Performance limitations in reference tracking and path following for nonlinear systems. *Automatica*, 44(3):598–610, 2008.

[4] L. E. Aguilar, T. Hame, and P. Soukre. Robust path following control for wheeled robots via sliding mode techniques. *Intelligent Robots and Systems*, 3:1389–1395, 1997.

[5] A. Akhtar. Dynamic path following controllers for planar mobile robots. *Master's thesis*, University of Waterloo, 2011.

[6] A. Akhtar and C. Nielsen. Path following for a car-like robot using transverse feedback linearization and tangential dynamic extension. *IEEE Conference on Decision and Control*, pages 7974–7979, December 2011.

[7] C. Altafini. Following a path of varying curvature as an output regulation problem. *IEEE Transactions on Automatic Control*, 47(9):1551–1556, September 2002.

[8] A. Rodriguez Angeles. Synchronization of mechanical systems. *Technische Universiteit Eindhoven*, 2002.

[9] H. Arai, T. Takubo, Y. Hayashibara, and K. Tanie. Human-robot cooperative manipulation using a virtual nonholonomic constraint. *IEEE International Conference on Robotics and Automation*, 4:4063–4069, April 2000.

[10] A. Astolfi, P. Bolzern, and A. Locatelli. Path-tracking of a tractor-trailer vehicle along rectilinear and circular paths: A lyapunov-based approach. *IEEE Transactions on Robotics and Automation*, 20(1):154–160, February 2004.

[11] S. P. Bhat and D. S. Bernstein. Continuous finite-time stabilization of the translational and rotational double integrators. *IEEE Transactions on Automatic Control*, 43(5):678–682, May 1998.

[12] Y. Bouteraa, J. Ghommam, N. Derbel, and G. Poisson. Adaptive synchronization control of multi-robot teams: Cooperative and coordinated schemes. *2010 18th Mediterranean Conference on Control and Automation*, pages 586–591, 2010.

[13] Y. Bouteraa, J. Ghommam, G. Poisson, and N. Derbel. Distributed synchronization control to trajectory tracking of multiple robot manipulators. *Journal of Robotics*, 2011.

[14] Y. Bouteraa, L. Ghommam, N. Derbel, and G. Poisson. Mutual and external synchronization control of multi-robot system. *International Multi-Conference on Systems, Signals and Device*, pages 1–6, 2010.

[15] Niklaus Burger. Control of a robot manipulator. *ECE499 Project*, 2012.

[16] L. Consolini, M. Maggiore, C. Nielsen, and M. Tosques. Path following for the PVTOL aircraft. *Automatica*, 46:1284–1296, August 2010.

[17] D. B. Dačić and P. V. Kokotović. Path-following for linear systems with unstable zero dynamics. *Automatica*, 42(10):1673 – 1683, 2006.

[18] C. C. de Wit, B. Siciliano, and B. Bastin. *Theory of Robot Control*. Springer, New York, 3^{rd}*edition*, 1997.

[19] M. I. El-Hawwary and M. Maggiore. Global path following for the unicycle and other results. *American Control Conference*, pages 3500–3505, 2008.

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

[21] D. Gates. Boeing using robots to boost 777 output. *The Seattle Times*, May 29 2013. http://seattletimes.com/html/businesstechnology/2021082952_boeing777xml.html.

[22] J. Ghommam, H. Mehrjerdi, and M. Saad. Coordinated path-following control for a group of mobile robots with velocity recovery. *Journal of Systems and Control Engineering*, 224(8):995–1006, December 2010.

[23] V. Guillemin and A. Pollack. *Differential Topology*. Prentice Hall, New Jersey, 1974.

[24] J. Hauser and R. Hindman. Maneuver regulation from trajectory tracking: Feedback linearizable systems. *IFAC Symposium on Nonlinear Control Systems Design*, 1995.

[25] A. Hladio. Path following for mechanical systems applied to robotic manipulators. *Master's thesis*, University of Waterloo, 2010.

[26] A. Hladio, C. Nielsen, and D. Wang. Path following for a class of mechanical systems. *IEEE Transactions on Control Systems Technology*, To appear. DOI: 10.1109/TCST.2012.2223470.

[27] I. F. Ihle, M. Arcak, and T. I. Fossen. Passivity-based designs for synchronized path-following. *Automatica*, 43(9):1508–1518, 2007.

[28] A. Isidori. *Nonlinear Control Systems*. London,U.K.:Springer-Verlag, 1995.

[29] H. K. Khalil. *Nonlinear systems*. Prentice Hall,, 2002.

[30] T. G. Kim and D. H. Cho. Performance analysis of synchronization communication protocols for real-time multimedia services. *Communications*, 1:256–260, 1995.

[31] W. Kuhnel. *Differential Geometry*. Student Mathematical Library, 2002.

[32] E. Kyrkjebo and K. Y. Pettersen. Output synchronization control of Euler-Lagrange systems with nonlinear damping terms. *IEEE Conference on Decision and Control*, pages 4951–4957, 2005.

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

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

[35] Y. Liu and N. Chopra. Controlled synchronization of heterogeneous robotic manipulators in the task space. *IEEE Transactions on Robotics*, 28(1):268–275, 2011.

[36] A. C. J. Luo. A theory for synchronization of dynamical systems. *Communications in Nonlinear Science and Numerical Simulation*, 14(5):1901C1951, May 2009.

[37] D. E. Miller and R. H. Middleton. On limitations to the achievable path tracking performance for linear multivariable plants. *IEEE Transactions on Automatic Control*, 52(11):2586–2601, 2008.

[38] A. Morro, A. Sgorbissa, and R. Zaccaria. Path following for unicycle robots with an arbitrary path curvature. *IEEE Transaction on Robotics*, 27(5):1016–1023, October 2011.

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

[40] C. Nielsen and M. Maggiore. Maneuver regulation via transverse feedback linearization: Theory and examples. *Symposium on Nonlinear Control Systems*, September 2004.

[41] S. Palchaudhuri, A. K. Saha, and D. B. Johnsin. Adaptive clock synchronization in sensor networks. *Information Processing in Sensor Networks*, pages 340 – 348, April 2004.

[42] A. Pressley. *Elementary Differential Geometry*. Springer-Verlag London, 2002.

[43] C. C. Pugh. *Real Mathematical Analysis*. Springer, New York, 2002.

[44] X. Qian, X. Shen, G. Dai, J. Zhang, and C. Lv. Clapping and broadcasting synchronization in wireless sensor network. *Mobile Ad-hoc and Sensor Networks*, 16(6):632–639, December 2010.

[45] L. Ren, J. K. Mills, and D. Sun. Adaptive synchronization control of a planar parallel manipulator. *American Control Conference*, 5:3980–3985, June 30-July 2 2004.

[46] C. H. Rentel and T. Kunz. A mutual network synchronization method for wireless ad hoc and sensor networks. *IEEE Transactions on Mobile Computing*, 7(5):633–646, May 2008.

[47] A. Rodriguez-Angeles and H. Nijmeijer. Mutual synchronization of robots via estimated state feedback: A cooperative approach. *IEEE Transactions on Control Systems Techonology*, 12(4):542–554, July 2004.

[48] L. Scardovi, M. Arcak, and E. D. Sontag. Synchronization of interconnected systems with applications to biochemical networks: An input-output approach. *IEEE Transactions on Automatic Control*, 55(6):1367–1379, 2004.

[49] M. Scheint, M. Sobotka, and Martin Buss. Virtual holonomic constraint approach for planar bipedal walking robots extended to double support. *IEEE Conference on Decision and Control*, pages 8180–8185, 2009.

[50] H. Si-yuan. The research of synchronization communication technology based on chaotic masking. *Intelligent Computing and Intelligent Systems*, 3:267 – 270, November 2009.

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

[52] R. Skjetne, T. I. Fossen, and P. V. Kokotovic. Robust output maneuvering for a class of nonlinear system. *Automatica*, 40(3):373–383, March 2004.

[53] M. W. Spong, S. Hutchinson, and M. Vidyasagar. *Robot Modeling and Control*. John Wiley & Sons, 2006.

[54] S. H. Strogatz. From kuramoto to crawford: Exploring the onset of synchronization in populations of coupled oscillators. *Physica*, 143:1–20, September 2000.

[55] D. Sun. Position synchronization of multiple motion axes with adaptive coupling control. *Automatica*, 39(6):997–1005, 2003.

[56] D. Sun and J. K. Mills. Adaptive synchronized control for coordination of two robot manipulators. *IEEE International Conference on Robotics and Automation*, 1:976–981, May 2002.

[57] D. Sun, C. Wang, W. Shang, and G. Feng. A synchronization approach to trajectory tracking of multiple mobile robots while maintaining time-varying formations. *IEEE Transaction on Robotics*, 25(5):1074–1086, October 2009.

[58] J. A. K. Suykens, P. F. Curran, and L. O. Chua. Robust synthesis for master-slave synchronization of Lur'e systems. *IEEE Transactions on Circuits and Systems*, 46(7):841–850, 1999.

[59] T. Takubo, H. Arai, and K. Tanie. Virtual nonholonomic constraint for human- robot cooperation in 3-D space. *IEEE International Conference on Intelligent Robots and Systems*, 1:300–305, 2005.

[60] M. Tillerson, L. Breger, and J. P. How. Distributed coordination and control of formation flying spacecraft. *Proceedings of the American Control Conference*, 2:1740–1745, June 2003.

[61] C. J. Tomlin and S. S. Sastry. Switching through singularities. *IEEE Conference on Decision and Control*, 1:1–6, 1997.

[62] P. K. C. Wang, F. Y. Hadaegh, and K. Lau. Synchronized formation rotation and attitude control of multiple free-flying spacecraft. *Journal of Guidance, Control, and Dynamics*, 22(1):28–35, 1999.

[63] X. Zhang, A. Behal, D. M. Dawson, and J. Chen. A novel passive path following controller for a rehabilitation robot. *IEEE Conference on Decision and Control*, 5:5374–5379, 2004.

[64] W. Zhu. On adaptive synchronization control of coordinated multirobots with flexible/rigid constraints. *IEEE Transactions on Robotics*, 21(3):520–525, 2005.