Perception Based Navigation for Underactuated Robots

by

Gabriel A. D. Lopes

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering: Systems)
in The University of Michigan
2008

Doctoral Committee:
Professor Daniel E. Koditschek, Co-Chair
Professor Jessy W. Grizzle, Co-Chair
Professor Anthony M. Bloch
Professor Stéphane Lafortune
Associate Research Professor Alfred A. Rizzi

To my parents.

ACKNOWLEDGMENTS

For the past years I was very fortunate to work with a group of fantastic researchers, professors, and colleagues.

I would first like to thank my advisor, Daniel Koditschek, for his continuous enthusiasm in science. I have learned a lot from him. Most importantly, he taught me how to think from a broader perspective, how to be very precise and formal, how to approach different audiences, and how to be critical and expose my opinions. I also thank him for his infinite patience and honesty, and for believing in me throughout this time.

I would also like to thank my committee members Jessy Grizzle, Anthony Bloch, Stéfane Lafortune and Alfred Rizzi. I have learned a great deal from them, either from taking their classes, reading their books, or working together in projects.

Working in Dan's group has been an immensely rewarding intellectually rich experience. I thank my mentors Noah Cowan, for introducing me to the laboratory culture; Rick Groff for the philosophical discussions we had; Greg Sharp for his tremendous patience and teaching me how to debug and compile any code; Uluc Saranli and Eric Klavins for their insightful opinions. I would also like to thank my colleagues at the University of Michigan and University of Pennsylvania: Pei Chun-Lin for his meticulous and systematic approach, Haldun Komsuoglu, for his passion, Joel Weingarten for his enthusiasm, and also Richard Altendorfer and Jonathan Clark.

I would like to thank my portuguese colleagues for the always interesting discussions: José Costa, Rui Castro, Paulo Tabuada, Eduardo Silva and Pedro Granja.

Finally I would like thank my girlfriend Emile Yane, I owe her my deepest gratitude. She has been a driving force. She has been there with love and support through good and bad times and kept me cheerful for the past few years. All the nights she stayed awake next to me while I worked, all the places and foods we experienced together, all the culture she introduced to me, for those I will forever thank her. She is the love of my life, my Tang empress.

The work on this thesis was supported by The University of Michigan, The University of Pennsylvania, by DARPA/ONR N00014-98-1-0747, by DARPA/SPAWAR N660011-03-C-8045 and by Fundação para a Ciência e Tecnologia - Portugal, with the fellowship PRAXIS XXI/BD/18148/98.

AGRADECIMENTOS

Gostaria de agradecer a toda a minha família e amigos que me ensinaram as experiências da vida e me ajudaram a criar a minha personalidade. Os meus amigos de infância: Nuno Carda, Pedro Pereira, Pedro Varela e muitos outros. À minha família: os tios Luís e Marlene Delgado, Rui e Paulinha, os tios Ricardo e Gracinda Delgado e Fatinha. E principalmente à minha irmã Daniela e aos meus pais Adelino e Odete Delgado Lopes, por todos os sacrifícios que fizeram para eu poder realizar os meus sonhos e por me ensinarem desde pequeno o amor pela ciência, arte e filosofia. Muito obrigado!!!!!!

TABLE OF CONTENTS

CHAPTERS

LIST OF TABLES

Table

LIST OF FIGURES

Figure

LIST OF APPENDICES

APPENDIX

NOMENCLATURE

Functions

φ	Navigation function, page 24
f_1	Energy decreasing function, page 28
f_2	Energy preserving function, page 29
$\bar{f}_2$	σ scaled energy preserving function, page 35
σ	Center manifold destabilizing function, page 35
ξ	Pinching function, page 31
c	Camera map, page 16
ϱ	Ray function, page 65
X_i	Vector field, page 6
L	Lagrangian, page 4
P	Poincaré function, page 36
V	Control Lyapunov function, page 40

Matrices

A	Constraints matrix, page 5
B	Actuation matrix, page 5
H	Nonholonomic projection matrix, page 27
Q	Converse nonholonomic projection matrix, page 31
M	Mass/inertia matrix, page 5
K	Positive definite matrix, page 18
J	Skew symmetric matrix, page 32
R_α	Rotation matrix, page 64
L_s	Iteraction matrix, page 17

Operators

T	Tangent space, page 6
$\mathcal{T}$	Tangent bundle, page 53

Φ_t^f	Flow, page 29
$L_X Y$	Directional derivative, page 29
$L_g h$	Left translation on Lie group, page 9
$R_g h$	Right translation on Lie group, page 9

Spaces

G	Lie group, page 8
$\mathcal{M}$	Base space, page 8

Sets

$\mathcal{Q}$	Configuration manifold, page 4
$\mathcal{Q}_\epsilon$	Goal neighborhood set, page 31
$\mathcal{Q}_s$	Hollow local surround set, page 31
$\mathcal{Q}_1$	Outer shell set, page 31
$\mathcal{W}^c$	Center manifold, page 23
$\mathcal{W}^u$	Unstable manifold, page 23
$\mathcal{W}^s$	Stable manifold, page 23
$\widehat{\mathcal{W}}^s_k$	Stable manifold approximation of order k, page 41
$\mathcal{R}$	Reachable set or "local surround", page 36
$\mathcal{Z}$	Unreachable set, page 36
$\mathcal{C}_\gamma$	Solid cone, page 34
$\mathcal{C}_\gamma^c$	Complement of solid cone, page 34
$\mathcal{L}_\alpha$	Level set, page 34
$\mathcal{D}$	Domain of attraction, page 26
$\mathcal{G}$	Goal domain, page 26
$\mathcal{B}$	Beacon set, page 63
$\mathcal{P}$	Intermediate set, page 64
$\mathcal{F}$	Facing set, page 65
$\mathcal{V}$	Visible set, page 65
$\mathcal{W}$	Workspace, page 66
$\mathcal{I}$	Camera image line, page 64
$\mathcal{O}$	Obstacle set, page 66
M	Manifold, page 22
U	Neighborhood, page 6

Variables

q	State variable, page 4

q^*	Goal state, page 24
q_0	Initial state, page 6
q_s	Switching state, page 45
q_f	Final state, page 6
u	System input, page 5
g	Group variable, page 8
e	Error associated with visual features, page 16
s	Features variable, page 16
φ^*	Target level set, page 77
ζ_i	Pinhole projection camera ray angle, page 64
ι_i	Ray projection into camera image plane, page 64
δ	Outer neighborhood scalar parameter of the Center manifold $\mathcal{W}^c$, page 30
γ	Inner neighborhood scalar parameter of the Center manifold $\mathcal{W}^c$, page 30
ϵ	Goal neighborhood scalar parameter , page 30
λ_j	Lagrange multipliers, page 4

Others

Δ	Distribution, page 6
F	Foliation, page 23
$\mathrm{B}_r(o)$	Ball of center o and radius r, page 22

ABSTRACT

Robot autonomous navigation is a very active field of robotics. In this thesis we propose a hierarchical approach to a class of underactuated robots by composing a collection of local controllers with well understood domains of attraction.

We start by addressing the problem of robot navigation with nonholonomic motion constraints and perceptual cues arising from onboard visual servoing in partially engineered environments. We propose a general hybrid procedure that adapts to the constrained motion setting the standard feedback controller arising from a navigation function in the fully actuated case. This is accomplished by switching back and forth between moving "down" and "across" the associated gradient field toward the stable manifold it induces in the constrained dynamics. Guaranteed to avoid obstacles in all cases, we provide conditions under which the new procedure brings initial configurations to within an arbitrarily small neighborhood of the goal. We summarize with simulation results on a sample of visual servoing problems with a few different perceptual models. We document the empirical effectiveness of the proposed algorithm by reporting the results of its application to outdoor autonomous visual registration experiments with the robot RHex guided by engineered beacons.

Next we explore the possibility of adapting the resulting first order hybrid feedback controller to its dynamical counterpart by introducing tunable damping terms in the control law. Just as gradient controllers for standard quasi-static mechanical systems give rise to generalized "PD-style" controllers for dynamical versions of those standard systems, we show that it is possible to construct similar "lifts" in the presence of non-holonomic constraints notwithstanding the necessary absence of point attractors. Simulation results corroborate the proposed lift.

Finally we present an implementation of a fully autonomous navigation application for a legged robot. The robot adapts its leg trajectory parameters by recourse to a discrete gradient descent algorithm, while managing its experiments and outcome measurements autonomously via the navigation visual servoing algorithms proposed in this thesis.

CHAPTER 1

Introduction

The ability to navigate is a fundamental aspect to the successful survival of an animal species. Ranging from bird or mammal migration to bees returning to their honeycombs, navigation is the next fundamental step once locomotion is achieved. Throughout the years, research has lead us to believe that animals use various navigation techniques for both the *local* scale, e.g. locating food nearby or achieving manipulation, and the *global* scale, as in migration. In particular, bees are known to modify their global navigation mechanism when returning to the honeycomb, depending on the weather [DG81, Gou86, Gou98, MGS+05]. They can switch from a landmark based approach to detecting the polarization of the sun's light in the sky. Not only are the sensors different but the navigation paradigm also changes.

In the robotics community it is common to accept the navigation paradigms to be metric (also called grid-based by some authors [Thr98]), topological or hybrid. Parallels can be found in nature for these paradigms, most notably the human: local navigation, typically metric, is accomplished using vision or touch. Global navigation, typically topological, uses landmarks, e.g. doors, sidewalks, street names, hills, trees, etc.

Metric navigation — In a perfect metric world a robot navigates by knowing the exact distances away from landmarks, obstacles and its desired position. When doing manipulation, in general the important parameters are the joint limits and geometry of the objects being manipulated. In practice, due to the difficulty of exactly representing or parameterizing the surrounding environment, it is common to use simplifications based on either *deterministic* or *stochastic* approaches.

Stochastic approaches for metric navigation arise naturally due to the imperfect nature of sensor measurements. When navigating in unknown environments and relying solely on onboard sensors, it is reasonable to represent the world as a grid and the position of the robot or manipulating arm as a probability distribution function in that grid. This type of modeling has been successfully applied in many real world robotic applications using occupancy grids as in [Elf89, AdRMF99, Ols00, SKP05, BC06] or non-parametric approaches such as particle filters as in [FTBD01, CLLH07, GSB07].

The deterministic approach normally relies on finding an approximate model for the surrounding environment or workspace. Mathematically this is accomplished by defining the configuration space of the robot to be an embedded manifold in the Euclidean space. (We define *configuration space* or *configuration manifold* to be the obstacle-free locations where the robot can safely navigate and/or manipulate). Koditschek and Rimon [Kod87c, RK92] developed a class of general bounded n-dimensional "workspaces" that result in compact manifolds. We use this representation of the world throughout this thesis. Although the deterministic approach cannot model the sensor error and uncertainty, using techniques such as visual servoing for robot navigation can result in excellent practical results. Moreover, there are many situations where the obstacles that either dent or puncture the configuration space are actually known à priori, such as field of view obstacles for cameras or joint limits for actuators. In these situations one can find closed form representations for the configuration space since the obstacles do not change over time nor are they acquired by exteroceptive sensors. The deterministic approach also has the advantage of being able to allow for a tight integration between the robot's motion model dynamics and its energy exchanging interaction with the environment by describing the world as compact manifolds. This way, a large body of work on differential geometry spanning back to the sixteenth century can be applied throughly. In the subsequent sections we provide solid argumentation for the real work utility of this approach.

Although the control strategy used for navigation is intimately related to the representation of the configuration space, it is important to distinguish between navigation and the map building process. The recent work accomplished on SLAM (Self localization and map building) [DWB06, BDW06] merges the two concepts by implementing navigation with the purpose of building a map of the environment and vice versa. However, in many applications a map can be provided à priori giving more freedom to choose the types of control strategies that can be utilized for metric navigation. In this thesis we use that assumption and rely on well structured environments, resulting in very robust feedback controllers that, although being modeled for continuous wheeled robots, perform very well in a discretely actuated legged robot.

Topological navigation describes the world from a graph theory point of view. Each node represents a particular situation, location or landmark of the environment, and each arc represents a direct path between two landmarks. This strategy becomes very relevant when the configuration space of the robot is very large and complex. Implementation results can be found in [Mat90, KB91, EM92, KW94, PK94, Tor94, YB96, Zim96].

Hybrid topological/metric navigation — Given the different strengths and weaknesses of both metric and topological approaches, it is common to use both in a hybrid way. See Thrun [Thr98] for a thorough comparison of both approaches. In this framework one decouples the normally locally continuous dynamics dealing with manipulation, obstacle avoidance and basic locomotion, from the normally global topological navigation, dealing with the high level "map reading". In general the classes of algorithms utilized in such a hybrid way arise from different fields of mathematics.

In this thesis we start by introducing a novel class of robust feedback controllers for local outdoor navigation of a legged robot guided only by visual cues. Conceptually, there are three broad problems associated with this task. First, the requirement for perceptually reliable landmarks, an instance of the long-standing "early vision" problem that we explicitly avoid by engineering the visual beacons that comprise the physical landmarks. Second, the transformation of discrepancies between perceived and desired visual landmark appearance into feedback forces capable of "safely" correcting the errors in pose that cause them is effected by a monocular camera via a slightly generalized extension of prior work. Finally, the effective application of these restoring forces in a manner that respects both the constrained control authority over rigid body motion afforded by a legged gait and the perceptual requirements.

In Chapter 3 we pursue a solution linking the second and third problem by encoding the perceived discrepancies in terms of artificial potential functions. For fully actuated systems, potential-dissipative force fields offer a natural and direct generalization of linear proportional-derivative servo control for general mechanical systems [Kod91]. The wide popularity of such PD controllers attests to their robustness against sensor noise and imperfect models. However, for underactuated systems when the number of independently actuated degrees of freedom decreases relative to the dimension of the total configuration space, there is no general method for applying PD control. For autonomous outdoor robots, it is crucial to develop perception-driven controllers, yet in consequence of ubiquitous power-to-weight limitations, autonomous robots are intrinsically underactuated. Hence, we draw the greatest practical motivation for extending PD methods to underactuated settings precisely in such contexts as visual servoing for a rugged and underactuated outdoor vehicle like the hexapod, RHex [SBK01].

This thesis presents an extension of PD control to the class of two-actuator, three-degree-of-freedom mechanical systems that includes the simple "unicycle" kinematics crudely descriptive of the horizontal plane behavior of RHex. By so modeling the robot as a drift-free constrained kinematic system and by treating the perceptual limitations of a monocular camera observing the robot's horizontal plane pose as incurring obstacles in the robot's configuration space, we arrive at the formal problem of set point regulation in the face of simultaneously nonholonomic motion constraints and holonomic perception constraints. We approach this problem by merging aspects of the work developed for nonholonomic systems with well verified visual servoing algorithms. The next two sections give an account on these two topics.

1.1 Nonholonomic mechanics

Bloch et al. [Blo03] presents an in-depth bibliographic review and history of nonholonomic systems from a mathematician's perspective. In this introduction we put more emphasis on the applied control for robotic platforms. We start by introducing the main concepts of nonholonomic mechanics and present the most general set of equations with nonholonomic constraints. We then proceed by adding structure until reaching specific robotic applications, discussing the contemporary bibliography throughout the exposure. Lie algebra theory is a very common tool on the study of nonholonomic systems, however, we choose not include a thorough description of its definition in this thesis since it is only relevant for the introduction and future work. For a reference on Lie groups and Lie algebras see [Cur70, DK99, Bak02, Hal03, EW06].

From Bloch et al. [Blo03], the study of nonholonomic systems dates back to the late nineteen century when Voss [Vos85] and Hertz [Her94] introduced the distinction between holonomic and nonholonomic systems. Vierkandt [Vie92] and Chaplygin [Cha97] provided the first analysis of the rolling disk and sphere, traditional examples of nonholonomic systems. The accepted general formulation for systems with nonholonomic constraints, after Korteweg [Kor99], is the Lagrange–d'Alembert equations that we review below. Let $q = (q^1, \dots, q^n)$ be generalized coordinates in the configuration manifold $\mathcal{Q}$. Consider a system with constraints given by the following m equations, linear in the velocity field, where $m < n$ and $j = 1, \dots, m$:

$$\sum_{i=1}^{n} a_i^j(q)\dot{q}^i = 0. \tag{1.1}$$

If there exist functions h^j that only depend on the positions, such that $h^j(q) = 0$, and its time derivative

$$\sum_{i=1}^{n} \frac{\partial h^j}{\partial q^i}\dot{q}^i = 0, \tag{1.2}$$

determine the same constraint distribution as the constraints (1.1), then one says that the constraints are *holonomic* or *integrable.* In this case the system lives in a submanifold of the configuration manifold. This submanifold is defined implicitly by the set of equations $h^j(q) = 0$. If no such functions h^j exist then the constraints are called *nonholonomic* or *nonintegrable.* The *Lagrange–d'Alembert equations*, also called *nonholonomic equations*, take the form

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}^i} - \frac{\partial L}{\partial q^i} = \sum_{j=1}^{m} \lambda_j a_i^j, \tag{1.3}$$

together with equation (1.1), where λ_j are the Lagrange multipliers. For the derivation of the previous equation see Bloch et al. [Blo03].

1.2 Nonholonomic Control

The philosophy behind nonholonomic systems lies in controlling the unactuated state variables by appropriately steering the actuated subset of the state variables, normally denoted *base* or *shape* variables. Since in general the base variables can be readily stabilized, it is common to build controllers that “loop” in the *base space* (the invariant manifold of the base variables). Researches have approached this problem from different perspectives, notably using classical nonlinear control tools such as Lyapunov analysis, or recurring to geometric control using Lie group/algebra techniques. The controlled version of the Lagrange–d'Alembert equations (1.3) is:

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}^i} - \frac{\partial L}{\partial q^i} = \sum_{j=1}^{m} \lambda_j a_i^j + \sum_{j=1}^{l} b_i^j u_j, \tag{1.4}$$

$$\sum_{i=1}^{n} a_i^j(q)\dot{q}^i = 0 \qquad j = 1, \ldots, m. \tag{1.5}$$

Some of the work in controllability and stability of the controlled Lagrande–d'Alembert include Sussmann [SJ72], Hermann [HK77], and Murray [MS93, MLS94]. When the Lagrangian L is defined for a mechanical system by taking the form of $K - V$, i.e. Kinetic energy minus the Potential energy, and the Kinetic energy takes its traditional quadratic form $K = \dot{q}^T M(q)\dot{q}$, one can rewrite equations (1.4),(1.5) using the *nonholonomic mechanical system equations*:

$$M(q)\ddot{q} + c(q, \dot{q}) = A^T(q)\lambda + B(q)u \tag{1.6}$$

$$A(q)\dot{q} = 0 \tag{1.7}$$

Bloch et al. [BR92] is a classical reference on stabilization and control using this set of equations. Contemporary applied articles that follow this formalism include Wang [WGL04], Dong [DK05], Beji [BB05], and Li [LGM07] for adaptive control; Oya [OSK03], Chwa [Chw04], Anupoju [ASO05], Chang [CC05], Coelho [CN05], Tsai [TWC06], and Tanner [Tan06] in tracking control. Other authors use this set of equations in specific topics including Lin in fuzzy control [LHC05], Faulring in a haptic display modeling [FLCP07], Ghommam in a vessel stabilization [GaABD06] and Salermo in a Segway-style vehicle control [SA07].

Equations (1.6)-(1.7) encapsulate both the dynamical and kinematic aspects of nonholonomic systems. In many practical cases, however, it is reasonable to ignore the dynamics, i.e. the masses and inertias, and analyze only the kinematic equations. The most commonly used wheeled robotic systems result in the *nonholonomic affine kinematic* model,

$$\dot{q} = f(q) + g(q)u, \tag{1.8}$$

or the *nonholonomic drift-free kinematic* model,

$$\dot{q} = B(q)u. \tag{1.9}$$

1.2.1 Aspects of controllability

The general notion of controllability (existence of an admissible input $u(t), t \in [0, T]$ such that $q(0) = q_0$ and $q(T) = q_f$, $q_0, q_f \in \mathcal{Q}$) is in general very difficult to verify for arbitrary systems. The properties of *accessibility* and *reachable sets* have been introduced to address the local structure of the system at hand. See Sussmann [SJ72] and Bloch [Blo03], page 177, for their definitions. Accessibility is simpler to verify and in specific situations can imply controllability. This is true for "well behaved" drift free systems. Consider the expanded form of the drift-free equation (1.9):

$$\dot{q} = X_1(q)u_1 + \cdots + X_m(q)u_m \tag{1.10}$$

Let $\Delta_1 = \text{span}\{X_1, \ldots X_m\}$ be the distribution associated with the previous drift-free system. Define the *filtration* of Δ_1 to be the chain of distributions Δ_i by

$$\Delta_i = \Delta_{i-1} + [\Delta_1, \Delta_{i-1}] \tag{1.11}$$

with

$$[\Delta_i, \Delta_{i-1}] = \text{span}\left\{[X_j, X_k] : X_j \in \Delta_i, X_k \in \Delta_{i-1}\right\} \tag{1.12}$$

and $[X_j, X_k] = DX_jX_k - DX_kX_j$ the function Lie bracket. A filtration is called *regular* if in a neighborhood U of q_0 each of its distributions have constant rank in U. A control system is called regular if its associated filtration is regular. Since for all i, $\Delta_i \subset T\mathbb{R}^n$ then there exists an integer $p < n$ such that $\Delta_i = \Delta_{p+1}$ for all $i \leq p + 1$. The smallest such p is called the *degree of nonholonomy* of the system. We obtain the important result:

Theorem 1.1 (Chow's theorem for regular systems)**.** *Let system (1.10) be regular. Then, there exist admissible controls u to steer system (1.10) between two given arbitrary points $q_0, q_1 \in U$ iff for some p*

$$\Delta_p(q) = T_q\mathbb{R}^n \simeq \mathbb{R}^n, \forall q \in U \tag{1.13}$$

This theorem is sometimes called the *Lie algebra rank condition* (LARC) or *accessibility* *rank condition.* It states that controllability of system (1.9) is achieved if

$$\dim\Delta_p(q) = n \tag{1.14}$$

The distribution $\Delta_p(q)$ can be constructed using the *Phillip Hall basis* (for more information see [Aus67, Lau93]). The class of systems analyzed in this thesis are assumed to be regular, equipped with real analytic vector fields, allowing the direct application of Theorem 1.1.

1.2.2 Classical nonholonomic control results for drift-free systems

Laumond's book [Lau98] includes a complete bibliographic reference of the accomplishments in applied nonholonomic control with emphasis to cars with multiple trailers and chain systems. The paper-collection book by Li and Canny [LC93] gives a good account of the state of the art in the early nineties. We revise here the classical work of Lafferriere and Sussemann [LS93] on nilpotent systems and Murray and Sastry [MS93] on steering with sinusoids. Both approaches take advantage of particular structural elements of the Lie bracket of the system's vector field algebra.

Let q be a point on a manifold and suppose X is a vector field well defined in a neighborhood of q. Then there exists a unique[1] curve γ such that $\gamma(0) = q$ and $\dot{\gamma}(\tau) = X(\gamma(\tau))$. Define the *exponential* of αX, with $\alpha \in \mathbb{R}$, to be the point $\gamma(\alpha)$ denoted by $e^{\alpha X}q$. Then $e^{\alpha X}$ is an operation on the manifold that results in following the vector field X for an α amount of time. Given two vector fields X and Y, applying e^{X+Y} is the equivalent of following the vector field $X+Y$ for a unit time. If the vector fields are not involutive, i.e. $[X,Y] \neq 0$, then following $X+Y$ is not equivalent to following first X and then Y. The Baker-Campbell-Hausdorff formula captures this property and relates the two representations:

$$e^{\alpha X}e^{\alpha Y} = e^{\alpha X + \alpha Y + \frac{1}{2}\alpha^2[X,Y] + O(\alpha^3)} \tag{1.15}$$

The higher order terms are functions of the higher order brackets of the vector fields. Using the log function for simplicity one can re-write equation (1.15) as:

$$\begin{aligned}\log(e^{\alpha X}e^{\alpha Y}) &= \alpha X + \alpha Y + \frac{1}{2}\alpha^2[X,Y] + \frac{1}{12}\alpha^3[X,[X,Y]] - \frac{1}{12}\alpha^3[Y,[X,Y]] \\ &= -\frac{1}{48}\alpha^4[Y,[X,[X,Y]]] - \frac{1}{48}\alpha^4[X,[Y,[X,Y]]] + \ldots \end{aligned} \tag{1.16}$$

A system is called *nilpotent* of order p if any $p+1$ composition of brackets of the system's vector fields X_i vanishes, i.e. if $[X_{\mu(1)}, [X_{\mu(2)}, \ldots [X_{\mu(p)}, X_{\mu(p+1)}], \ldots]$ vanishes for any integer function $\mu(i)$. Lafferriere and Sussemann take advantage of the Baker-Campbell-Hausdorff formula and nilpotent systems to find sequences of motion $e^{\alpha_1 X_{\mu(1)}}e^{\alpha_2 X_{\mu(2)}} \cdots e^{\alpha_k X_{\mu(k)}}$ (piecewise continuous control) that can reach any point exactly. This is due to the fact that the

[1] See Curtis [Cur70] for proof.

higher order brackets of the system vanishes and equation (1.15) becomes exact. Although most systems are not nilpotent, Lafferriere and Sussemann present a mechanism for transforming particular systems into a nilpotent form by feedback.

Murray and Sastry [MS93] start from an optimal control perspective to find sinusoidal inputs to steer the chain form system (a class of nonholonomic systems that we describe in Appendix B.1.2). The idea for their algorithm is the following:

1. Steer the directly actuated variables to their desired values ignoring the evolution of the unactuated variables.

2. Using sinusoids at integrally related frequencies find an input that steers the unactuated variables to their desired values. By choice of the input the directly actuated variables are unchanged.

Many extensions to this idea followed, notably Sørdalen [rda93] shows the conversion of the kinematics of a car with n trailers into a chained form, and Tilbury et al. [TMS95] steers all the coordinates simultaneously.

1.2.3 Contemporary results using geometric control

The idea of looping in the base variables to achieve motion on the unactuated state variables was carefully explored formally by Bloch [Blo03] and Marsden [MR99]. Later, Ostrowski [Ost96], Shammas [Sha06] and others, applied this theory to a class of rolling robots. In this body of work one partitions the n-dimensional configuration manifold $\mathcal{Q}$ into two parts: an l-dimensional Lie group G and an $n-l$ dimensional base space $\mathcal{M}$, arising as a quotient space in the sense that $\mathcal{M} = \mathcal{Q}/G$. This formulation works particularly well for robot control since they generally either live in the plane $\mathsf{SE}(2)$ or in 3D space $\mathsf{SE}(3)$, both of which clearly have a Lie group structure. The base space $\mathcal{M}$ will normally contain the actuated degrees of freedom of the robot. Looping in the base space will result in motion in the group, i.e. motion in the physical world. This partition generates a *principal fiber bundle* structure, denoted $\mathcal{Q}(\mathcal{M}, G)$. If the configuration manifold can be written as the Cartesian product of the group with the base spaces, $\mathcal{Q} = G \times \mathcal{M}$, then the manifold $\mathcal{Q}$ is called *trivial principal fiber bundle*. The previous decomposition allows moving from a $2n$-dimensional systems equations in the Lagrangian form, to a $n + p < 2n$ reduced form. The dynamics described by the latter system of equations have the form of a *reconstruction equation* for the group element $g \in G$, an equation for the *nonholonomic momentum* p, and the equations of motion for the *reduced variables* r defined in the base space $\mathcal{M}$. In terms of these variables, the equations of motion have the functional form:

$$g^{-1}\dot{g} = -A(r)\dot{r} + B(r)p \tag{1.17}$$

$$\dot{p} = \dot{r}^T\alpha(r)\dot{r} + \dot{r}^T\beta(r)p + p^T\gamma(r)p \tag{1.18}$$

$$M(r)\ddot{r} = -C(r,\dot{r}) + N(r,\dot{r},p) + \tau \tag{1.19}$$

Please see [BKMM96] for the derivation of the previous equations[2]. The important aspect of this set of equations (besides being reduced order) is the decoupling of the group variables g from the base variables r in equation (1.17), normally called the action of the group on the shape. Equations (1.18),(1.19) encapsulate solely the virtual moment p and the shape variables. Ostrowski [Ost96], and later Shammas [Sha06] follow the classification from Bloch et al. [BR92] for different types of nonholonomic systems that simplify equations (1.17)–(1.19). We summarize them in Table 1.1.

Table 1.1: Classification of nonholonomic systems.

Mixed systems	equations (1.17)–(1.19)
Principally kinematic	$g^{-1}\dot{g} = -A(r)\dot{r}$ (1.20) $\dot{r} = u$ (1.21)
Purely dynamic	$g^{-1}\dot{g} = B(r)p$ (1.22) plus equations (1.18),(1.19)

When a principal fiber bundle structure is available and the Kinetic energy takes its typical quadratic form, the mixed system equations (1.17)–(1.19) are equivalent to the mechanical systems equations (1.6),(1.7). Moreover, the principally kinematic equations (1.20),(1.21) are equivalent to the 1st order affine equations (1.8).

Recently, Morin and Samson [MS03] presented an interesting feedback control solution for principally kinematic systems by introducing the notion of transverse functions f. These functions exhibit the property that their Jacobian is always orthogonal to the group distribution. By creating a new set of variables θ that live in a $n-m$–torus (variables that will induce loops in the base space), together with the special crafted transverse functions result in a full rank Jacobian H that allows for inverse dynamics. This way feedback control can be accomplished for an intermediate system $\dot{z} = Z(z)$ that results in practical stability for the group variables, i.e. stability up to a arbitrary small neighborhood ϵ of the goal. Let $gh : G \times G \to G$ be the group product and $L_g h : h \mapsto gh$, $R_g h : h \mapsto hg$ be the left and right translations respectively. Using Morin's notation, let $z, g \in G$, $\epsilon > 0$, let X_i be left-invariant vector fields, and define the intermediate variable to be $z = g f_\epsilon^{-1}$ with

$$\dot{g} = \sum_{i=1}^{m} X_i(g) u_i + P(g,t), \tag{1.23}$$

[2] The term $g^{-1}\dot{g}$ is written $T_g L_{g^{-1}} \dot{g}$ where $T_h L_g$ is the tangent space of the left translation [EW06].

an nonholonomic affine system with drift term $P(g,t)$ (equivalent to a principally kinematic system). They prove the existence of a map $f_\epsilon : U \times \mathbb{T}^{n-m} \to G$, where $U \subset G$ is a neighborhood of the identity e in G parameterized by ϵ, such that:

$$G_{f_\epsilon(\theta)} = span\{X_1(f_\epsilon(\theta)), \dots, X_m(f_\epsilon(\theta))\} \oplus df_\epsilon(\theta) \tag{1.24}$$

i.e. The Jacobian matrix $H(\theta) = \left[\begin{array}{cccccc} X_1(f_\epsilon(\theta)) & \dots & X_m(f_\epsilon(\theta)) & \frac{\partial f_\epsilon}{\partial \theta_1}(\theta) & \dots & \frac{\partial f_\epsilon}{\partial \theta_{n-m}}(\theta) \end{array}\right]$ is full rank. Using this result they find a controller for system (1.23) by following the steps:

$$\begin{aligned}
g = zf_\epsilon \quad \Leftrightarrow \quad & \dot{g} = T_z R_{f_\epsilon} \dot{z} + T_{f_\epsilon} L_z \dot{f}_\epsilon \\
\Leftrightarrow \quad & T_z R_{f_\epsilon} \dot{z} = \sum_{i=1}^{m} X_i(g) u_i + P(g,t) - T_{f_\epsilon} L_z \sum_{j=1}^{n-m} \dot{\theta}_j \frac{\partial f_\epsilon}{\partial \theta_j}(\theta) \\
\Leftrightarrow \quad & \dot{z} = (T_z R_{f_\epsilon})^{-1} \left[\underbrace{T_{f_\epsilon} L_z \sum_{i=1}^{m} X_i(f_\epsilon) u_i}_{X_i \text{ is left-invariant}} + P(g,t) - T_{f_\epsilon} L_z \sum_{j=1}^{n-m} \dot{\theta}_j \frac{\partial f_\epsilon}{\partial \theta_j}(\theta) \right] \\
\Leftrightarrow \quad & \dot{z} = T_g R_{f_\epsilon^{-1}} T_{f_\epsilon} L_z \underbrace{\left[\sum_{i=1}^{m} X_i(f_\epsilon) u_i + T_g L_{z^{-1}} P(g,t) - \sum_{j=1}^{n-m} \dot{\theta}_j \frac{\partial f_\epsilon}{\partial \theta_j}(\theta) \right]}_{\text{Since } H \text{ is invertible then replace by } T_g L_{z^{-1}} T_z R_{f_\epsilon} Z(z)} \\
\Leftrightarrow \quad & \dot{z} = Z(z)
\end{aligned}$$

The resulting system input is

$$\begin{bmatrix} u \\ -\dot{\theta} \end{bmatrix} = H(\theta)^{-1} T_g L_{z^{-1}} \left(T_z R_{f_\epsilon} Z(z) - P(g,t) \right). \tag{1.25}$$

Note that since the function $Z(z)$ is arbitrary it can possibly be chosen to encode obstacles up to a neighborhood ϵ, however, at the time of writing of this thesis, this option has not been not explored by the authors. I hope to address this structure in future work due to its promising properties.

1.2.4 Examples: the Heisenberg system and the Unicycle

The Heisenberg system or *nonholonomic integrator* [Bro82,Bro83], originated from quantum mechanics algebra, is a canonical example that has played an important role in the development of nonholonomic mechanics and control. Let $q = (x_1, x_2, x_3)$. For the velocity constraint equation

$$\dot{x}_3 = x_2 \dot{x}_1 - x_1 \dot{x}_2 \tag{1.26}$$

one can find the maximal set of independent vectors that verify (1.26), resulting in the system

$$\dot{q} = B(q)u = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -x_2 & x_1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \tag{1.27}$$

or in full form:

$$\begin{aligned} \dot{x}_1 &= u_1 \\ \dot{x}_2 &= u_2 \\ \dot{x}_3 &= x_1 u_2 - x_2 u_1 \end{aligned} \tag{1.28}$$

Bloch [Blo03] provides an example of optimal control on this particular system. The Heisenberg system can be generalized to $\mathsf{SO}(n+1)$, with $x, u \in \mathbb{R}^n$ and $Y \in \mathsf{SO}(n)$ by the following equations:

$$\dot{x} = u \tag{1.29}$$

$$\dot{Y} = [u, x] \tag{1.30}$$

If we represent Y as a matrix group element, then equation (1.30) can be written as matrix commutator:

$$\dot{Y} = xu^T - ux^T \tag{1.31}$$

In [BD97, BDK00], Bloch et al. present a stabilization solution for the generalized Heisenberg system.

The planar unicycle, illustrated in Figure 1.1, is by far the most referenced and implemented kinematic nonholonomic system. Many specific controllers have been developed for this system. Kim et al. [KT02] compare the performance of some of these algorithms via experimental validation. Recently, some interesting work has spawn in flocking and pursuit topics using unicycle-like robots, as in Vidal et al. [VSS04], Marshall et al. [MBF04], and Moshtagh et al. [MJ07]. The unicycle is defined by the following velocity constraint equation, with $q = (x, y, \theta)$:

$$\dot{y}\cos(\theta) - \dot{x}\sin(\theta) = 0, \tag{1.32}$$

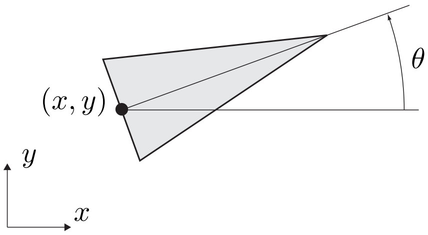


Figure 1.1: The kinematic unicycle on a plane.

resulting in the system

$$\dot{q} = B(q)u = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \tag{1.33}$$

or in full form:

$$\begin{aligned} \dot{x} &= \cos(\theta)u_1 \\ \dot{y} &= \sin(\theta)u_1 \\ \dot{\theta} &= u_2 \end{aligned} \tag{1.34}$$

We utilize the unicycle model in most of the simulation work developed for this thesis since it provides a very simple (although only loosely accurate) motion control model for RHex's robot horizontal plane behavior.

1.2.5 Classification and Complexity

The notion of algorithmic *complexity* from a nonholonomic control point of view has been addressed by Risler et al. [Ris96], Laumond [LR96], and others. In particular, Frédéric Jean [Jea01] discusses various proposals of its definition and shows that the degree of nonholonomy is an important measure for the complexity of a stabilizing algorithm. In this section, together with Appendix B, we group and revise the most commonly cited kinematic nonholonomic systems in an attempt to understand what are the important classes of systems that one should tackle. In Figure 1.2, accompanied by Table 1.2, we plot the various systems on a degree of nonholonomy versus dimension graph. By enumerating the most common systems in this fashion we observe that the "worst" complexity occurs in the diagonal (there can be no systems below the diagonal since one needs at least two vector fields to realize a Lie bracket). However, only the cars with trailers and chained systems (known to be equivalent up to a change of coordinates [rda93]) lie on the diagonal, with the exception of the recently modeled needle system [IKC$^+$06]. This table shows that most of the systems are clustered in particular degree of nonholonomy/dimension pairs, although in theory one should be able to construct specialized systems to fill in all the pairs. One can also observe that most systems have a degree of nonholonomy 1 or 2. In fact, due to algorithmic complexity, systems with a high degree of nonholonomy will most likely not be efficient, and therefore should be avoided. In general, by adding more actuators one can reduce the degree of nonholonomy and therefore its complexity.

Table 1.2: Legend for Figure 1.2. See Appendix B for a complete reference of the parameters presented in this table.

	System	Degree of nonholonomy	Dimension
H(n)	Heisenberg systems with $n \geq 2$ inputs	1	$n(n+1)/2$
U	The kinematic unicycle	1	3
R	Hopping robot in flight	1	3
Sat	Satellite with 2 control inputs [WS95]	1	3
Hi(n)	Hirose snake with n links [Hir93]	1	$2n+2$
Sh(n)	Shammas snake with n links [SCR05]	2	$n+2$
Sub(n)	Submarine with n control inputs	{1,2}	6
J(n)	Mod Jacobi n control inputs	2	$n(1+3n+2n^2)/6$
B	The ball-plate system	2	5
F	The fire truck	2	6
O(n)	One chain system of dimension $n \geq 3$	{1,2,3,...}	n
C(n)	Car with n trailers	{2,3,...}	$4+n$
N	Needle [IKC+06]	4	6

Dimension ⟶

⟵ Degree of nonholonomy	3	4	5	6	7	8

1	O(3),H(2)
	U,R,Sat
				H(3),Hi(3)
				Sub(4)
						Hi(5)

2		O(4),C(0)	J(2),Sh(3) B	Sub(3),F		
3			O(5),C(1)			
4				O(6),C(2) N		
5					O(7),C(3)	
6						O(8),C(4)

Figure 1.2: Classification of common nonholonomic systems.

We finalize this section with Figure 1.3. Here we present graphically the stratification of nonholonomic systems presented in Section 1.2. We start at the top with the general Lagrange–d'Alembert equations. Adding structure results in increasingly specialized systems for whom more control algorithms are available. The blocks on the right of Figure 1.3 represent the contemporary developments utilizing the method of reduction [Ost96]. The classes of systems addressed in this thesis are encapsulated in the boxes with the thicker solid borders.

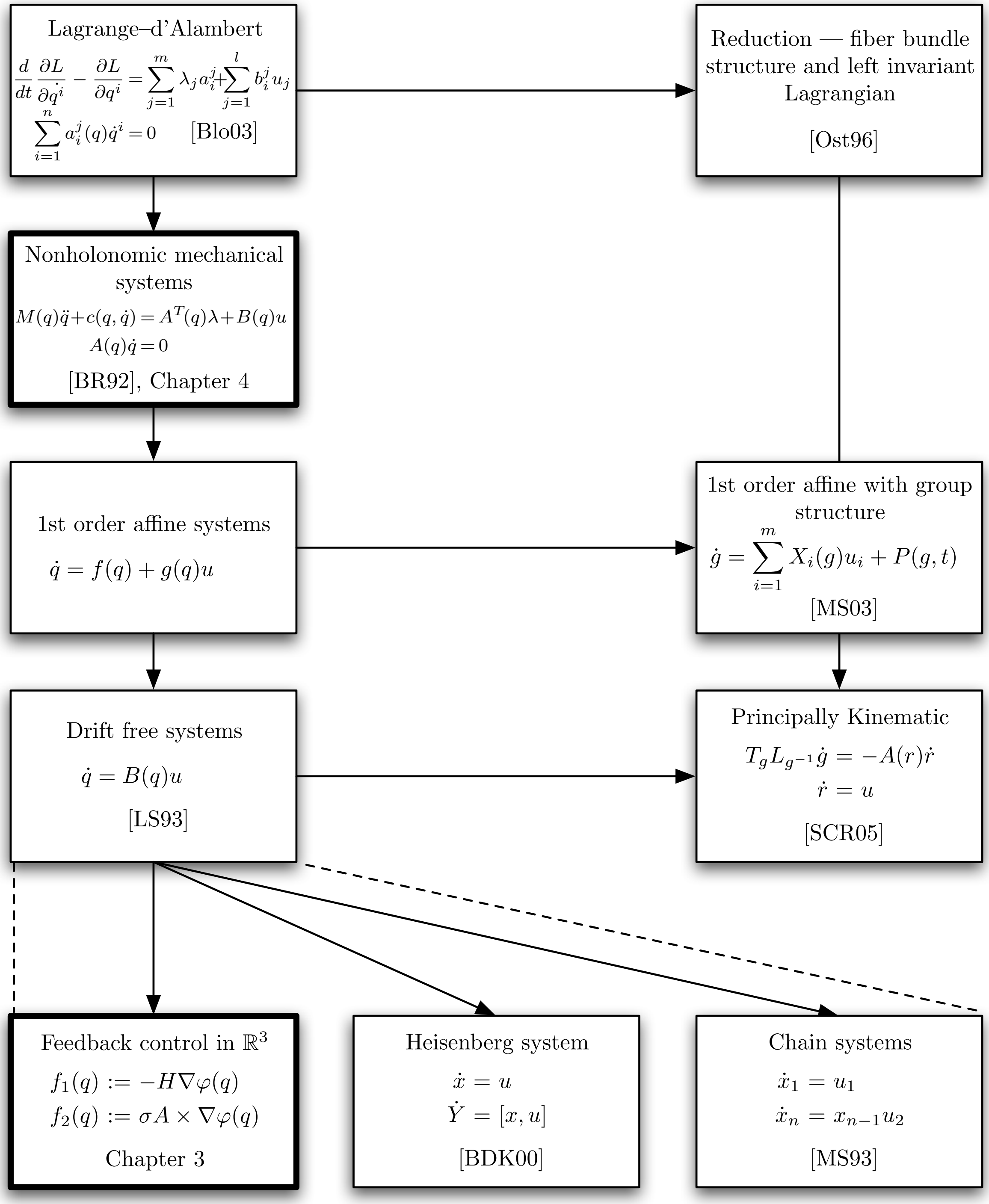

Lagrange–d'Alambert
$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}^i} - \frac{\partial L}{\partial q^i} = \sum_{j=1}^{m} \lambda_j a_i^j + \sum_{j=1}^{l} b_i^j u_j$
$\sum_{i=1}^{n} a_i^j(q)\dot{q}^i = 0$
[Blo03]
Reduction — fiber bundle structure and left invariant Lagrangian
[Ost96]
Nonholonomic mechanical systems
$M(q)\ddot{q} + c(q,\dot{q}) = A^T(q)\lambda + B(q)u$
$A(q)\dot{q} = 0$
[BR92], Chapter 4
1st order affine systems
$\dot{q} = f(q) + g(q)u$
1st order affine with group structure
$\dot{g} = \sum_{i=1}^{m} X_i(g)u_i + P(g,t)$
[MS03]
Drift free systems
$\dot{q} = B(q)u$
[LS93]
Principally Kinematic
$T_g L_{g^{-1}}\dot{g} = -A(r)\dot{r}$
$\dot{r} = u$
[SCR05]
Feedback control in $\mathbb{R}^3$
$f_1(q) := -H\nabla\varphi(q)$
$f_2(q) := \sigma A \times \nabla\varphi(q)$
Chapter 3
Heisenberg system
$\dot{x} = u$
$\dot{Y} = [x,u]$
[BDK00]
Chain systems
$\dot{x}_1 = u_1$
$\dot{x}_n = x_{n-1}u_2$
[MS93]

Figure 1.3: Stratification of nonholonomic system equations with relevant references.

1.3 Visual Servoing

The notion of visual servoing, controlling the motion of a robot by using computer vision data, has been studied intensively in recent years due to the advances in computing platforms.

Some of the configurations used in visual servoing include a camera attached to a robot, an exterior fixed camera looking at the robot's motion, or for example, a mechanical arm with a camera attached at its extremity (normally called eye-in-hand). In this thesis we focus on the first: the motion of the robot inducing motion on the camera.

The classical approach to visual servoing, described in the tutorial papers by Chaumette and Hutchinson [CH06, CH07], encodes the previous configurations by minimizing an error e associated with a vector of visual features $s \in \mathcal{B} \subset \mathbb{R}^k$ in relation to a set of fixed features s^* in the following manner:

$$e(t) = s(m(t), a) - s^*, \tag{1.35}$$

where $m(t)$ is a vector of measurements and a encodes additional known parameters of the system. There exists two main techniques to solve this problem: *image-based visual servoing* (IBVS) and *position-based visual servoing* (PBVS). One identifies the difference between these two approaches by either building the controller in the image feature space or in the robot's configuration space. The advantages and disadvantages of both approaches are described in detail in [CH06]. In practice the feature set s and its goal location s^* will either be defined by features directly extracted from the image or they will live in the 3D world after a pose computation step. This thesis takes an image-based visual servoing approach by extracting centroids of colorful objects in the image plane directly. The full image processing pipeline developed for this thesis is presented in detail in Chapter 6.

1.3.1 Image-based visual servoing

Suppose one has a fully actuated robot with a camera attached to it, and a set of well defined features or landmarks scattered throughout the surroundings of the robot. Typically, the set of image features s relates to the camera pose q via a nonlinear transformation c, denoted the *camera map*.

$$s = c(q). \tag{1.36}$$

The spacial velocity of the camera $v_c = \dot{q}$ (normally $v_c \in \mathbb{R}^n$ is defined in $\mathsf{se}(2)$ or $\mathsf{se}(3)$, i.e., $n = \{3, 6\}$) will induce a velocity in the image features $\dot{s}$:

$$\dot{s} = L_s v_c. \tag{1.37}$$

The matrix $L_s = Dc \circ c^{-1}(s) \in \mathbb{R}^{k\times n}$, denoted *iteraction matrix* or *feature Jacobian*. Assuming that the velocities v_c are the inputs for the robot, i.e. it is fully actuated, then a simple exponential stabilizer can be found by letting

$$v_c = -\lambda L_s^{\dagger} e, \tag{1.38}$$

where $L_s^{\dagger} = L_s^T(L_s L_s^T)^{-1}$ is the pseudo-inverse of L_s. The variation of the error e is

$$\dot{e} = L_s v_c = -\lambda L_s L_s^{\dagger} e \leq 0 \tag{1.39}$$

In practice since L_s and $L_s^{\dagger}$ are not exactly known, approximations are built for both matrices. A large body of work is dedicated to model and approximate the iteraction matrix for different types of image features as in the references within [CH06, CH07].

Due to the structure of IBVS, controllers built in the image space may result in non-optimal trajectories in the configuration space. This is due to the fact that vision based controllers are normally built to "minimize" the trajectories of the features in the image plane of the camera, "ignoring" the configuration space of the robot. This problem does not occur in the PBVS paradigm.

1.3.2 Position-based visual servoing

In this paradigm the camera features are used to compute a vector s in $\mathsf{SE}(2)$ or $\mathsf{SE}(3)$ that in general, represent the pose of the features in the local body coordinates. Hence, the camera map c is a transformation in $\mathsf{SE}(n)$ resulting in a square iteraction matrix. Using the same linear controller as before, we obtain the following equation for the error variation:

$$\dot{e} = L_s v_c = -\lambda L_s L_s^{-1} e = -\lambda e \leq 0 \tag{1.40}$$

For this configuration one obtains exponentially decreasing speeds and most likely geodesic trajectories in the configuration space, depending on the parameterization of s. This results in good trajectories in the configuration space and less good trajectories of the features in the camera image plane.

1.3.3 Visual servoing via exact changes of coordinates

When a known diffeomorphism[3] c exists between the feature space $\mathcal{B}$ and the robot's configuration space $\mathcal{Q}$, the visual servoing problem becomes considerably simple. Let $c : \mathcal{Q} \to \mathcal{B}$ be C^1 such that $s = c(q)$, with $q \in \mathcal{Q}$ and $s \in \mathcal{B}$. Choose $\varphi : \mathcal{B} \to \mathbb{R}^+$ to be a positive definite function in $\mathcal{B}$ for s^*, such that $\varphi(s^*) = 0$ and $\varphi(s) > 0, \forall s \neq s^*$. Using the

[3] For definition see Section 2.1

negative gradient of $\varphi \circ c(q)$ as a velocity input results in the error variation equation:

$$\begin{aligned}\dot{e} &= -L_s \, \nabla_q \left(\varphi \circ c(q)\right)\big|_{q=c^{-1}(s)} = \\ &= -Dc \circ c^{-1}(s) Dc^T \circ c^{-1}(s) \nabla_s \varphi(s) \\ &= -K \nabla \varphi(s) \leq 0 \end{aligned} \tag{1.41}$$

where $K = DcDc^T$ is positive definite since c is a proper diffeomorphism. If a PBVS algorithm is being utilized, then the camera map c will be a rotation R plus translation t. The iteraction matrix is $L_s = R$ and equation (1.41) becomes

$$\dot{e} = -RR^T \nabla \varphi(s) = -\nabla \varphi(s) \leq 0 \tag{1.42}$$

1.4 Navigation with perceptual limitations and motion constraints

As described before, both the IBVS and PBVS control schemes have advantages and disadvantages. While the IBVS can be good for maintaining the integrity of the features on the image plane, it may result in non-optimal trajectories of the robot. The PBVS on the other hand, can result in poor feature trajectories in the image plane and good trajectories of the robot. These shortcomings become even more relevant in the presence of obstacles in the image plane. These obstacles normally refer to feature occlusions by leaving the camera field of view or by object self-occlusion. From this perspective, the IBVS seems more appropriate since one can model the iteraction matrix and the camera velocity controllers to accommodate for such obstacles directly. However, in the presence of motion constraints, such as nonholonomic, it is expected that the PBVS scheme be better suited since its controllers are built in the same manifold as the robot's configuration manifold. In this thesis we take an IBVS approach by developing a class of diffeomorphic camera maps and building gradient controllers directly in the image feature space.

Few researchers have addressed the dual problem of motion constraints and perceptual limitations. Among these we find the work of Kantor [KR03], Murrieri [MFB04], Bhattacharya [BMCH04] and Folio [FC05]. However, these authors assume a particular set of constraints for which a feedback control law is subsequently developed to take into account the special form. Murrieri et al. develops a collection of specialized Lyapunov based controllers for a wheeled vehicle, with perception limited by the monocular camera's field of view. Folio et al. proposes switching between three controllers that each deal with the visual servoing task at hand, beacon occlusion or obstacle avoidance. However this task is facilitated by a pan camera mechanism on a cart-like car, resulting in the problem of stabilization in $\mathsf{SE}(2)$ with 3 available inputs. Kantor et al. combine Ikeda's variable constraint control [INMA99] with the notion of sequential composition of controllers [BRK99] to drive the robot RHex to a specified goal location. This approach results in efficient trajectories but can be difficult to reuse on systems with different motion models and/or different perceptual constraints. Bhattacharya et al. take a geometric approach and find minimum length paths, but again, implemented for a specific configuration space.

1.5 Contributions of the thesis

In this thesis, we take a few steps toward a more general approach to perception-based servoing that decouples the (typically holonomic) perceptual constraints from the (typically nonholonomic) kinematic constraints by adapting an "arbitrary" *navigation function* [Kod92] to an "arbitrary" nonholonomically constrained first order mechanism operating in the configuration space comprising the navigation function's domain. The encoding of holonomic constraints via navigation functions is a very effective means of constructing "designer" basins around specified goal points for fully actuated first and second order mechanisms. For example, in visual servoing applications, the navigation function takes into account external constraints like limited field of view, obstacles, and so on. We are most immediately motivated by the prospect of extending Cowan's [CWK02] work on navigation with triple-beacon landmarks to the robot RHex, but we introduce a more general framework for nonholonomically constrained visual servoing via PD control. We offer two other examples of navigation functions arising from perceptual apparatus that provide some feeling for the virtue of the more general view.

This thesis makes four specific contributions. First, we adapt Cowan's construction of a navigation function for moving landmarks viewed by a stationary monocular camera to the "inside out" case of a moving monocular camera viewing a fixed landmark. This entails generalizing the "camera map" for convex landmarks to the more general setting relevant to outdoor mobile robotics with landmarks formed by any triplet of beacons in general position (i.e., whose convex hull encloses a set with non-empty interior on the plane).

Second, we construct a hybrid controller for arbitrary navigation functions applied to arbitrary drift-free three dimensional control systems with two independent control inputs. The resulting switching feedback law guarantees "practical stability" (in the sense of Morin and Samson [MS03]: convergence to an arbitrarily small specified neighborhood of the goal) with the added guarantee that no obstacle will ever be encountered along the way. We offer very general (and easily verified) sufficient conditions under which the basin (the set of initial configurations brought into the goal's small neighborhood) includes a far larger "local surround" bounded by the "highest" level set of the navigation function that is still a topological sphere. We present additional "global" conditions (albeit much more narrowly adapted to the specific examples at hand) sufficient to guarantee that the basin includes all initial conditions except possibly a set of measure zero. Note that in general, there is no guarantee that these conditions will prevail, and we can make no general statements about the global versions of this problem. However, one of the central motivations for embracing navigation functions and their associated controllers is the possibility of developing further abstracted compositions using suitably arranged "deployments" of their computable basins, such as the "back-chaining" sequential composition in [BRK99], that we present in Section 2.3 and implement in Section 6.4.

Third, we implement an instance of this visual servoing framework on the robot RHex [SBK01] in general outdoor terrain viewing through a monocular camera landmarks comprised of three beacons in arbitrary general position. We provide extensive experimental data to document the robustness of the algorithm against the inaccuracies in the (very crude) control model and the many practical sources of noise in the sensor suite's acquisition of the naturally illuminated outdoor scene.

Fourth, we study the possibility of adapting the first order hybrid feedback controller for nonholonomic constrained systems, developed in Chapter 3, to their dynamical counterpart. The simulations presented support the proposed lift.

The content of this thesis has been published in the following articles: Chapter 3 in [LK04,LK05,LK07], Chapter 4 in [LK06], Chapter 5 in [CLK00,LK03] and part of Chapter 6 in [WLB$^+$04].

1.6 Organization of the thesis

A summary of the mathematical background recommended for reading this thesis is introduced in Chapter 2. We present some notions of differential geometry in Section 2.1, including the definitions of differential manifold, diffeomorphisms, foliations, and Morse functions. We also revise the center manifold theorem for flows and the singular perturbation theorem. In Section 2.2, we establish the main tool utilized for point stabilization in this thesis: the definition of a Navigation function. We finish in Section 2.3 by revising the notion of Sequential Composition, utilized in the real world implementation described in the experimental sections of Chapter 6.

The main contributions of this thesis appear in Chapters 3 and 4. Starting in Chapter 3, we introduce a hybrid controller for stabilization of drift free nonholonomically constrained systems in $\mathbb{R}^3$, defined in punctured configuration spaces. Starting in Section 3.1.1 with a number of preliminary technical developments, we subsequently present our general mathematical result in Section 3.1.2. Namely, we lay out the hybrid algorithm, state the sufficient conditions for convergence from the "local surround", and provide more specific sufficient conditions for essential global convergence at the price of less general assumptions appropriate to the particular case — necessarily so, because a "general extension" would include a constructive solution of the global navigation problem which we are far from claiming to encompass within the scope of this thesis. Simulations for various configuration spaces are presented in Section 3.5.

In Chapter 4 we review some mechanical system equation properties. Next we adapt the embeding of gradient dynamics to nonholonomically constrained mechanical systems in Section 4.1, and adapt the embbeding of generic reference dynamics in Section 4.2. Simulations and discussion are presented in Section 4.3.

We present the ingredients of the specific RHex servoing problem in Chapter 5. In Section 5.1 we introduce the camera map arising from a monocular camera's view of a three-beacon landmark, and in Section 5.2 the navigation function associated with it.

Finally, in Chapter 6 we apply the foregoing constructions to the central motivation for the experimental work developed in this thesis: the specific case of a visual servo algorithm for the robot RHex. We start by describing the perceptual software and hardware in Section 6.1 along with the specific controller implementation in Section 6.2. We present statistics documenting the successful indoor and outdoor implementations in Sections 6.3 and 6.4.

We close with the concluding remarks and future work in Chapter 7.

CHAPTER 2

Mathematical Background

In this chapter we present most of the necessary tools and constructions required for the understanding of this thesis. The following definitions, borrowed from differential topology and differential geometry fields, can also be found in most nonlinear control books such as [GH83, Arn89, Kha96, Son98, Sas99, MR99, Blo03, BL05].

2.1 Differential equations on manifolds

Throughout this thesis we call $q \in M \subset \mathbb{R}^n$ the state variable that lives in some set M. Let (2.1) be a general differential equation with f globally Lipchitz:

$$\dot{q} = f(q) \tag{2.1}$$

Definition 2.1. A space M is an *n-manifold* if each $q \in M$ lies in some open set homeomorphic to some ball $\mathrm{B}_r(o) \subset \mathbb{R}^n$. An n-manifold is said to have dimension n.

Let M be a n-manifold. By definition, for any $q \in M$ we have U open in $\mathbb{R}^n$ and a homeomorphism $\phi : U \to M$ into M with $\phi(U)$ being an open neighborhood of q. We call such a pair (U, ϕ) a *chart*. A Collection of charts that cover M is called an *atlas*. Two charts $\phi : U \to M$ and $\psi : V \to M$ are said to *overlap smoothly* if either $\phi(U) \cap \psi(V) = \emptyset$, or $\psi^{-1} \circ \phi$ is a diffeomorphism.

Definition 2.2. M is a *differentiable manifold* if it has an atlas of smoothly overlapping charts.

Definition 2.3. A *diffeomorphism* ϕ is a map between manifolds which is differentiable and has a differentiable inverse. Two manifolds M and N are *diffeomorphic*, written $M \simeq N$, if there is a diffeomorphism ϕ from M to N.

Given the previous notions, an important theorem that captures the invariant manifold structure of a vector field f with a local kernel, that also applies directly to constrained drift-free nonholonomically constrained systems that which we are mostly interested, is presented by Guckenheimer et al. [GH83] and is stated as follows:

Theorem 2.4 (Center manifold theorem for flows)**.** *Let f be a C^r vector field on $\mathbb{R}^n$ vanishing at the origin $f(0) = 0$ and let $A = Df(0)$. Divide the spectrum of A into three parts, $\sigma_s, \sigma_c, \sigma_u$ with*

$$\mathrm{Re}\lambda \begin{cases} < 0 & \textit{if} \quad \lambda \in \sigma_s, \\ = 0 & \textit{if} \quad \lambda \in \sigma_c, \\ > 0 & \textit{if} \quad \lambda \in \sigma_u. \end{cases} \tag{2.2}$$

Let the (generalized) eigenspaces of σ_s, σ_c and σ_u be E^s, E^c and E^u, respectively. Then there exist C^r stable and unstable invariant manifolds $\mathcal{W}^s$ and $\mathcal{W}^u$ tangent to E^s and E^u at 0 and a C^{r-1} center manifold $\mathcal{W}^c$ tangent to E^c at 0. The manifolds $\mathcal{W}^u, \mathcal{W}^s$ and $\mathcal{W}^c$ are all invariant for the flow of f. The stable and unstable manifolds are unique, but $\mathcal{W}^c$ need not be.

The consequences of this theorem have the largest impact on the work presented in this thesis, since for drift-free nonholonomically constrained systems with any time-invarient feedback control law (or any other constrained kinematic system) it will always exhibit a Center / Stable manifold structure (assuming that the feedback controller is not unstable). This theorem also indicates how to compute 1st order approximations for such manifolds.

Definition 2.5. [Rol76] Let M be a n-manifold and let $F = \{F_\alpha\}$ denote a partition of M into disjoint pathwise-connected subsets. Then F is called a *foliation* of M of codimension m (with $0 < m < n$) if there exists a cover of M by open sets U, each equipped with a homeomorphism $\phi : U \to \mathbb{R}^n$ or $\phi : U \to \mathbb{R}^n_+$ which throws each nonempty component of $F_\alpha \cap U$ onto a parallel translation of the standard hyperplane $\mathbb{R}^{n-m}$ in $\mathbb{R}^n$. Each F_α is then called a *foliation leaf* and is not necessarily closed or compact.

Foliation structures appear in Chapter 3 as a result of attempting to use navigation functions directly on systems with constraints.

Now consider the system with perturbation α and $q = (q_1, q_2) \in \mathbb{R}^n$:

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} := h_\alpha(q) = \begin{bmatrix} h_1(q) \\ \alpha h_2(q) \end{bmatrix} \tag{2.3}$$

Theorem 2.6 (Singular perturbation theorem [Fen79])**.** *Consider the system (2.3) with $0 \leq \alpha \ll 1$. Suppose that for $\alpha = 0$, (2.3) admits an equilibrium manifold of dimension m, $0 < m < n$, denoted by $\mathcal{W}_h^0$ and for all $q^* \in \mathcal{W}_h^0$, the Jacobian matrix, $D_q h_\alpha|_{(q^*,0)}$ admits $n - m$ eigenvalues with a strictly negative real part. Then, for every open and bounded subset Ω of $\mathcal{W}_h^0$, there exists an open neighborhood U of Ω in $\mathbb{R}^n$, such that, for α positive and small enough, the perturbed system (2.3) admits an attractive invariant sub-manifold $\mathcal{W}_h^\alpha$ contained in U and close to $\mathcal{W}_h^0$.*

The previous theorem establishes that under appropriate conditions the "slow" dynamics of h_α, defined in equation (2.3), approaches the center manifold of h_0 as α goes to zero. This result is important in Chapter 3 to help understand the structure of the center manifold in the configuration space of the system. For a tutorial treatment of Singular Perturbations please see [Wig94] or [Kha96].

We end this section with the notion of a Morse function, important for the definitions to come:

Definition 2.7. Let M be a smooth manifold. A critical point of a map $\varphi : M \to \mathbb{R}$ at $q \in M$ is called *non-degenerate* if the Hessian matrix H_φ (in any local coordinate system) at q is non-degenerate.
A smooth function $\varphi : M \to \mathbb{R}$ is called *Morse* if all of its critical points are non-degenerate. Morse functions exist on any smooth manifold, and in fact form an open dense subset of smooth functions on M.

2.2 Navigation functions

The use of total energy as a Lyapunov function for mechanical systems has a long history [Kod89b] stretching back to Lord Kelvin [TT86]. The notion of total energy presupposes the presence of potential forces arising from the gradient of a scalar valued function over the configuration space. In this thesis we focus our interest on "artificial cost functions" introduced by a designer to encode some desired behavior, as originally proposed by Khatib [KM78, Kha86]. However, we take a global view of the task, presuming a designated set of prohibited configurations — the "obstacles" — and a designated set of selected configurations — the "goal," which we restrict throughout the thesis to be an isolated single point. One can achieve the global specification through the introduction of a *Navigation Function* (NF) [Kod92] — an artificial potential function that attains its maximum on the entire boundary of the obstacle set, and its only local minimum exactly on the isolated goal point:

Definition 2.8. [Kod89a] Let $\mathcal{Q} \subset \mathbb{R}^n$ be a smooth compact connected manifold with boundary, and $q^* \in \mathcal{Q}$ be a point in its interior. A C^2 Morse function, $\varphi : \mathcal{Q} \to [0, 1]$ is called a *navigation function* if

1. φ takes its unique minimum at $\varphi(q^*) = 0$;

2. φ achieves its maximum of unity uniformly on the boundary, i.e. $\partial\mathcal{Q} = \varphi^{-1}[1]$.

Such functions are guaranteed to exist over any configuration space of relevance to physical mechanical systems [KR90], and constructive examples have been developed for a variety of task domains [RK92, KB01, CWK02, KBK04]. NF-generated controls applied to completely actuated mechanical systems force convergence to the goal from almost every initial condition and guarantee that no motions will intersect the obstacle set along the way. In the dynamical setting, where the role of kinetic energy is important, they achieve a pattern of behavior analogous to that of similarly controlled corresponding quasi-static dynamics. For example, in the one degree of freedom case, the dynamical setting is represented by the familiar spring-mass-damper system

$$m\ddot{q} + c\dot{q} + kq = 0 \tag{2.4}$$

and the corresponding quasi-static model arises through a neglect of the inertial forces, $m \to 0$ in (2.4), yielding

$$c\dot{q} + kq = 0 \tag{2.5}$$

To illustrate the nature of NF-gradient-based controllers in this simple setting, take the configuration space to be $\mathcal{Q} := \{q \in \mathbb{R} : |q| \leq 1\}$ with navigation function $\varphi(q) := \frac{1}{2}kq^2$, implying that $\{0\} = \varphi^{-1}[0]$ is the goal and $\{-1, 1\} = \varphi^{-1}[1]$ the obstacle set. We imagine that both systems, (2.4), (2.5), arise from application of the NF-gradient control law, $u := -\nabla\varphi$, to the respective open loop,

$$\begin{aligned} u &= m\ddot{q} + c\dot{q} \qquad (2.6)\\ &\text{or}\\ u &= c\dot{q}. \qquad (2.7) \end{aligned}$$

We observe that φ is a global Lyapunov function for (2.5), guaranteeing that all initial conditions give rise to motions that avoid the obstacle set while converging asymptotically on the goal set. Analogously, the total energy, $\mu := \frac{1}{2}\dot{q}^2 + \varphi(q)$ is a Lyapunov function for the velocity-limited extension of $\mathcal{Q}$, $\mathcal{X} := \mu^{-1}[0, 1] = \left\{(q, \dot{q}) \in \mathbb{R}^2 : \mu(q, \dot{q}) \leq 1\right\}$. This guarantees that all initial conditions in $\mathcal{X}$ give rise to motions that avoid the obstacle (and, in fact, are repelled from the entire boundary, $\mu^{-1}[1]$) while converging asymptotically on the zero velocity goal set, $\mu^{-1}[0] = \{(0, 0)\}$. These results will be carefully studied in Chapter 4 when we present the lift of the kinematic algorithms developed in this thesis to the dynamical setting.

Navigation functions, due to their smooth structure, can be pushed through changes of coordinates, enlarging the domain of applications beyond the construction methods developed in [RK91]. Practical implementations include [CWK02] and those described in Chapter 6. Let φ be a navigation function defined in M and let N be a manifold isomorphic to M endowed with the diffeomorphic map $c : N \to M$. Let $q \in M$ and $p \in N$. One can write a navigation function for N using the composition rule:

$$\bar{\varphi}(p) := \varphi \circ c(p) \tag{2.8}$$

With velocities defined by the pullback:

$$\nabla\bar{\varphi}(p) = Dc^T(p)\nabla\varphi \circ c(p) \tag{2.9}$$

The collection of the previous results allows for a very rich set of tools that we explore in the robotics framework in this thesis.

2.3 Sequential composition

For most practical systems it is difficult to fully encode the robot and environment workspaces via a single navigation function (or any other control function). It is reasonable to partition the workspace into smaller and simpler tasks that can be studied in more detail. One such method denoted *Sequential Composition* [BRK99], naturally extends the results obtained for navigation functions to more complex topologies by defining a partition of the workspace and associating different navigation functions to each partition element. These concepts may be formalized as follows. Let Φ_i be a controller with domain of attraction $\mathcal{D}(\Phi_i)$ and goal $\mathcal{G}(\Phi_i)$. We say that controller Φ_i *prepares* controller Φ_{i+1}, denoted by $\Phi_i \succeq \Phi_{i+1}$, if the goal of the first lies in the domain of attraction of the second: $\mathcal{G}(\Phi_i) \subset \mathcal{D}(\Phi_{i+1})$. To guarantee that the robot can handle any situation, its workspace, denoted by $\mathcal{W}$, should be covered by the domains of attraction of the set of controllers: $\mathcal{W} \subset \bigcup_{\Phi_i \subset \mathcal{U}} \mathcal{D}(\Phi_i)$. Sequential composition can be readily implemented in a robotic platform via state machines by associating each domain of attraction to a state. We present one such implementation in Section 6.4.

CHAPTER 3

Kinematic machines in $\mathbb{R}^3$

In this section we introduce a hybrid controller for stabilization of drift free nonholonomically constrained systems in $\mathbb{R}^3$. We present a set of verifiable conditions that guarantee stabilization for the successive application of a two step controller: the first moves on level sets of a gradient function, escaping the system's center manifold and obstacles in the configuration space, and if possible reaching the goal's stable manifold; the second uses a gradient control law to reach the goal. The objective of this section is to develop a class of robust algorithms that are subsequently implemented on the legged robot RHex. The emphasis is on developing a feedback control algorithm that encodes perception constraints side by side with motion constraints in a seemly manner.

Consider the class of smooth and piecewise analytic, three degree of freedom, drift-free control systems

$$\dot{q} = B(q)u, \quad q \in \mathcal{Q} \subset \mathbb{R}^3; \quad u \in \mathbb{R}^2, \tag{3.1}$$

where $B \in \mathbb{R}^{3\times 2}$ and $\mathcal{Q}$ is a smooth and piecewise analytic, compact[1], connected three dimensional manifold with a boundary, $\partial\mathcal{Q}$ (that separates the acceptable from the forbidden configurations of $\mathbb{R}^3$), possessing a distinguished interior *goal point*, $q^* \in \mathcal{Q}$. In this section we will impose very general assumptions on B and construct a hybrid controller that guarantees local convergence to an arbitrarily small neighborhood of the goal state while avoiding any forbidden configurations along the way. In the next section, we will introduce more specialized assumptions that extend the basin of attraction to include almost every initial configuration in $\mathcal{Q}$.

We find it convenient to rewrite (3.1) using the *nonholonomic projection matrix* [LK04],

[1] We consider the configuration space $\mathcal{Q}$ to be a compact set since this requirement is built into the definition of a navigation function. The changes of coordinates for the camera maps are mostly defined in $\mathsf{SE}(2)$ because they are valid there. In general, as in the present application, due to the limitations of the vision sensors, the workspace is always bounded, hence its closure is compact.

H into the image of B:

$$H(q) = B(q)B(q)^\dagger = B(q)\left(B(q)^T B(q)\right)^{-1} B(q)^T \tag{3.2}$$

$$\dot{q} = H(q)v, \quad q \in \mathcal{Q} \subset \mathbb{R}^3; \quad v \in \mathbb{R}^3 \tag{3.3}$$

Throughout this section it is assumed that B has rank two at each point.

3.1 Two controllers and their associated closed loop dynamics

It is useful to compare the unconstrained system $\dot{q} = v$ with the constrained version (3.3). Let φ be a navigation function defined in $\mathcal{Q}$. For the input $v = -\nabla\varphi$ the unconstrained system is globally asymptotically stable at the origin. Using φ as a control Lyapunov function yields $\dot{\varphi} = -\|\nabla\varphi\|^2$. Given this result, a naive approach to attempt stabilizing system (3.3) is to use the same input $v = -\nabla\varphi$.

Define the vector field $f_1 : \mathcal{Q} \to T\mathcal{Q}$ such that $f_1(q) := -H(q)\nabla\varphi(q)$ and the system

$$\dot{q} = f_1(q) = -H(q)\nabla\varphi(q) \tag{3.4}$$

Since H has a 1-dimensional kernel and $D^2\varphi$ is full rank at q^* it follows that (3.4) has a 1 dimensional center manifold

$$\mathcal{W}^c := \{q \in \mathcal{Q} : H(q)\nabla\varphi(q) = 0\}, \tag{3.5}$$

as corroborated by explicitly computing[2] the Jacobian of f_1 at q^*:

$$Df_1\big|_{q^*} = -DH \underbrace{\nabla\varphi\big|_{q^*}}_{=0} - HD^2\varphi = - HD^2\varphi\big|_{q^*} \tag{3.6}$$

Using φ as a Lyapunov function, La Salle's invariance theorem states that system (3.4) has its limit set in $\mathcal{W}^c$:

$$\begin{aligned} \dot{\varphi} &= -\nabla\varphi^T H \nabla\varphi \\ &= -\|H\nabla\varphi\|^2 \begin{cases} = 0 & \text{if} \quad q \in \mathcal{W}^c \\ < 0 & \text{if} \quad q \notin \mathcal{W}^c \end{cases} \end{aligned} \tag{3.7}$$

Figure 3.1 illustrates the topology associated with (3.4): the projection H imposes a co-dimension 1 foliation complementary to the center manifold. The *stable manifold,* $\mathcal{W}^s$, is

[2] note the abuse of notation in equation (3.6): DH is actually a tensor.

the leaf containing the goal, q^*. The input

$$u_1 := -B(q)^\dagger \nabla\varphi(q) \tag{3.8}$$

alone cannot stabilize system (3.4) at the origin, since no smooth time invariant feedback controller has a closed loop system with an asymptotically stable equilibrium point [Bro83]. Nevertheless, for any initial condition outside $\mathcal{W}^c$ an infinitesimal motion in the direction of f_1 reduces the energy φ. If there can be found a second controller that "escapes" $\mathcal{W}^c$ without increasing φ, then it is reasonable to imagine that iterating the successive application of these two controllers might well lead eventually to the goal. We now pursue this idea by introducing the following controller,

$$u_2 := B(q)^\dagger \left[A(q) \times \nabla\varphi(q)\right], \tag{3.9}$$

leading to the closed loop vector field[3]

$$\begin{aligned} \dot{q} &= H(q)f_2(q) = f_2(q) \\ f_2(q) &:= A(q) \times \nabla\varphi(q), \end{aligned} \tag{3.10}$$

where $A(q)$ can be computed by the normalized cross product of the columns of $B := [B_1\ B_2]$:

$$A(q) := \frac{B_1 \times B_2}{\|B_1 \times B_2\|} \tag{3.11}$$

Note that the nonholonomic constraint expressed in (3.1) can be represented by the implicit equation $A^T(q)\dot{q} = 0$. Since the (Lie) derivative of φ in the direction of f_2 is

$$L_{f_2}\varphi = \nabla\varphi(q)^T(A(q) \times \nabla\varphi(q)) = 0, \tag{3.12}$$

it follows that f_2 is φ-invariant — i.e. the energy, φ, is constant along its motion. Moreover $Hf_2 = (I - AA^T)(A \times \nabla\varphi) = A \times \nabla\varphi = f_2$, verifying that f_2 indeed satisfies the constraint (3.1).

3.1.1 Assumptions, a Strategy, and Preliminary Analysis

Having introduced two vector fields — one which is energy decreasing; the other energy conserving — we now sketch a strategy that brings initial conditions of system (3.1) to within an arbitrarily small neighborhood ϵ of the goal, by way of motivating the subsequent definitions, and claims that arise in the formal proofs to follow. Let $\Phi_t^{f_1}$ and $\Phi_t^{f_2}$ denote the flows of f_1 and f_2 respectively. The point stabilization strategy is as follows:

[3] Below we show that $\forall q : Hf_2 = f_2$

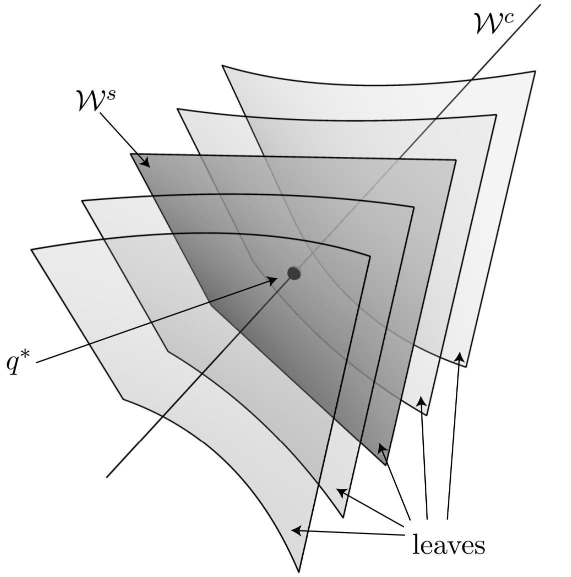


Figure 3.1: Conceptual illustration of the flow associated with equation (3.4). Each leaf is an invariant manifold with all trajectories collapsing into $\mathcal{W}^c$.

1. If $q_0 \in \mathcal{W}^c$ then follow a constant direction f_0 in image(H) for a finite amount of time t_0 such that $\Phi_{t_0}^{f_0}(q_0) \notin \mathcal{W}^c$ and $\varphi \circ \Phi_{t_0}^{f_0}(q_0) < 1$ for all $t \in (0, t_0)$.

2. If $q_0 \notin \mathcal{W}^c$ and $\varphi(q_0) > \epsilon$

   2.1) Use a scaled version of f_2 for time τ_2 to escape a δ-neighborhood of $\mathcal{W}^c$, keeping the energy φ constant.

   2.2) Use controller f_1, for time τ_1, to decrease the energy φ, stopping at a γ-neighborhood of $\mathcal{W}^c$ such that $\Phi_{\tau_1}^{f_1}(q) \notin \mathcal{W}^c$ and $\gamma < \delta$.

We now introduce a number of assumptions, definitions and their consequences that will allow us to formalize each of the previous steps:

A1. $\mathcal{Q}$ is a smooth compact connected manifold with boundary.

A2. φ is a navigation function in $\mathcal{Q}$.

A3. H has rank two, uniformly throughout $\mathcal{Q}$.

Assumption A1 gives the proper setting for the existence of a navigation function in the configuration space. Assumption A3 assures the foliation sketched in Figure 3.1.

Define the *local surround* of the goal, illustrated in Figure 3.2, to be the closed "hollow sphere", $\mathcal{Q}_s := \varphi^{-1}[I_{\epsilon s}]$, with $I_{\epsilon s} := [\epsilon, \varphi_s]$ whose missing inner "core" is the arbitrarily small open neighborhood, $\mathcal{Q}_\epsilon := \varphi^{-1}[I_{0\epsilon}]$; $I_{0\epsilon} := [0, \epsilon)$, and whose outer "shell", $\mathcal{Q}_1 := \varphi^{-1}[I_{s1}]$, with $I_{s1} := (\varphi_s, 1]$, includes the remainder of the free configuration space. φ_s is defined to be the largest level such that all the smaller levels, $\varphi_0 \in (0, \varphi_s)$, are homeomorphic to the sphere, S^2, and are all free of critical points, $\|\nabla\varphi\|^{-1}[0] \cap \varphi^{-1}[(0, \varphi_s)] = \emptyset$.

The restriction to φ-invariant topological spheres precludes limit sets of f_2 more complex than simple equilibria in the local surround. In the examples of sections 3.5.2 and 3.5.4, we provide more specialized conditions that allow us to guarantee that the algorithm brings almost every initial condition in the "outer" levels, $\mathcal{Q}_1$ into the local surround, $\mathcal{Q}_s$ and, thence, into the goal set $\mathcal{Q}_\epsilon$.

Lemma 3.1. *Given the previous assumptions*

$$f_1^{-1}[0] \cap \mathcal{Q}_s \equiv f_2^{-1}[0] \cap \mathcal{Q}_s \equiv \mathcal{W}^c \cap \mathcal{Q}_s. \tag{3.13}$$

Proof. If $q \in f_2^{-1}[0]$ then $\nabla\varphi = \alpha A$, where α is a non-zero scalar, hence $\nabla\varphi \in \ker H$ and $q \in \mathcal{W}^c$ as defined by (3.5). □

To formally express the "δ-neighborhood" described in the stabilization strategy, we start by defining the function $\xi : \mathcal{Q} - \{q^*\} \to [0, 1]$:

$$\xi(q) := \frac{\|H(q)\nabla\varphi(q)\|^2}{\|\nabla\varphi(q)\|^2} \tag{3.14}$$

The quantity $\|H(q)\nabla\varphi(q)\|^2$ evaluates to zero only in $\mathcal{W}^c - \{q^*\}$. Therefore, in a small neighborhood of $\mathcal{W}^c$ the level sets of $\|H(q)\nabla\varphi(q)\|^2$ define a "tube" around $\mathcal{W}^c$. The denominator of (3.14) normalizes ξ such that $0 \leq \xi \leq 1$. Moreover, it produces a "pinching" of the tube at the goal q^*.

Lemma 3.2. *For all $\varphi_0 \in I_{\epsilon s}$, $\varphi^{-1}[\varphi_0]$ intersects the unit level set of ξ, i.e., $\xi^{-1}[1] \cap \varphi^{-1}[\varphi_0] \neq \emptyset$.*

Proof. Observe that $\xi(q) = 1$ is equivalent to the condition $\nabla\varphi^T Q \nabla\varphi = 0$ where $Q := I - H$. Now consider the family of vector fields

$$h_\alpha(q) := -[Q(q) + \alpha H(q)]\nabla\varphi(q), \tag{3.15}$$

$$\dot{q} = h_\alpha(q), \tag{3.16}$$

Note, for $\alpha > 0$ the goal point q^* is globally asymptotically stable over the domain $\mathcal{Q}_\epsilon \cup \mathcal{Q}_s$, since φ is a Lyapunov function for (3.16),

$$\begin{aligned}\dot{\varphi} &= -\nabla\varphi^T(Q + \alpha H)\nabla\varphi = \\ &= -\nabla\varphi^T(Q(1-\alpha) + \alpha I)\nabla\varphi = \\ &= -(1-\alpha)\nabla\varphi^T Q \nabla\varphi - \alpha\|\nabla\varphi\|^2 \leq -\alpha\|\nabla\varphi\|^2\end{aligned} \tag{3.17}$$

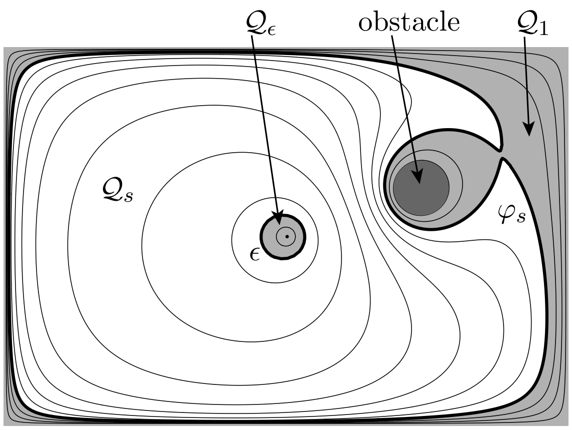


Figure 3.2: Illustration of the local surround $\mathcal{Q}_s$ of the goal in white. The thin lines represent various levels of φ. This image is presented in the plane for readability purpose. However it should be interpreted as a section of $\mathcal{Q} \subset \mathbb{R}^3$

and φ has no other critical points other than q^* in $\mathcal{Q}_\epsilon \cup \mathcal{Q}_s$.

Next, observe that $\xi^{-1}[1] = \mathcal{W}_h^0$ is a center manifold for h_0. Hence, according to Fenichel's Singular Perturbation Theorem there persists a "slow stable manifold" of h_α, $\mathcal{W}_h^\alpha$, that is arbitrarily close to $\xi^{-1}[1]$ as the positive scalar α approaches 0. Configurations $q_0 \in \varphi^{-1}[\epsilon] \cap \mathcal{W}_h^\alpha$, that are arbitrarily close to q^* on this invariant set are associated with reverse time trajectories $\Phi_{-t}^{h_\alpha}(q_0)$ that pass through every level set $\varphi^{-1}[\varphi_0]$, for $\varphi_0 \in I_{\epsilon s}$ since $\dot{\varphi} \circ \Phi_t^{h_\alpha}(q_0) < 0$ according to the previous paragraph. It follows that $\mathcal{W}_h^\alpha$ intersects every level set, $\varphi^{-1}[\varphi_0]$, for $\varphi_0 \in I_{\epsilon s}$ and $\alpha = 0$ as well. □

Corollary 3.3. *For all $\varphi_0 \in I_{\epsilon s}$ the level set $\varphi^{-1}[\varphi_0]$ intersects every level set of ξ, i.e., $\xi^{-1}[\alpha] \cap \varphi^{-1}[\varphi_0] \neq \emptyset$ for all $\alpha \in [0,1]$.*

Proof. Choose $q_1 \in \xi^{-1}[1] \cap \varphi^{-1}[\varphi_0]$ as guaranteed to exist by Lemma 3.2. Choose $q_0 \in \xi^{-1}[0] \cap \varphi^{-1}[\varphi_0]$ as guaranteed to exist since $\xi^{-1}[0]$ coincides with $\mathcal{W}^c$, the center manifold of f_1, which intersects each level set $\varphi^{-1}[\varphi_0]$ twice. Since for all $\varphi_0 \in I_{\epsilon s}$ the set $\varphi^{-1}[\varphi_0]$ is simply connected then there can be found a continuous curve, $c : [0,1] \to \varphi^{-1}[\varphi_0]$ connecting q_0 and q_1. The function $\xi \circ c(\alpha)$ must vary continuously between 0 and 1 and the result follows as claimed. □

Lemma 3.4. *A sufficient condition for the Jacobian of $f_2(q)$ evaluated at $\mathcal{W}^c - \|\nabla\varphi\|^{-1}[0]$ to have at least one eigenvalue with non-zero real part is that the control Lie algebra on B spans $\mathbb{R}^3$.*

Proof. Let $J(v)$ be the 3×3 skew symmetric matrix associated with the vector v. We will show that the rank condition implies a nonvanishing trace by explicitly computing the eigenvalues of $Df_2|_{\mathcal{W}^c}$:

$$\begin{aligned} Df_2 &= J(A)D^2\varphi - J(\nabla\varphi)DA \\ Df_2|_{\mathcal{W}^c} &= J(A)[D^2\varphi - \|\nabla\varphi\|DA] \end{aligned} \tag{3.18}$$

Now consider the change of coordinates $R = [A, A_\perp]$, where R defines a rotation matrix and $A_\perp = [A_2\ A_3]$ are orthogonal to A. Find the eigenvalues of Df_2:

$$\begin{aligned} &\det(Df_2 - \lambda I_3) = \det(R^T Df_2 R - \lambda I_3) \\ &= \det\left(\begin{bmatrix} 0 \\ A_\perp^T J(A) \end{bmatrix}[D^2\varphi - \|\nabla\varphi\|DA]R - \lambda I_3\right) \end{aligned} \tag{3.19}$$

Using Cramer's rule we obtain:

$$= -\lambda \det\left(A_\perp^T J(A)[D^2\varphi - \|\nabla\varphi\|DA]A_\perp - \lambda I_2\right) \tag{3.20}$$

One zero eigenvalue can be immediately factored out from the previous expression leaving as the second factor the characteristic polynomial of a 2×2 matrix whose trace we compute as:

$$\begin{aligned} &\operatorname{trace}\left(A_\perp^T J(A)[D^2\varphi - \|\nabla\varphi\|DA]A_\perp\right) \\ &= \operatorname{trace}\left(A_\perp^T J(A)D^2\varphi A_\perp\right) + \\ &\quad -\|\nabla\varphi\|\operatorname{trace}\left(A_\perp^T J(A)DAA_\perp\right) \end{aligned} \tag{3.21}$$

Since $D^2\varphi$ is symmetric, the first term in the sum just presented can be shown to vanish by noting:

$$\begin{aligned} &\operatorname{trace}\left(A_\perp^T J(A)D^2\varphi A_\perp\right) = \\ = \quad &\operatorname{trace}\left(\begin{bmatrix} A_2^T \\ A_3^T \end{bmatrix} J(A)D^2\varphi \begin{bmatrix} A_2 & A_3 \end{bmatrix}\right) \\ = \quad &\operatorname{trace}\left(\begin{bmatrix} A_3^T \\ -A_2^T \end{bmatrix} D^2\varphi \begin{bmatrix} A_2 & A_3 \end{bmatrix}\right) \\ = \quad &A_3^T D^2\varphi A_2 - A_2^T D^2\varphi A_3 = 0 \end{aligned} \tag{3.22}$$

Equation (3.21) becomes:

$$\begin{aligned} &= -\|\nabla\varphi\|\operatorname{trace}\left(A_\perp^T J(A)DAA_\perp\right) \\ &= -\|\nabla\varphi\|A_3^T(DA - DA^T)A_2 \end{aligned} \tag{3.23}$$

Since $A^T A_2 \equiv 0$ and $A^T A_3 \equiv 0$ we obtain the relations using the Lie derivative:

$$L_{A_3}(A^T A_2) = \left(A_2^T DA + A^T DA_2\right) A_3 = 0 \tag{3.24}$$
$$L_{A_2}(A^T A_3) = \left(A_3^T DA + A^T DA_3\right) A_2 = 0 \tag{3.25}$$

Replacing the previous relations into (3.23) we obtain:

$$\begin{aligned} & -\|\nabla\varphi\| A_3^T (DA - DA^T) A_2 \\ = & -\|\nabla\varphi\| \left(A^T DA_2 A_3 - A^T DA_3 A_2\right) \\ = & -\|\nabla\varphi\| A^T \left[A_2, A_3\right] \end{aligned} \tag{3.26}$$

Since the span of $\{A_2, A_3\}$ is equal to the span of $\{B_1, B_2\}$ then there exist continuous functions $\alpha_i(q), \beta_i(q)$ such that $A_i = \alpha_i B_1 + \beta_i B_2$, and (3.26) becomes

$$\begin{aligned} = & -\|\nabla\varphi\| A^T \left((\alpha_2\beta_3 - \alpha_3\beta_2)[B_1, B_2] + M_1 B_1 + M_2 B_2\right) \\ = & \ \gamma(q) A^T [B_1, B_2], \end{aligned} \tag{3.27}$$

where M_1, M_2 are matrix functions with left kernel A that contain derivatives of α_i, β_i and $\gamma(q) \neq 0,\ \forall q \notin \|\nabla\varphi\|^{-1}[0]$ is a continuous function. If the matrix $|B_1\ B_2\ [B_1, B_2]|$ is full rank then $A^T[B_1, B_2] \neq 0$. □

Lemma 3.5. *The Jacobian of $f_2(q)$ evaluated at $\mathcal{W}^c \cap \mathcal{Q}_s$ has two non-zero real part eigenvalues with the same sign.*

Proof. Let $\mathcal{L}_\alpha = \varphi^{-1}[\alpha],\ \alpha < \varphi_s$. The function $f_2|_{\mathcal{L}_\alpha}$ is a flow on a topological sphere. By Lemma 3.1 and Corollary 3.3 it only has two critical points with index +1 (Poincaré-Hopf [Arn73]). Therefore $Df_2|_{\mathcal{W}^c \cap \mathcal{Q}_s}$ has two non-zero real part eigenvalues with the same sign. □

Now consider the implicit equation,

$$\xi(q) = \xi^* \Leftrightarrow \|H(q)\nabla\varphi(q)\|^2 = \xi^* \|\nabla\varphi(q)\|^2 \tag{3.28}$$

At the goal any ξ^* satisfies (3.28). Although ξ is not defined at q^*, all of its level sets intersect at q^*. Finally, define the parameterized cone $\mathcal{C}_\gamma$ around $\mathcal{W}^c$, and its complement $\mathcal{C}_\gamma^c := \mathcal{Q} - \mathcal{C}_\gamma - \{q^*\}$, by:

$$\mathcal{C}_\gamma = \{q \in \mathcal{Q} - \{q^*\} : \xi(q) \leq \gamma\} \tag{3.29}$$

We follow by imposing conditions on H and A such that the vector field f_2 can afford the needed "escape" from $\mathcal{W}^c$.

Lemma 3.6. *Suppose system (3.1) satisfies assumptions A1-A3 and, hence, the previous lemmas. Then, there exists a function $\sigma : \mathcal{Q} \to \mathbb{R}$ that renders the system*

$$\dot{q} = \sigma(q)A(q) \times \nabla\varphi(q) = \bar{f}_2(q) \tag{3.30}$$

unstable at $\mathcal{W}^c \cap \mathcal{Q}_s$.

Proof. Let $\Xi : \mathcal{Q} \to \mathbb{C};\ \Xi(q) \mapsto \max(\mathrm{Re}(\{\lambda_1, \lambda_2, \lambda_3\}))$ return the eigenvalue with largest real part of the Jacobian of f_2 evaluated at the closest point to q that lives in $\mathcal{W}^c$. Consider the function $\sigma : \mathcal{Q}_s \to \mathbb{R}$ such that

$$\sigma(q) = \begin{cases} 1 & \text{if} \quad \mathrm{Re}(\Xi(q)) > 0 \\ -1 & \text{if} \quad \mathrm{Re}(\Xi(q)) \leq 0 \end{cases} \tag{3.31}$$

Partition $\mathcal{Q}_s$ into its two pieces, $\mathcal{Q}^+ = \{q \in \mathcal{Q}_s : \sigma(q) = 1\}$ and $\mathcal{Q}^- = \{q \in \mathcal{Q}_s : \sigma(q) = -1\}$ where $\mathcal{Q}^+ \cup \mathcal{Q}^- = \mathcal{Q}_s$ and $\mathcal{Q}^+ \cap \mathcal{Q}^- = \emptyset$.

In $\mathcal{W}^c \cap \mathcal{Q}^+$ two eigenvalues of $D\bar{f}_2$ have positive real part, rendering (3.30) unstable. In $\mathcal{W}^c \cap \mathcal{Q}^-$ we get that $D\bar{f}_2 = -Df_2$. Therefore the two nonzero real part eigenvalues of $D\bar{f}_2$ have a positive sign, also rendering (3.30) unstable. □

Corollary 3.7. *Under the conditions of the previous lemma, there can be found a $\tau \in (0, \infty)$ such that for all $q_0 \in \xi^{-1}[\delta/2]$ we have $\xi \circ \Phi^{\bar{f}_2}_\tau(q_0) \geq \delta$.*

Proof. Since $\mathcal{W}^c \cap \mathcal{Q}_s$ is unstable, for every level $\mathcal{L}_\alpha$ with $\alpha < \varphi_s$ and $q_\alpha \in \mathcal{L}_\alpha \cap \mathcal{W}^c$ there exists an $\kappa^+(\alpha) > 0$ and a neighborhood $\mathrm{B}_{\kappa^+}(q_\alpha) := \{q \in \mathcal{Q}_s | \|q - q_\alpha\| \leq \kappa^+\}$ such that every trajectory of $\bar{f}_2$ with initial condition inside $\mathrm{B}_{\kappa^+}(q_\alpha) - \mathcal{W}^c$ will eventually leave $\mathrm{B}_{\kappa^+}(q_\alpha)$. Let δ be the largest scalar such that $\mathcal{C}_\delta \subset \mathcal{N}^+ := \bigcup_\alpha \mathrm{B}_{\kappa^+}(q_\alpha)$. Let $\kappa^- > 0$ be the largest scalar such that $\mathcal{N}^- := \bigcup_\alpha \mathrm{B}_{\kappa^-}(q_\alpha) \subset \mathcal{C}_\gamma$, with $2\gamma = \delta$. The set $\mathcal{N} := \mathcal{N}^+ - \mathrm{int}\,(\mathcal{N}^-)$ is compact. Therefore, since $\mathcal{N} \cap \mathcal{W}^c = \emptyset$, trajectories of f_1 and $\bar{f}_2$ traverse $\mathcal{N}$ in finite time. Let $\tau_0(q_0) := \min\{t > 0 \mid \xi \circ \Phi^{\bar{f}_2}_t(q_0) = \delta\}$. Then define $\tau := \max\{\tau_0(q_0) | q_0 \in \mathcal{N}\}$. □

Figure 3.3 illustrates the steps used in the previous proof. Trajectories starting inside $\mathcal{N} - \mathcal{C}^c_\gamma$ will traverse $\partial\mathcal{C}_\gamma$ and $\partial\mathcal{C}_\delta$ in finite time.

3.1.2 A Hybrid Controller and Proof of its Local Convergence

Given the previous result define the time variables τ_1, τ_2 and the scalars $\gamma < \delta$ such that:

$$\tau_1(q, \gamma) := \begin{cases} \min\left\{t > 0 \mid \xi \circ \Phi^{f_1}_t(q) = \gamma\right\} \text{ if } q \in \mathcal{C}^c_\gamma \\ 0 \text{ otherwise} \end{cases} \tag{3.32}$$

$$\tau_2(q, \delta) := \begin{cases} \min\left\{t > 0 \mid \xi \circ \Phi^{\bar{f}_2}_t(q) = \delta\right\} \text{ if } q \in \mathcal{C}_\delta - \mathcal{W}^c \\ 0 \text{ otherwise} \end{cases} \tag{3.33}$$

I.e., τ_1 is the time to reach the γ neighborhood of $\mathcal{W}^c$ using vector field f_1 and τ_2 is the time to reach the boundary of $\mathcal{C}_\delta$ using vector field $\bar{f}_2$, thus escaping the γ neighborhood of $\mathcal{W}^c$. This results in the following maps:

$$\Phi_{\tau_1}^{f_1} \quad : \quad \overline{\mathcal{C}_\gamma^c} \to \partial\mathcal{C}_\gamma \tag{3.34}$$

$$\Phi_{\tau_2}^{\bar{f}_2} \quad : \quad \mathcal{Q}_s - \mathcal{W}^c \to \overline{\mathcal{C}_\delta^c} \subset \overline{\mathcal{C}_\gamma^c}, \tag{3.35}$$

where $\overline{\mathcal{C}}$ is the closure of $\mathcal{C}$. With $\delta = 2\gamma$ define the map $P : \mathcal{Q}_s - \mathcal{W}^c \to \partial\mathcal{C}_\gamma$

$$P(q) = \Phi_{\tau_1(\cdot,\gamma)}^{f_1} \circ \Phi_{\tau_2(q,2\gamma)}^{\bar{f}_2}(q) \tag{3.36}$$

and consider the recursive equation:

$$q_{k+1} = P(q_k). \tag{3.37}$$

The set $\partial\mathcal{C}_\gamma$ can be interpreted as a Poincaré section for the discrete system (3.37). We are now ready to present the final result:

Theorem 3.8. *There exists an iteration number, $N : \mathcal{Q}_s \to \mathbb{N}$ such that the iterated hybrid dynamics, P^N brings $\mathcal{Q}_s$ to $\mathcal{Q}_\epsilon$.*

Proof. Define

$$N := \min\left\{n \in \mathbb{N} | 0 \leq N \leq N_\epsilon | \varphi \circ P^n(q_0) \leq \epsilon\right\}, \tag{3.38}$$

and $\Delta\varphi(q) := \varphi \circ P(q) - \varphi(q)$. Since $\mathcal{Q}_s$ is a compact set, it follows that $|\Delta\varphi|$ achieves its minimum value, Δ_ϵ, on that set, hence at most $N_\epsilon :=$ ceiling$(\varphi_s - \epsilon)/\Delta_\epsilon$ iterations are required before reaching $\mathcal{Q}_\epsilon$. □

Note that all initial conditions in the pre-image of the "local surround", $\mathcal{R} := \bigcup_{t>0} \Phi_{-t}^{f_1}(\mathcal{Q}_s - \mathcal{W}^c)$ are easily included in the basin of the goal, $\mathcal{Q}_s$, by an initial application of the controller u_1. While it is difficult to make any general formal statements about the size of $\mathcal{R}$, we show in the next section that for all the examples we have tried, the "missing" initial conditions, $\mathcal{Q} - \mathcal{R} = \mathcal{Z}$, comprise a set of empty interior (in all but one case $\mathcal{Z}$ is actually empty) because all of $\mathcal{W}^c$, excepting at most a set of measure zero, is included in $\mathcal{Q}_s$. In configuration spaces with more complicated topology, there is no reason to believe that this pleasant situation will prevail. To summarize, we rewrite the strategy presented in Section 3.1.1 using now the explicit input controls:

1. $\forall q_0 \in \mathcal{W}^c$ use the input

$$u_3 := \begin{bmatrix} \alpha_1 & \alpha_2 \end{bmatrix}^T, \tag{3.39}$$

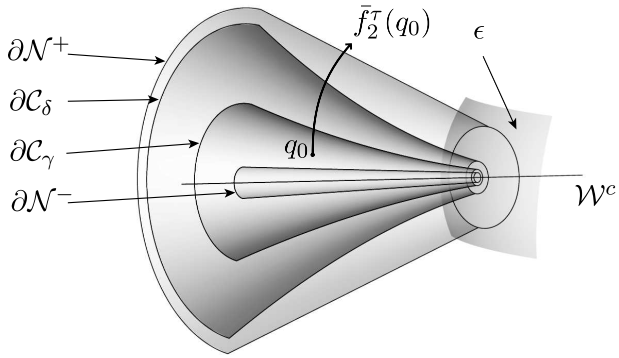

$\bar{f}_2^{\mathcal{T}}(q_0)$
ϵ
$\partial\mathcal{N}^+$
$\partial\mathcal{C}_\delta$
$\partial\mathcal{C}_\gamma$
q_0
$\partial\mathcal{N}^-$
$\mathcal{W}^c$

Figure 3.3: Illustration of the construction used in the proof of Corollary 3.7.

for a small amount of time t_3, where α_1, α_2 are scalar constants not both simultaneously zero, such that $\varphi \circ \Phi_{t_3}^{f_3}(q_0) < 1$ and $\Phi_{t_3}^{f_3}(q_0) \notin \mathcal{W}^c$, with $f_3(q) := B(q)u_3$.

2. $\forall q_0 \in \mathcal{Q}_s - \mathcal{W}^c$, follow successive applications of (3.37), i.e. use the inputs to equation (3.1):

$$u_1(q) := -B^\dagger(q)\nabla\varphi(q) \tag{3.40}$$
$$u_2(q) := \sigma(q)B^\dagger(q)J(A(q))\nabla\varphi(q) \tag{3.41}$$

3. $\forall q_0 \in \mathcal{R} - \mathcal{Q}_s$, use the input u_1 for time t until $\Phi_t^{f_1}(q_0) \in \mathcal{Q}_s$.

Having discussed the volume of convergence, the next most crucial question bearing on the practicality of this scheme, speed of convergence, will also be addressed on a case by case basis in Section 3.5.4 using two additional formal ideas that we now present.

3.2 Limit cycles in the level sets of φ

In many practical applications, switching between controllers f_1 and f_2 using a small δ-neighborhood is far too conservative. It may be possible to escape $\mathcal{W}^c$ by more than just the small collar $\xi^{-1}[\delta]$. In Section 3.5.4 we show an example where the trajectories of f_2 flow from $\mathcal{W}^c \cap \varphi^{-1}[\varphi_0]$ with positive real part eigenvalues, where $\varphi_0 < \varphi_s$ is some energy, to $\mathcal{W}^c \cap \varphi^{-1}[\varphi_0]$ with negative real part eigenvalues, crossing in between the stable manifold at the goal $\mathcal{W}^s$. If we could recognize the passage into $\mathcal{W}^s$ and switch off controller u_2 (i.e. turn $\mathcal{W}^s$ into an attractor of a suitable modified form of f_2) then a final application of controller u_1 is guaranteed to achieve the goal state, q^*. The hope of reworking the form of u_2 so that the resulting closed loop vector field, f_2, has its forward limit set solely in $\mathcal{W}^s$ thus raises the question of when there exists limit cycles in the level sets of φ for the flow of f_2. More importantly, we seek a condition that guarantees that every trajectory of f_2, starting in a small neighborhood of $\mathcal{W}^c$, can intersect $\mathcal{W}^s$ either by forward or inverse time integration of system (3.10). Note that f_2 generates a planar flow, making the Bendixon's criteria a natural candidate for such conditions. Several authors [Fec01b, Fec01a, Li00, GG00] have developed extensions to Bendixson's criteria for higher dimensional spaces, obtaining, in general, conditions that preclude invariant sub-manifolds on some set. For systems with first integrals, such as some classes of systems that result from nonholonomic constraints, the conditions simplify to a divergence style test. Feckan's theorem (see Appendix A and [Fec01b]) states that in open subsets where $\text{div} f_2 \neq 0$ there can exist no invariant submanifolds of any level precluding cyclic orbits. Note that the previous result does not preclude quasi-periodic orbits. In Section 3.5.3 we give an example that, by having $\varphi_s < 1$, results in quasi-periodic orbits on a torus. Using Cauchy-Riemann, the divergence of the vector field f_2 results in:

$$\begin{aligned} \text{div}(f_2) &= \text{div}\left(A(q) \times \nabla\varphi(q)\right) \\ &= \left(\nabla \times A(q)\right)^T \nabla\varphi(q) \\ &= A_R(q)^T \nabla\varphi(q) \end{aligned} \tag{3.42}$$

In the examples described here, the set $\mathcal{D} := \left\{q \in \mathcal{Q} : A_R(q)^T \nabla\varphi(q) = 0\right\}$ is a 2-manifold that contains the goal. If $\mathcal{D} \cap \mathcal{W}^c = \{q^*\}$ and $\mathcal{D}$ is not itself invariant for f_2 then we are guaranteed that there exists no limit cycles on the level sets of φ.

3.3 Euler-Poincaré characteristic or index on spheres

We revise here informally some results from index theory that shed light into the types of equilibria, sources, sinks, or saddles, that can occur in the planar flow of f_2 at $\mathcal{W}^c$. Corollary 3.7 assumes that $\mathcal{Z} = \emptyset$. However, by analyzing the type of equilibria on the planar vector fields f_2 (since they reside in level sets, locally homeomorphic to planes) one can infer on the volume of $\mathcal{Z}$.

Proposition 3.9 (Andronov *et al.* [AVK66, GH83])**.**

- *The index of a sink, a source, or a centers is* $+1$.
- *The index of a hyperbolic saddle point is* -1.
- *The index of a closed orbit is* $+1$.
- *The index of a closed curve not containing any fixed points is* 0.
- *The index of a closed curve is equal to the sum of the indices of the fixed points within it.*

Theorem 3.10 (Arnold [Arn73])**.** *The sum of the indices of all the singular points of a field on the sphere is independent of the choice of field. This sum equals 2.*

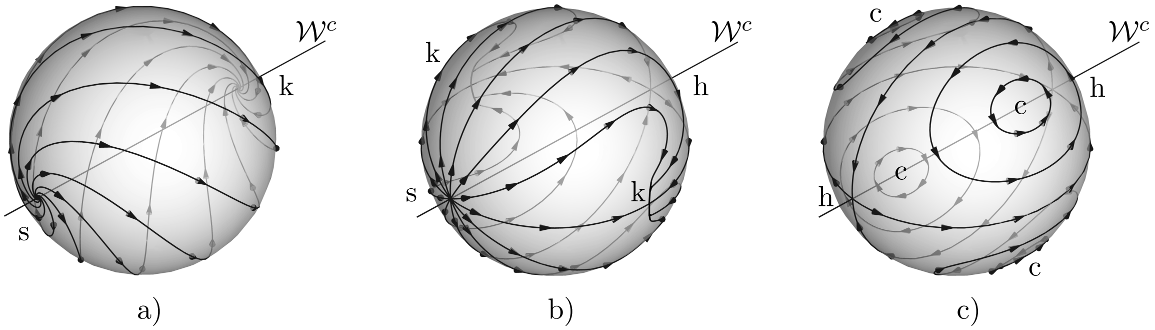


Figure 3.4: Euler-Poincaré characteristic on spheres. The letter "s" stands for source, "k" for sink, "h" for hyperbolic saddle, and "c" for center. **a)** example of one sink (+1) and one source (+1), $A = [x_2 \ -x_1 \ 1]$. **b)** example of one hyperbolic saddle (-1), one source and two sinks (+3), $A = [x_2 \ x_1 \ 0]$. **c)** example of two hyperbolic saddles (-2) and four centers (+4), $A = [-x_2x_3 \ -x_1 \ 0]$.

Given Lemma 3.1 and observing that close to the origin the Center Manifold Theorem states that $\mathcal{W}^c$ is 1-dimensional, therefore intersecting low energy level sets of φ in two unique points, we conclude that f_2 will have 2 equilibria points on the "small" spheres close to the origin. Theorem 3.10 along with Lemma 3.4 and Proposition 3.9 show that such equilibria must be either a sink or a source in order to have index +2. If one of the equilibria is a hyperbolic saddle then there must exist 3 other sinks or sources to sum the index to +2, contradicting Lemma 3.1. Figure 3.4 illustrates different equilibria configurations on spherical level sets for different nonholonomic constraints defined in $\mathbb{R}^3$. Only the constraint in figure Figure 3.4.a) verifies assumption A3, resulting in a single source and a sink. The examples of figures 3.4.b)-c) fail Lemma 3.1.

3.4 Computational heuristic substitutes for σ

The σ function introduced in Lemma 3.6 modifies the flow of f_2 rendering the center manifold unstable. Having that property is sufficient for stabilization, but more can be accomplished. By careful craft of σ, one can minimize the number of switches between controllers f_1 and $\bar{f}_2$ necessary to reach the desired neighborhood of the goal. If the stable manifold $\mathcal{W}^s$ matches the zero set of σ and $\mathcal{W}^s$ is made attractive by $\bar{f}_2$ for any point in $\mathcal{Q}_s$ then one gets $\Phi_\infty^{f_1} \circ \Phi_\infty^{\bar{f}_2}(\mathcal{Q}_s) = q^*$, i.e., only 2 steps are necessary to reach the goal. Note however, that if the zero set of σ intercepts $\mathcal{W}^c$ more than one time, then there exists the possibility that the system will not progress to the goal. In this section we present practical computational heuristic substitutes for σ with zero sets that locally approximate $\mathcal{W}^s$.

3.4.1 Divergence

Following the results obtained in Section 3.2, using the divergence operator seems natural. In the neighborhood of the center manifold, if the eigenvalues have all negative real part, then the divergence will be negative, reversing the flow of f_2. If the real part of the eigenvalues are all positive the divergence operator will not reverse the flow. Even in the event of the eigenvalues having simultaneously positive and negative real parts, the sign of the divergence will not change the instability. Define the function $\sigma_1 : \mathcal{Q} \to \mathbb{R}$ by

$$\sigma_1(q) := \operatorname{div}(f_2) = A_R(q)^T . \nabla\varphi(q). \tag{3.43}$$

3.4.2 Maximizing ξ

Another way of escaping $\mathcal{W}^c$ is to follow the direction that maximizes ξ. By definition its maximum is the unity. Let the function $\sigma_2 : \mathcal{Q} \to \mathbb{R}$ be defined by:

$$\sigma_2(q) := \nabla\xi(q)^T f_2(q). \tag{3.44}$$

Using $V(q) = \xi(q) \geq 0$ as a candidate control Lyapunov function for the system $\dot{q} = \sigma_2(q) f_2(q)$ we observe that

$$\begin{aligned} \dot{V} &= \nabla\xi^T \left(\nabla\xi(q)^T f_2(q)\right) f_2(q) \\ &= \|\nabla\xi(q) f_2(q)\|^2 \geq 0 \end{aligned} \tag{3.45}$$

The function $\sigma_2(q)$ destabilizes (3.30) at $\mathcal{W}^c$ if there exists a $\xi^* > 0$ such that the set $\{\dot{V}(\mathcal{C}_{\xi^*} - \mathcal{W}^c) = 0\}$ does not define an invariant manifold (following La Salle's).

3.4.3 Stable manifold approximation

The third heuristic computation of σ presented here aims directly at approximating the stable manifold so to minimize the number of switches between controllers f_1 and $\bar{f}_2$. Suppose there exists a function $G : \mathcal{Q} \to \mathbb{R}$ whose pre-image $G^{-1}[0]$ is $\mathcal{W}^s$. Using the same argumentation as in equations (3.44) and (3.45), and replacing ξ by $G(q)^2$, we obtain:

$$\sigma_{3a}(q) := -\nabla(G(q)^2)^T f_2(q). \tag{3.46}$$

Again, taking $V(q) = G(q)^2 \geq 0$ as a candidate control Lyapunov function, we observe that the system $\dot{q} = \sigma_{3a}(q) f_2(q)$ will have its forward limit set in $\mathcal{W}^s$ if $\forall q \in \mathcal{Q} - \mathcal{W}^s : \dot{V}(q) = -\|2G(q)\nabla G(q)^T f_2(q)\|^2 \neq 0$. Note that it is possible for the zero set of $\nabla(G(q)^2)^T f_2(q)$ not to be contained in $\mathcal{W}^s$, breaking the desired result. In some cases however, it is possible to use the function

$$\sigma_{3b}(q) := sG(q) \tag{3.47}$$

where $s \in \{-1, +1\}$. The sign s is chosen so that $\mathcal{W}^c$ can be made unstable, i.e. in a neighborhood of $\mathcal{W}^c$ the signs of $\operatorname{div}(f_2)$ and G should match.

In general, finding an exact approximation of $\mathcal{W}^s$ by an algebraic implicit equation is unattainable since that requires solving a set of partial differential equations [GH83]. We proceed by finding a k-order polynomial approximation to $\mathcal{W}^s$, denoted by $\widehat{\mathcal{W}}^s_k$. Without loss of generality, we assume that the goal is at the origin, $q^* = 0$, and the tangent of $\mathcal{W}^s$ evaluated at the origin is the span of the first two canonical base vectors[4]. Let h be the "aligned" version of f_1. We seek to find a function[5] $g : \mathbb{R}^2 \to \mathbb{R}$ such that its graph is $\mathcal{W}^s$, i.e., $x_3 = g(x_1, x_2)$. Define the implicit function G as:

$$G(x_1, x_2, x_3) := g(x_1, x_2) - x_3 \tag{3.48}$$

Let $\hat{g}_k$ be a k-order polynomial approximation of g at the origin parameterized by $\gamma_{i,j}$:

$$\hat{g}_k(x_1, x_2) = \sum_{\substack{i,j \geq 0 \\ i+j \leq k}} \frac{x_1^i x_2^j}{i!j!} \gamma_{i,j} \tag{3.49}$$

and let $\hat{h}_k$ be the k-order Taylor expansion of h at the origin:

$$\hat{h}_k(x_1, x_2, x_3) = \sum_{\substack{i,j,l \geq 0 \\ i+j+l \leq k}} \frac{x_1^i x_2^j x_3^l}{i!j!l!} \left(\frac{\partial^i}{\partial x_1^i} \frac{\partial^j}{\partial x_2^j} \frac{\partial^l}{\partial x_3^l} h \right) \tag{3.50}$$

For the system $\dot{q} = h(q)$ the manifold $G(q) = 0$ is invariant. Therefore for trajectories that start in $G(q) = 0$ we obtain

$$\dot{G}(q) = \nabla G(q)^T h(q) = 0. \tag{3.51}$$

Replacing g by $\hat{g}_k$, h by $\hat{h}_k$, and x_3 by $\hat{g}_k(x_1, x_2)$ we obtain the following approximation equation:

$$\left(\begin{bmatrix} \frac{\partial \hat{g}_k}{\partial x_1} & \frac{\partial \hat{g}_k}{\partial x_2} & -1 \end{bmatrix} \cdot \hat{h}_k \right) \circ (x_1, x_2, \hat{g}_k) = 0 \tag{3.52}$$

Equation (3.52) is polynomial in $\gamma_{i,j}$ and in x_i. Since by assumption the tangent space of

[4] It is always possible to align the tangent of $\mathcal{W}^s$ at the goal with the span of the first two canonical base vectors by means of a translation $p = q - q^*$ and a rotation R. The matrix R is obtained by applying the Gram-Schmidt orthogonalization on the matrix of the eigenvectors of $Df_1(q)|_{q=q^*}$ with eigenvalues sorted by absolute magnitude.

[5] In general this function may not exist outside a neighborhood of the origin

$\mathcal{W}^s$ at the origin is the plane $x_3 = 0$, we immediately obtain:

$$h(0) = \left.\frac{\partial h}{\partial x_1}\right|_{q=0} = \left.\frac{\partial h}{\partial x_2}\right|_{q=0} = 0, \text{ and} \tag{3.53}$$

$$\gamma_{0,0} = \gamma_{1,0} = \gamma_{0,1} = 0 \tag{3.54}$$

The 2nd order terms of $\gamma_{i,j}$ are obtained by solving the following equation evaluated at the origin, where h_i is the i-th component of h:

$$\begin{bmatrix} \gamma_{0,2} \\ \gamma_{1,1} \\ \gamma_{2,0} \end{bmatrix} = \begin{bmatrix} \frac{\partial h_2}{\partial x_2} & \frac{\partial h_1}{\partial x_2} & 0 \\ \frac{\partial h_2}{\partial x_1} & \frac{\partial h_1}{\partial x_1} + \frac{\partial h_2}{\partial x_2} & \frac{\partial h_1}{\partial x_2} \\ 0 & \frac{\partial h_2}{\partial x_1} & \frac{\partial h_1}{\partial x_1} \end{bmatrix}^{-1} \cdot \begin{bmatrix} \frac{1}{2}\frac{\partial^2 h_3}{\partial x_2^2} \\ \frac{\partial^2 h_3}{\partial x_1 \partial x_2} \\ \frac{1}{2}\frac{\partial^2 h_3}{\partial x_1^2} \end{bmatrix} \tag{3.55}$$

Note that a measure of the curvature of $\mathcal{W}^c$ at the origin is given by $\gamma_{1,1}^2 - \gamma_{2,0}\gamma_{0,2}$. The higher order terms of $\gamma_{i,j}$ are obtained recursively by incrementally increasing k in equation (3.52) and solving for $\gamma_{i,j}$ with $i + j = k$.

The computational process for stable manifold approximation presented in this section is readily generalizable to n-dimensional systems by defining the implicit function $G : \mathbb{R}^n \to \mathbb{R}^{n-m}$ to be $G(q) = g(x_1, \ldots, x_m) - [x_{m+1} \cdots x_n]^T$, with $g : \mathbb{R}^m \to \mathbb{R}^{n-m}$. In similar fashion, Equations (3.49) to (3.52) can be written for a n-dimensional system where the computation process remains the same. Please see [LK04] for a reference on the n-dimensional computational algorithm implemented for this thesis.

3.5 Simulations

In this section we present simulations for the Heisenberg system using a norm-like navigation functions that results in spherical level sets. This allows for very simple closed form expressions of the σ functions introduced in Section 3.4. We proceed with numerical examples of the unicycle with a norm-like navigation function and two real robotic navigation applications utilizing 1 and 3 beacons for visual servoing.

3.5.1 The Heisenberg System

The Heisenberg system, revised in Section 1.2.4, assumes the following set of equations for $\mathbb{R}^3$:

$$\begin{aligned} \dot{x}_1 &= u_1 \\ \dot{x}_2 &= u_2 \end{aligned} \tag{3.56}$$

$$\dot{x}_3 = x_1 u_2 - x_2 u_1$$

The nonholonomic constraint is written in the form $\dot{x}_3 = x_1\dot{x}_2 - x_2\dot{x}_1$, resulting in

$$A(q) = (1 + x_1^2 + x_2^2)^{-1} \begin{bmatrix} x_2 & -x_1 & 1 \end{bmatrix}. \tag{3.57}$$

Consider the following simply connected configuration space: let $\mathcal{Q} = \{q \in \mathbb{R}^3 : \|q\| \leq 1\}$ and

$$\varphi(q) = x_1^2 + x_2^2 + x_3^2 = \|q\|^2. \tag{3.58}$$

We conclude that φ is a navigation function in $\mathcal{Q}$ since $\varphi(\partial\mathcal{Q}) = 1$ and φ has a unique minima at the origin. Note that all the level sets of φ are spheres, hence φ_s can be stretched to the boundary of $\mathcal{Q}$ resulting in $\mathcal{R} \equiv \mathcal{Q}$. The remaining quantities are computed from A and φ:

$$B^T(q) = \begin{bmatrix} 1 & 0 & -x_2 \\ 0 & 1 & x_1 \end{bmatrix} \tag{3.59}$$

$$H(q) = \frac{1}{1 + x_1^2 + x_2^2} \begin{bmatrix} 1 + x_1^2 & x_1 x_2 & -x_2 \\ x_1 x_2 & 1 + x_2^2 & x_1 \\ -x_2 & x_1 & x_1^2 + x_2^2 \end{bmatrix} \tag{3.60}$$

$$\xi(q) = \frac{\left(x_1^2 + x_2^2\right)\left(1 + x_1^2 + x_2^2 + x_3^2\right)}{\left(1 + x_1^2 + x_2^2\right)\left(x_1^2 + x_2^2 + x_3^2\right)} \tag{3.61}$$

For the navigation function (3.58) the center manifold of (3.4) is $\mathcal{W}^c = \{q \in \mathcal{Q} : x_1 = 0 \wedge x_2 = 0\}$ and the stable manifold is $\mathcal{W}^s = \{q \in \mathcal{Q} : x_3 = 0\}$. The heuristic computations of σ described in Section 3.4, are:

$$\sigma_1 = x_3 \frac{2 + x_1^2 + x_2^2}{(1 + x_1^2 + x_2^2)^{3/2}} \tag{3.62}$$

$$\sigma_2 = x_3 \frac{2\xi^{-1}}{(1 + x_1^2 + x_2^2)^{3/2}} \tag{3.63}$$

$$\sigma_{3a} = x_3 \frac{2(x_1^2 + x_2^2)}{\sqrt{1 + x_1^2 + x_2^2}} \tag{3.64}$$

$$\sigma_{3b} = x_3 \tag{3.65}$$

In the particular case of the Heisenberg system with a norm-like navigation function, the stable manifold can be described by the equation $x_3 = 0$, appearing as a subset of the zero sets of each of the σ_i functions. Using the method described in Section 3.4.3 one obtains $G(q) = x_3$. For function σ_{3b} the sign is chosen such that $\sigma_{3b} = +x_3$. Since all the functions

σ are approximately scaled versions of each other we use:

$$\sigma(q) = x_3 \tag{3.66}$$

Following the analysis of Section 3.2 we find the zero set of $\text{div}(f_2) = \sigma_1$ to be $\mathcal{W}^s$. This set is not invariant for f_2, therefore precluding limit cycles anywhere in $\mathcal{Q}$. Assuming that the north hemisphere of the level sets of φ is repellent, we observe that trajectories starting there will travel to the south hemisphere, always intercepting the stable manifold of f_1 in the equator. The control functions for the input of (3.1) are:

$$u_1(q) = -2 \begin{bmatrix} x_1 + \dfrac{x_2 x_3}{1 + x_1^2 + x_2^2} \\ x_2 + \dfrac{x_1 x_3}{1 + x_1^2 + x_2^2} \end{bmatrix} \tag{3.67}$$

$$u_2(q) = 2x_3 \begin{bmatrix} x_2 + x_1 x_3 \\ -x_1 + x_2 x_3 \end{bmatrix} \tag{3.68}$$

Table 3.1 illustrates simulation results for different pairs of δ and ϵ defined in Section 3.1.2. Since the zero set of function σ matches the stable manifold, all the simulations for whom ξ is maximized result in convergence in a single iteration. Bloch *et al* present in [BDK00] a general solution for the stabilization of the Heisenberg system in $\mathbb{R}^n$. It is interesting to compare their approach with the one presented here. In their paper, two variables x and Y represent the unconstrained and constrained variables respectively. In $\mathbb{R}^3$ these are $x = [x_1\ x_2]$ and $Y = x_3$. The algorithm described, in the general form, alternates from minimizing $\|x\|$ with $\|Y\|$ constant to minimizing $\|Y\|$ with $\|x\|$ constant. Similarly, in our algorithm we seek to maximize ξ while keeping φ constant, alternating with minimizing φ. The main difference lies in the "energy" function used as a measure of convergence, either $\|Y\|$ or φ. The prior has the advantage of being defined in $\mathbb{R}^n$. The latter has the advantage of allowing more general energy functions. Figure 3.5 illustrates comparative simulation results for the Bloch-Dracunov-Kinyon algorithm and the one presented in this paper. One can observe that the state variable plots are similar. In both cases the constrained variable x_3 is first driven to zero, in effect reaching the stable manifold. Next, x_1 and x_2 are simultaneously driven to zero without changing x_3. Here, the difference lies in the trajectories living in a cylinder $\|x\| = const.$ or a sphere $\varphi(q) = const.$ during the initial step.

3.5.2 Unicycle with a norm-like Navigation Function

Consider the unicycle systems equation (1.34) with the same simply connected configuration space $\mathcal{Q} = \{q \in \mathbb{R}^3 : \|q\| \leq 1\}$ and navigation function $\varphi(q) = \|q\|^2$. Below we present the σ_3 function for different approximation levels k. Note that for this particular

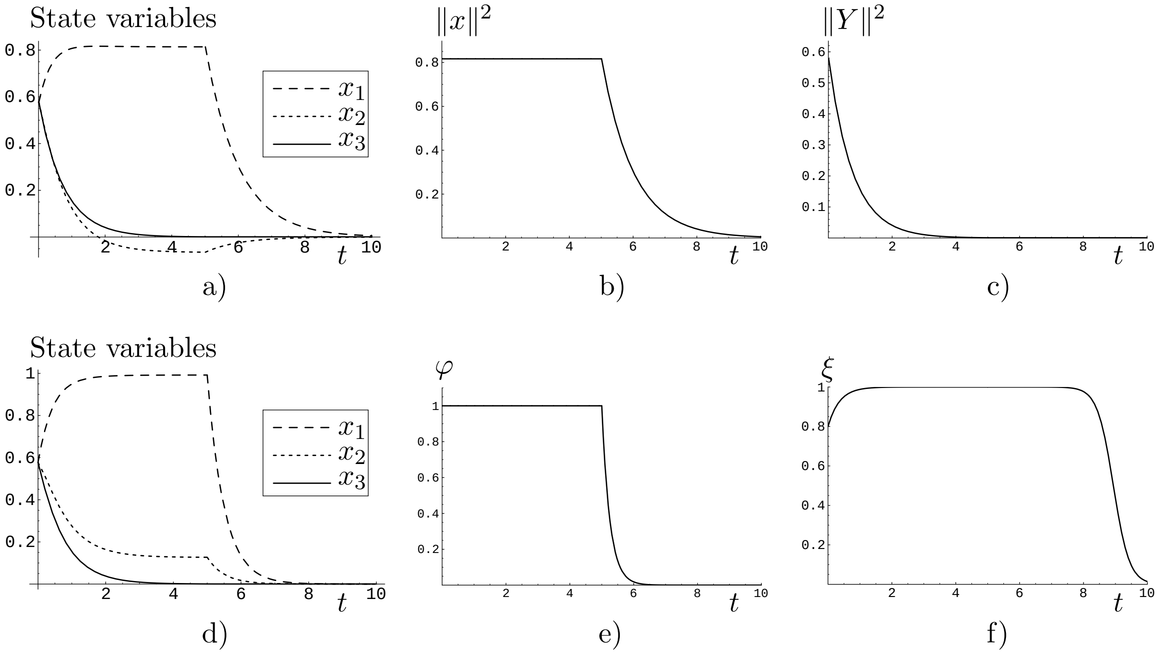


Figure 3.5: **a)-c)** Simulation of the Bloch-Dracunov-Kinyon algorithm. **d)-f)** Simulation of the algorithm presented in this chapter.

configuration all the σ_i functions differ from each other.

$$\sigma_3 = \begin{cases} x & \text{if} \quad k = 1 \\ x + \dfrac{y\theta}{2} & \text{if} \quad k = 2 \\ x + \dfrac{y\theta}{2} + \dfrac{y\theta^3}{48} & \text{if} \quad k = 4 \\ x + \dfrac{y\theta}{2} + \dfrac{y\theta^3}{48} + \dfrac{y\theta^5}{480} & \text{if} \quad k = 6 \end{cases} \tag{3.69}$$

Table 3.1 compiles the simulation results. One can conclude, as expected, that the number of iterations of (3.37) required to reach a fixed neighborhood of the goal dramatically decreases when δ increases. Moreover, although σ_1 and σ_2 do a good job at escaping $\mathcal{W}^c$, they require more iterations on average than the higher order approximation of $\mathcal{W}^s$. The best results, in terms of iteration number, are obtained for σ_3 when $k \geq 2$, where the approximation of $\mathcal{W}^s$ is very good. Figure 3.6 illustrates two representative simulations with the same initial condition and different order of approximation of the stable manifold. Again as expected, using a higher order approximation results in faster convergence to the goal after iterating controllers f_2 followed by f_1.

Table 3.1: Simulations for the Heisenberg systems and a unicycle with a norm type navigation function. Each entry corresponds to the average number of switches "N" from a random initial condition 5 meters away from the goal for 50 simulations. We use the parameter $\gamma = 10^{-3}$.

$\delta =$	0.2		0.5		1	
$\epsilon =$	1 cm	1 mm	1 cm	1 mm	1 cm	1 mm
Heisenberg						
$\sigma = x_3$	3.4	12.8	1.9	4.4	1.0	1.0
Unicycle						
σ_1	27.9	37.8	9.9	13.2	2.9	3.1
σ_2	29.3	41.0	9.8	12.9	2.7	3.2
$\sigma_3, k = 1$	28.7	38.5	10.3	12.5	2.7	3.0
$\sigma_3, k = 2$	28.2	38.3	9.7	13.5	1.3	1.8
$\sigma_3, k = 4$	29.9	37.8	10.6	13.4	1.3	1.5
$\sigma_3, k = 6$	30.2	37.9	9.2	13.8	1.4	1.5

3.5.3 A single beacon visual servoing problem

We present here a simulation of the visual servoing problem of positioning a robot in relation to a single engineered beacon. This problem has been addressed by Kantor [KR03] and Bhattacharya [BMCH04], as discussed in the introduction. Their alternative solution approach can be readily compared to the present scheme. Since the visibility set (the complete configuration space) is not a topological sphere, this example also provides a simple illustration of the additional effort required to reason about initial conditions outside the "surround". Figure 3.2 illustrates how the level sets, which are topological spheres (the components of $\mathcal{Q}_s$), form a proper subset of the toroidal visibility set in this case.

We applied the algorithm developed in Section 3.1 to this problem, using again, the unicycle motion model. The navigation function is developed in double polar coordinates and is brought back to $\mathsf{SE}(2)$ by the change of coordinates $c : \mathsf{SE}(2) \rightarrow S^1 \times S^1 \times \mathbb{R}^+$:

$$\begin{bmatrix} \eta \\ \mu \\ d \end{bmatrix} = c(x, y, \theta) := \begin{bmatrix} \arctan(y/x) \\ \theta - \arctan(y/x) \\ \sqrt{x^2 + y^2} \end{bmatrix} \tag{3.70}$$

The navigation function reflects the following physical attributes of the sensor:

1. The robot must be in an interval of distances away from the beacon, so as to not get too close or too far away from it, specifically $d_m < d < d_M$.

2. The robot's camera must face the beacon at all times, encoded as $\mu_m < \mu < \mu_M$,

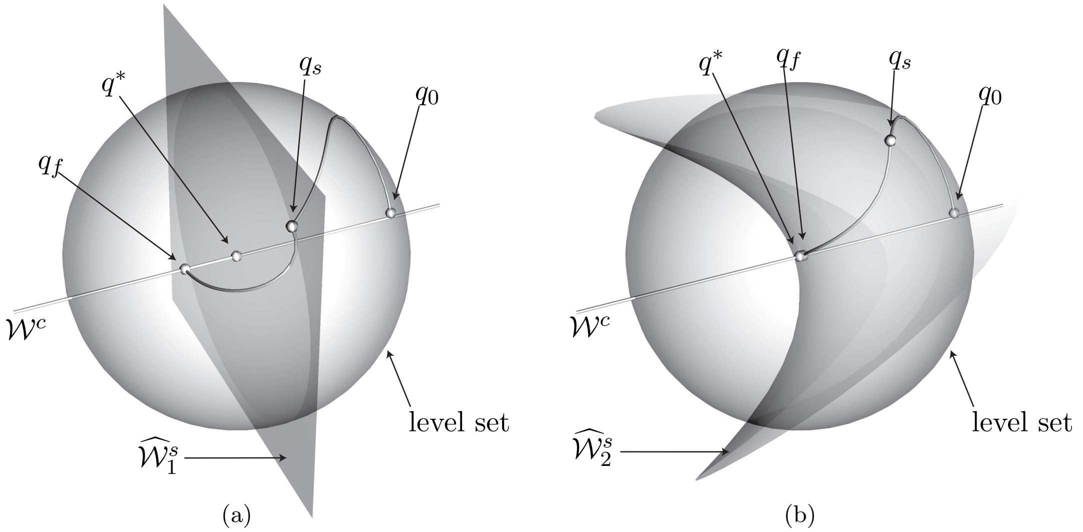


Figure 3.6: Simulation of a unicycle with a norm-like navigation function using: **a)** 1st order approximation of stable manifold **b)** 2nd order approximation of stable manifold. The point q_0 represents the initial condition, q_s the switching from controller f_2 to controller f_1, q_f is the final point, and q^* is the goal location.

where μ_m, μ_M are the field of view boundaries of the camera in polar coordinates.

Consider the potential function:

$$\bar{\bar{\varphi}} := \frac{\left(2 - \cos(\eta - \eta^*) - \cos(\mu - \mu^*) + (d - d^*)^2\right)^k}{(1 - \cos(\mu - \mu_m))(1 - \cos(\mu - \mu_M))(d_M - d)(d - d_m)} \tag{3.71}$$

For the previous potential function we have:

- The goal location in $\mathsf{SE}(2)$, denoted by (x^*, y^*, θ^*), is mapped by c to (η^*, μ^*, d^*). We thus assume that the final orientation of the robot is important.
- The cosine functions are used here, e.g. $(1 - \cos(\mu - \mu_m))$, since the state variables η and μ live in S^1. The desired goal is actually $(\eta^* + 2k_1\pi, \mu^* + 2k_2\pi, d^*)$ with $k_1, k_2 \in \mathbb{N}$.
- k is a shaping term.

By construction, $\bar{\bar{\varphi}}$ explodes at the obstacles and is zero at the goal. The resulting navigation function $\bar{\varphi} : S^1 \times S^1 \times \mathbb{R}^+ \to [0, 1]$ is the squashed version of $\bar{\bar{\varphi}}$, with $\kappa > 0$:

$$\bar{\varphi} := \frac{\bar{\bar{\varphi}}}{\kappa + \bar{\bar{\varphi}}} \tag{3.72}$$

In the configuration space $\mathcal{Q} \subset \mathsf{SE}(2)$, the navigation function φ is the composition

$$\varphi(q) := \bar{\varphi} \circ c(q) \tag{3.73}$$

and the gradient is the pullback: $\nabla\varphi(q) = Dc^T(q)\nabla\bar{\varphi} \circ c(q)$. Note that by imposing a

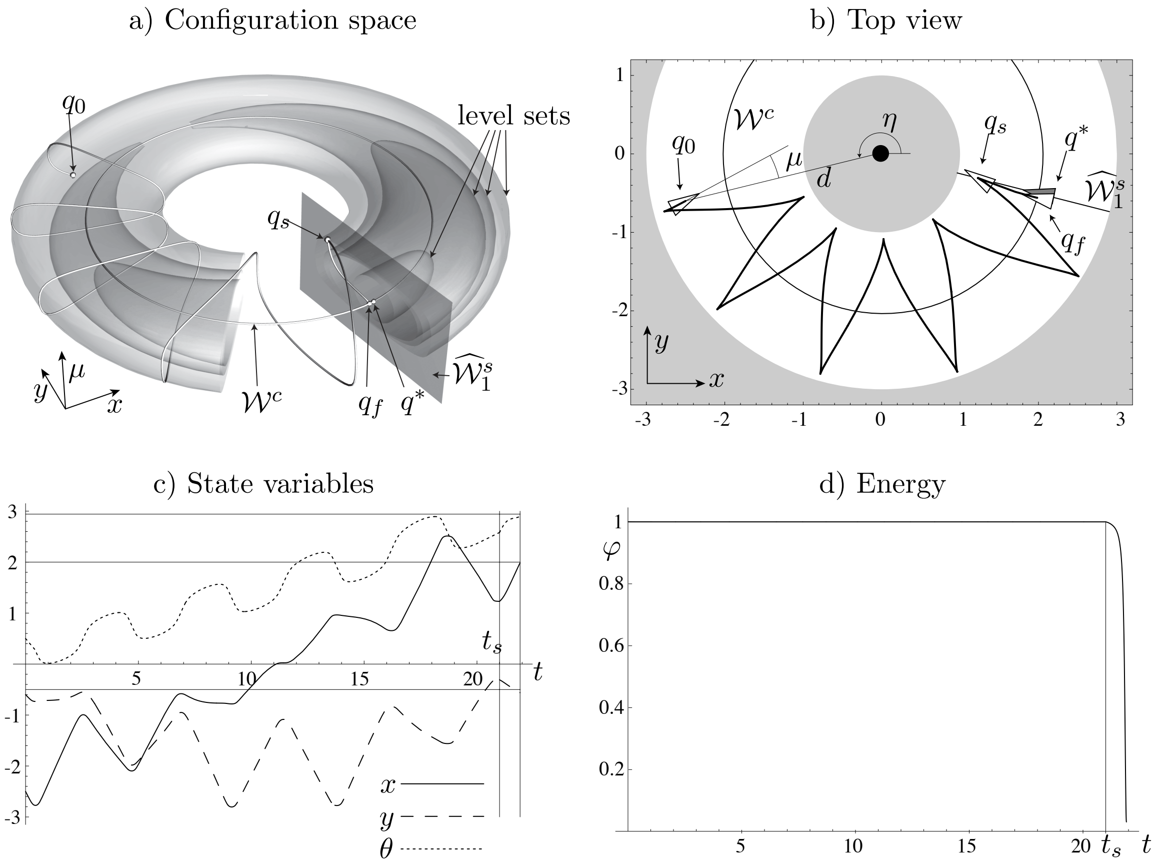


Figure 3.7: Simulation of the hybrid controller operating in the visible set of a single beacon landmark, described in Section 3.5.3. The initial configuration is q_0, the controller switches at time t_s in position q_s and the final configuration is q_f. **a)** Configuration space plotted on (x, y, μ) for readability purpose. **b)** Top view. The visual beacon is represented by the large black dot. The gray areas violate the visual constraints. **c)** and **d)** State variables and energy plots.

minimum distance to the beacon d_m, the configuration space is not simply connected. It is in fact homeomorphic to a solid torus as illustrated in Figure 3.7. This results in $\varphi_s < 1$. Here, some level sets are topological torus and others topological spheres. However, it is observed that the center manifold $\mathcal{W}^c$ is a circle, every level set homeomorphic to the sphere intersects $\mathcal{W}^c$ and every level set homeomorphic to the torus does not intersect $\mathcal{W}^c$. Since for all points in the domain $\mathcal{Q}$, by following the flow of function f_1, have its limit set in $\mathcal{W}^c$ then one can argue that the domain of attraction for the hybrid stabilization algorithm presented here is the entire $\mathcal{Q}$ up to a zero measure set. In fact, experience shows that better trajectories (in the sense of minimum number of "back and forward" parallel parking motion for the vector field f_2) are obtained if the energy level φ is kept very high, i.e., in the torus level sets. There, the trajectories define quasi-periodic orbits that intersect the stable manifold $\mathcal{W}^s$ indefinitely.

For the simulations presented here, we consider the interesting parameters to be the *mean error position* defined by:

$$\text{mean error position} := \text{mean}_i \left[\| q_f^i - q^* \| \right], \tag{3.74}$$

where q_f^i is the final position reached on the ith run; and the *mean arc-length ratio*, which compares the performance against a fully actuated robot that can always follow a straight line to the goal. For continuous time the mean arc-lenght ratio is defined by:

$$\text{mean arc-length ratio} := \text{mean}_i \left[\frac{\int_0^{t_f^i} \| q'(q_0^i, t) \| \mathrm{d}t}{\| q_0^i - q^* \|} \right], \tag{3.75}$$

where t_f^i is the final time and $q'(q_0^i, t)$ the derivative of the trajectory starting at the initial position q_0^i for the ith run. For the 383 simulations run of a single beacon visual servoing problem, we obtained a mean error position of 4.3cm and a mean arc-length ratio of 4.1. These results seem promising since we utilized a 1st order approximation to the stable manifold.

Note that in Figure 3.7 the robot executes a parallel parking maneuver in the plane. Although it is well known that for the unicycle the parallel parking motion is required to move sideways, the trajectory obtained on the plane is a natural consequence of moving on a level set of the navigation function. Moreover, the navigation function enforces that the robot does not hit the obstacles, since doing that would require puncturing the level sets away from the goal.

3.5.4 The 3 beacon visual servoing problem

We now present simulations of the main visual servoing application described in this thesis, implemented in the RHex robot, where a vision sensor is designed to observe the position of three known artificial beacons that comprise a landmark. We defer the details of the camera map associated with the vision sensor to the entire Chapter 5 in order to give a better flow of the exposure of the developments on this thesis. We describe here the navigation function utilized and the results of the numerical simulations.

Consider the following potential function $\bar{\bar{\varphi}} : \mathcal{I} \to \mathbb{R}^+$,

$$\bar{\bar{\varphi}} := \frac{\left((\zeta_1 - \zeta_1^*)^2 + (\zeta_2 - \zeta_2^*)^2 + (\zeta_3 - \zeta_3^*)^2 \right)^k}{(\zeta_M - \zeta_1)(\zeta_1 - \zeta_2)(\zeta_2 - \zeta_3)(\zeta_3 - \zeta_m)(\zeta_1 - \zeta_3 - \zeta_d)}, \tag{3.76}$$

See Figure 5.2(a) in Chapter 5 for an illustration of the above parameters. For the previous potential function we consider the following:

- The vector $(\zeta_1^*, \zeta_2^*, \zeta_3^*)$ defines the goal in the image projection space $\mathcal{I}$, normally measured by taking a “snapshot” of the beacons at the desired position.

- k is a positive constant scalar shaping. In both simulations and experiments we take

$k = 1$. For more information on the shaping parameter see [Kod92].

- ζ_m and ζ_M are the field of view obstacles. These are computed based on the aperture of the camera's lenses.

- The denominator encodes the obstacles by "exploding" $\bar{\bar{\varphi}}$ when the 1st beacon reaches the left field of view boundary $(\zeta_M - \zeta_1)$; the 1st and 2nd beacon intercept $(\zeta_1 - \zeta_2)$ and so forth. Notice that since the beacon angles are ordered in the image projection space then the beacons 1 and 3 cannot intercept unless 1 and 2 or 2 and 3 intercept first, thus, allowing for a simpler denominator of the navigation function.

- The term $(\zeta_1 - \zeta_3 - \zeta_d)$ is introduced to limit the distance away from the set of beacons, where ζ_d is a positive scalar. Notice that the difference of the angles $\zeta_1 - \zeta_3$ will become smaller as the robot increases its distance from the beacons. The following formula gives a rough idea of how to approximately compute the parameter ζ_d given the distance between beacons 1 and 3, denoted by d_b, and the robot's maximum distance away from the beacons, denoted by d_{max}, both with units in meters:

$$\zeta_d = 2 \arcsin \left(\frac{d_b}{2d_{max}} \right) \tag{3.77}$$

The resulting navigation function follows the same "squashing" and change of coordinates as in equations (3.72) and (3.73). A representative numerical simulation is illustrated in Figure 3.8. Since the potential function $\bar{\bar{\varphi}}$, presented in equation (3.76), is defined in a convex set and has a unique critical point at q^*, all of its level sets are topological spheres. For the 368 simulations implemented we found the mean error position to be 5.3cm and the mean arc-lenght ratio to be 2.9, again using a 1st order stable manifold approximation. In Chapter 6 we present the experimental implementation of this system along with Table 6.1 that compares the performance of the simulations with the real-world experiments.

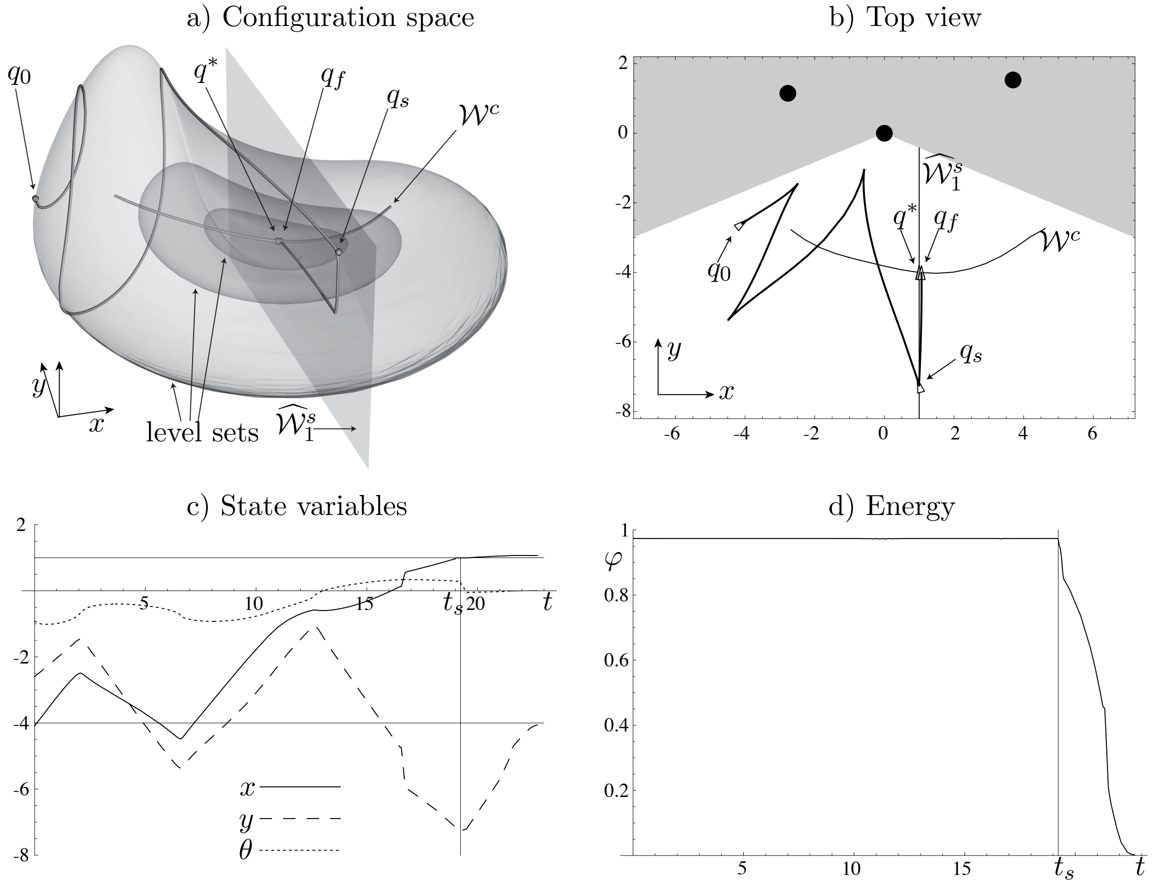


Figure 3.8: Simulation of the hybrid controller operating in the visible set of a three beacon landmark. The initial configuration is q_0, the controller switches at time t_s in position q_s and the final configuration is q_f. **a)** Configuration space plotted on $(x, y, 5\theta - 5\arctan(y/x))$ for readability purpose. **b)** Top view. The visual beacons are represented by the large black dots. The gray areas violate the visual constraints. **c)** and **d)** State variables and energy plots.

CHAPTER 4

Dynamic machines in general settings

In this chapter we discuss the “lift” of the algorithm proposed in the previous chapter for nonholonomically constrained dynamical systems. The resulting corollaries arise naturally from the ideas introduced in [Kod87a]. Let (4.1) and (4.2) be the mechanical system equations for fully actuated systems [Gol50] and nonholonomically constrained systems [BR92] with $q, u \in \mathbb{R}^n$ and $v \in \mathbb{R}^m, m < n$:

$$M(q)\ddot{q} + c(q, \dot{q}) = u \tag{4.1}$$

$$\begin{aligned} M(q)\ddot{q} + c(q, \dot{q}) &= A(q)^T \lambda + B(q) v \\ A(q)\dot{q} &= 0 \end{aligned} \tag{4.2}$$

Where M is the mass matrix, c the coriolis term, A and B represent the actuation constraints defined in Chapter 1, and λ is a vector of Lagrange multipliers. We start by recalling some notation and lemmas required for the subsequent proofs. Using the “stack-kronecker notation” [Mac46, Bel65, Kod87b] consider the following linear map:

$$\acute{M}_q : x \mapsto [x \otimes I]^T D_q M^S \tag{4.3}$$

and the skew-symmetric value operator:

$$J_q(x) := \acute{M}_q(x) - \acute{M}_q^T(x) \tag{4.4}$$

Lemma 4.1 ([Kod87a]). *For any curve, $q : \mathbb{R} \to \mathcal{Q}$, and any vector, $x \in T_{q(t_0)}\mathcal{Q}$,*

$$\dot{M}_q|_{t_0} x = \acute{M}_{q(t_0)}(x)\dot{q}|_{t_0} \tag{4.5}$$

Lemma 4.2 ([Kod87a]). *Given a lagrangian with kinetic energy, κ, with no potential forces present, and with an external torque or force actuating at every degree of freedom as specified* *by the vector, τ, the equations of motion may be written in the form:*

$$M(q)\ddot{q} + c(q, \dot{q}) = \tau \tag{4.6}$$

where

$$c(q, x) = C(q, x)x \tag{4.7}$$

and

$$C(q, x) := \frac{1}{2}\dot{M}(q_x) - \frac{1}{2}J_q(x) \tag{4.8}$$

Notice that the representation of the coriolis and centripetal forces in terms of the bilinear operator valued map C only coincide at $\dot{q}$ with the quadratic expression $c(q, \dot{q})$. In general they are not the same.

Corollary 4.3 ([Kod87a]). *For any motion $q : \mathbb{R} \rightarrow \mathcal{Q}$, and any tangent vector, $x \in T_{\mathcal{Q}}q(t)$,*

$$x^T \left[\frac{1}{2}\dot{M}(q) - C(q, \dot{q})\right] x \equiv 0 \tag{4.9}$$

Proof. From the previous lemma,

$$x^T \left[\frac{1}{2}\dot{M}(q) - C(q, \dot{q})\right] x = -\frac{1}{2}x^T J_q(\dot{q})x = 0 \tag{4.10}$$

□

4.1 Embedding the limit behavior of gradient dynamics

Controller $f_1(q) = -H(q) \cdot \nabla\varphi(q)$, introduced in Section 3.1, aims to reach a fixed point in the center manifold $\mathcal{W}^c$. In order to lift the controller into a 2nd order system, Theorem 4.4, concerning limit sets of gradient dynamics, is complemented with Corollary 4.5. Let the state variables p_1, p_2 represent $q, \dot{q}$ respectively and let $\mathcal{P} = \mathcal{TQ}$ be the tangent bundle of $\mathcal{Q}$ for system (4.1).

Theorem 4.4 (Koditschek [Kod87a]). *Let φ be a Morse function on $\mathcal{Q}$ which is exterior directed on the boundary $\partial\mathcal{Q}$, surpasses the value $\mu > 0$ on the boundary, and has a local minima at the points $\mathcal{G} := \{q_i\}_{i=1}^n \subset \mathcal{Q}$. Let $K_2 > 0$ denote some positive definite symmetric matrix. Consider the set of "bounded total energy" states*

$$\mathcal{P}^\mu := \left\{ \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} \in \mathcal{P} : \varphi(p_1) + \frac{1}{2}p_2^T M p_2 \leq \mu \right\} \tag{4.11}$$

Under the feedback algorithm

$$u := -K_2 p_2 - D\varphi^T(p_1) \tag{4.12}$$

$\mathcal{P}^\mu$ is a positive invariant set of the closed loop dynamical system within which all initial conditions excluding a set of measure zero take $\mathcal{G}$ as their positive limit set.

Let H be the nonholonomic projection matrix. Define $Q(q) := I - H(q)$ to be the nonholonomic converse projection matrix. Notice that $\ker(A) = \ker(Q)$ and therefore $Q(q)\dot{q} = 0$. Let $\mathcal{W}_0^c := \{q \in \mathcal{W}^c \wedge \dot{q} = 0\}$. As shown in Section 3.1, $\mathcal{W}^c$ is the center manifold of the system $\dot{q} = -H(q)\nabla\varphi(q)$. Rewriting equation (4.2) with a new input $v := B(q)^\dagger u$ we obtain:

$$\begin{aligned} M(q)\ddot{q} + c(q,\dot{q}) &= A(q)^T\lambda + H(q)u \\ A(q)\dot{q} &= 0 \end{aligned} \tag{4.13}$$

Corollary 4.5. *Let $K_2 = \bar{K}_2 H(q)$ with $\bar{K}_2 > 0$ denoting a positive definite symmetric matrix. Under the conditions of Theorem 4.4 all the initial conditions of the system (4.13), excluding a set of measure zero, take $\mathcal{W}_0^c$ as their positive limit set.*

Proof. Let $V = \varphi(q) + \frac{1}{2}\dot{q}^T M(q)\dot{q}$ be a Lyapunov function for (4.13). Then

$$\begin{aligned} \dot{V} &= D\varphi\dot{q} + \frac{1}{2}\dot{q}^T\dot{M}\dot{q} - \dot{q}^T\left(HK_2\dot{q} + HD\varphi^T + HC\dot{q}\right) + \underbrace{\dot{q}^T A^T\lambda}_{=0} \\ &= \underbrace{D\varphi Q\dot{q}}_{=0} + \underbrace{\dot{q}^T J_q \dot{q}}_{=0} - \dot{q}^T HK_2\dot{q} \\ &= -\dot{q}^T H^T \bar{K}_2 H\dot{q} \end{aligned} \tag{4.14}$$

$H^T\bar{K}_2 H$ is a semi-definite positive matrix. $\dot{V}$ is null when either $\dot{q} = 0$ or $H\dot{q} = 0$. Since $A(q)\dot{q} = 0 \Rightarrow H\dot{q} \neq 0$ then the largest invariant set is the interception of the previous sets with $\mathcal{W}_0^c$ resulting in $\mathcal{W}_0^c$. La Salle's theorem guarantees that (4.13) with input (4.12) takes $\mathcal{W}_0^c$ as the forward limit. □

4.2 Embedding of more general dynamics

We now seek to lift the controller $f_2(q)$ defined in Section 3.1 to a 2nd order system. We do so by adding once again a level regulator term, so that the reference dynamics attracts to a particular level set. First recall the embedding of general reference dynamics: let f be a reference vector field with Lyapunov function μ, and let $F(p) := p_2 - f(p_1)$. Consider the control algorithm,

$$u = -K_2 F - D\mu^T + MDfp_2 + Cf \tag{4.15}$$

which applied to the mechanical system (4.1) yields a closed loop form, $\dot{p} = h(p)$,

$$h(p) := \left[\begin{array}{c} p_2 \\ Dfp_2 - M^{-1}\left[K_2F + CF + D\mu^T\right] \end{array} \right] \tag{4.16}$$

Theorem 4.6 (Koditschek [Kod87a]). *If μ is a strict Lyapunov function for f on $\mathcal{Q}$, then*

$$V := \mu + \frac{1}{2}F^TMF \tag{4.17}$$

is a strict Lyapunov function for h on $\mathcal{P}$.

In system (4.13) the set of images of H for each point on $\mathcal{Q}$ is the tangent bundle of $\mathcal{Q}$. Therefore, since H is a projection operator then $\forall x \in \mathcal{P}$ we have $H(p_x).x = x$. Define $\bar{H} = M^{-1}HM$ and $\bar{Q} = M^{-1}QM$. For system (4.13) with input (4.15) the closed loop is written in the following way:

$$h(p) := \left[\begin{array}{c} p_2 \\ \bar{H}Dfp_2 - M^{-1}\left[HK_2F + HCF + HD\mu^T + QCp_2 - A^T\lambda\right] \end{array} \right] \tag{4.18}$$

Corollary 4.7. *Suppose $Hf = f$, i.e. the reference vector field respects the nonholonomic constraints. If μ is a strict Lyapunov function for f on $\mathcal{Q}$ for system (4.13), then*

$$V := \mu + \frac{1}{2}F^TMF \tag{4.19}$$

is a strict Lyapunov function for h on $\mathcal{P}$.

Proof. First note that $Q.f = 0$; $Q.p_2 = 0$; $A.f = 0$.

$$\begin{aligned} \dot{V} &= D\mu\; p_2 + \frac{1}{2}F^T\dot{M}F + F^TM\bar{Q}Dfp_2 + \\ &\quad -F^T\left(HK_2F + HCF + HD\mu^T + QCp_2\right) + F^TA^T\lambda \end{aligned} \tag{4.20}$$

$$\begin{aligned} &= D\mu Hf - F^THK_2F - \underbrace{f^TQ(MDf\dot{q} - Cf)}_{=0} + \underbrace{f^TA^T\lambda}_{=0} \\ &= D\mu f - F^TK_2F \end{aligned} \tag{4.21}$$

According to the hypothesis, the first term is negative except on the largest invariant set of $F = 0$. The second term is always negative except in $F^{-1}[0]$. The interception of the two results in the limit set of $\dot{q} = f(q)$. □

The previous result shows that as long as the reference dynamics respects the nonholonomic constraints we can apply Theorem 4.6 directly.

Notice that Corollary 4.7 also applies to controller $-H(q).\nabla\varphi(q)$. In general, such restrictive dynamics are not necessary for that controller, so using Corollary 4.5 provides a

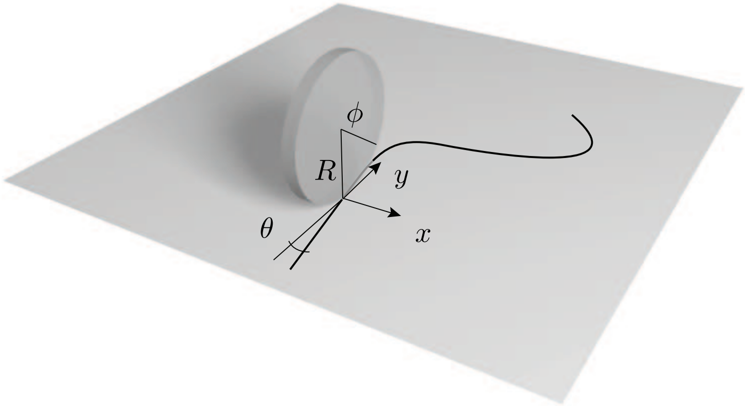


Figure 4.1: The vertical rolling disk.

better tool since we are only interested in the limit set.

4.3 Simulations

In this section we present simulation examples for the dynamic unicycle or vertical rolling disk depicted in Figure 4.1. Since we are interested in simulations in a dynamic setting we follow Bloch's vertical rolling disk [BKMM96], defined in the configuration space $\mathcal{Q} = \mathbb{R}^2 \times S^1 \times S^1 = \mathsf{SE}(2) \times S^1$ with coordinates $q = (x, x, \theta, \phi)$, mass m, inertias I, J, and disk radius R. The equations of motion for the vertical rolling disk are:

$$\begin{aligned} (mR^2 + I)\ddot{\phi} &= u_1 \\ J\ddot{\theta} &= u_2, \end{aligned} \tag{4.22}$$

with the constraint equations:

$$\begin{aligned} \dot{x} &= R\cos(x)\dot{\phi} \\ \dot{y} &= R\sin(x)\dot{\phi}. \end{aligned} \tag{4.23}$$

Differentiating (4.23) in time and replacing $\ddot{\phi}$ from (4.22), one obtains a complete set of equations of motion that verifies the nonholonomic constraints for initial conditions that also verify (4.23):

$$\begin{aligned} \ddot{x} &= -R\sin(\theta)\dot{\theta}\dot{\phi} + \frac{R\cos(\theta)}{mR^2 + I}u_1 \\ \ddot{y} &= R\cos(\theta)\dot{\theta}\dot{\phi} + \frac{R\sin(\theta)}{mR^2 + I}u_1 \end{aligned} \tag{4.24}$$

This system is now written in the form $M(q)\ddot{q} + c(q, \dot{q}) = B(q)u$ for which Corollaries 4.5, 4.7 apply directly. The $A(q)$ and $B(q)$ matrices are:

$$A := \begin{bmatrix} -\dfrac{I + mR^2}{R} & -\sin(\theta) \\ 0 & \cos(\theta) \\ 0 & 0 \\ \cos(\theta) & 0 \end{bmatrix}; \qquad B := \begin{bmatrix} \dfrac{R\cos(\theta)}{mR^2 + I} & 0 \\ \dfrac{R\sin(\theta)}{mR^2 + I} & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{4.25}$$

Although the algorithms presented in Chapter 3 are defined only in $\mathbb{R}^3$, by close inspection of A and B one realizes that, for this particular example, by choosing a $\mathbb{R}^4$ navigation function defined only by the first three parameters of q we will obtain the "same" controller as in $\mathbb{R}^3$. Let φ be a navigation function such that $\nabla\varphi = [\varphi_x, \varphi_y, \varphi_\theta, 0]^T$. Next compute the $\mathbb{R}^4$ cross product of A and $\nabla\varphi$:

$$\begin{aligned} \times(A, \nabla\varphi) &= \sum_{i,j,k,l=0}^{4} \epsilon_{ijkl} \nabla\varphi_j A_{k1} A_{l2} \hat{e}_i && (4.26) \\ &= \frac{-1}{\cos(\theta)} \begin{bmatrix} \varphi_\theta \cos(\theta) \\ \varphi_\theta \sin(\theta) \\ -\varphi_x \cos(\theta) - \varphi_y \sin(\theta) \\ -\dfrac{I + mR^2}{R}\theta \end{bmatrix}, && (4.27) \end{aligned}$$

where ϵ_{ijkl} denotes the permutation tensor, $\hat{e}_i$ are the canonical basis vectors and $A_{i,j}$ is the ith row, jth column of A. We now compare with the function f_2, defined in equation (3.10) for the $\mathbb{R}^3$ unicycle with $A_3 = [-\sin(\theta), \cos(\theta), 0]^T$ and $\nabla\varphi_3 := [\varphi_x, \varphi_y, \varphi_\theta]^T$:

$$\begin{aligned} f_2 &= A_3 \times \nabla\varphi_3 = && (4.28) \\ &= \begin{bmatrix} \varphi_\theta \cos(\theta) \\ \varphi_\theta \sin(\theta) \\ -\varphi_x \cos(\theta) - \varphi_y \sin(\theta) \end{bmatrix} && (4.29) \end{aligned}$$

The two previous computations produce, in effect, the same behavior for the variables x, y and θ. For each fixed coordinate ϕ in $\mathcal{Q} \subset \mathbb{R}^4$ one obtains a copy of the topology of $\mathsf{SE}(2)$. Therefore, from here on, although the configuration space is defined in $\mathbb{R}^4$, we will only be interested in x, y, and θ.

4.3.1 Navigation function

We revisit here the navigation problem addressed by Kantor [KR03] and Bhattacharya [BMCH04], described in Section 3.5.3. Let h be a change of coordinates from $\mathcal{Q} \subset \mathsf{SE}(2) \times S^1$ to double polar coordinates times S^1 that we denote by $\mathcal{P}$ with coordinates $p = [\eta, \mu, d, \phi]^T$:

$$\begin{bmatrix} \eta \\ \mu \\ d \\ \phi \end{bmatrix} = h(x, y, \theta, \phi) = \begin{bmatrix} \arctan(y/x) \\ \theta - \arctan(y/x) \\ \sqrt{x^2 + y^2} \\ \phi \end{bmatrix} \tag{4.30}$$

Again, the obstacles are introduced in the field of view so that the robot maintains a range of distances to the beacon and keeps facing it:

$$\mu_m < \mu < \mu_M; \qquad d_m < d < d_M \tag{4.31}$$

Consider the same potential function as the one described in equation (3.71):

$$\bar{\bar{\varphi}} := \frac{\left(2 - \cos(\eta - \eta^*) - \cos(\mu - \mu^*) + (d - d^*)^2\right)^k}{(1 - \cos(\mu - \mu_m))(1 - \cos(\mu - \mu_M))(d_M - d)(d - d_m)} \tag{4.32}$$

and its "squashed" navigation function version $\bar{\varphi} : \mathcal{P} \to [0, 1]$:

$$\bar{\varphi} := \frac{\bar{\bar{\varphi}}}{\kappa + \bar{\bar{\varphi}}} \tag{4.33}$$

The navigation function written in the $\mathcal{Q}$ coordinates is $\varphi(q) = \bar{\varphi} \circ h(q)$ and its derivative:

$$\nabla\varphi(q) = Dh^T(q) \cdot \nabla\bar{\varphi} \circ h(q) \tag{4.34}$$

We choose to present the Kantor-Bhattacharya example as the canonical illustration of our ideas due to the interesting topology of the configuration space. Since $\mathcal{Q}$ is not simply connected, the level sets of φ change from topological spheres close to the goal q^* to topological tori close to the boundary of $\mathcal{Q}$. Initial conditions starting in the tori will generate quasi-periodic orbits when f_2 is used. In the dynamical setting this provides a good example of the applicability of Corollary 4.7, resulting in the generation of reference dynamics that attract to a particular level set.

4.3.2 Kinematic rolling disk

We first simulate the previously described system in a kinematic setting by solving the system:

$$\dot{q} = B(q)u, \tag{4.35}$$

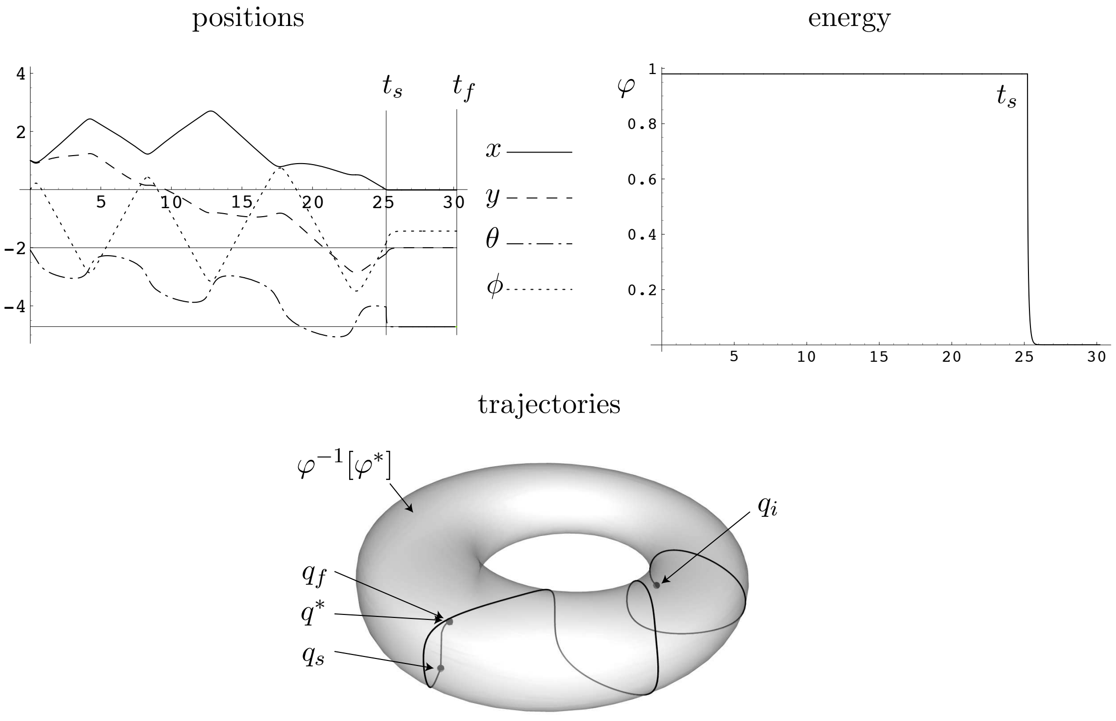


Figure 4.2: Kinematic simulation of the vertical rolling disk.

and using the control functions f_1, f_2 defined in Section 3.1:

$$\begin{aligned} u_1(q) &:= f_1(q) = -H\nabla\varphi && (4.36) \\ u_2(q) &:= \sigma(q) f_2(q) = \times(A, \nabla\varphi)\sigma && (4.37) \end{aligned}$$

Figure 4.2 illustrates the resulting simulation where the initial condition is $q_0 = [1, 1, -\dfrac{3\pi}{4}, 0]^T$, the body parameters are $I = J = m = R = 1$, the obstacles are $\mu_m = -\frac{\pi}{4}; \mu_M = \frac{\pi}{4}$; $d_m = 1$; $d_M = 3$, and $\sigma(q) = x$. The manifold $\widehat{\mathcal{W}}_1^s = \{q \in \mathcal{Q} : x = 0\}$ is a good local approximation for the stable manifold $\mathcal{W}^s$ of the system $\dot{q} = f_1(q)$ at the specific goal $q_0 = [0, -2, \pi/2, 0]^T$. One can observe that from the initial time to t_s the controller f_2 keeps the energy constant while moving exactly in the level set $\varphi^{-1}[\varphi^*]$, with $\varphi^* = 0.98$. At time t_s we switch to controller f_1 and the resulting final position is very close to the goal. Looking at ϕ in the "positions" graphic one observes that the robot does a back and forward motion, necessary to the parallel park maneuver. This comes as a natural consequence of moving in the surface of the torus shown in the "trajectories" plot.

4.3.3 Dynamic rolling disk

For the dynamic setting we solve the system defined by equations (4.22) and (4.24):

$$M(q)\ddot{q} + c(q,\dot{q}) = H(q)u, \tag{4.38}$$

with control functions (4.12) and (4.15):

$$\begin{aligned} u_1(q,\dot{q}) &:= -K_2\dot{q} - \nabla\varphi & (4.39)\\ u_2(q,\dot{q}) &:= -K_2F - D\mu^T + MDf_2\dot{q} + Cf_2, & (4.40) \end{aligned}$$

where $F(q,\dot{q}) := \dot{q} - f_2(q)$ and $\mu := \alpha(\varphi - \varphi^*)^2$.

The stable manifold for the full system (4.38) with input (4.39) can be found using the same process as in Section 3.4.3 by, again, defining the vector field $p = (p_1, p_2) = (q, \dot{q})$ and writting equation (4.38) as

$$\begin{aligned} \dot{p}_2 &= M^{-1}(p_1)\left[H(p_1)u(p_1,p_2) - c(p_1,p_2)\right] & (4.41)\\ \dot{p}_1 &= p_2 & (4.42) \end{aligned}$$

or in compact form $\dot{p} = F(p)$. However, we find that for this particular set of simulations the stable manifold approximation defined by $\widehat{\mathcal{W}}_1^{sk} := \left\{q \in \widehat{\mathcal{W}}_1^s, \dot{q} = 0\right\} \subset \mathcal{TQ}$ gives a good approximation for the true stable manifold (we assume that the approach to the true stable manifold is acomplished at low speed so that the approximation defined for the quasi-static system remains valid).

The first simulation, depicted in Figure 4.3, uses a high gain $\alpha = 5000$ in the function μ to track the level set as close as possible while the controller f_2 is in use. This results in a good tracking but very jerky steering motion, visible in the first part of the "velocities" and "trajectories" plots. The damping matrix K_2 is set to the identity matrix, resulting in low damping, as observed in the intervals $[t_s, t_f]$ of the "positions" and "velocities" plots.

For the second simulation in the dynamic setting, depicted in Figure 4.4, the parameter $\alpha = 250$ provokes a less accurate tracking of the desired level set φ^*, when using f_2, as one can observe in the "energy" and "trajectories" plots. However, the resulting motion is smoother then the previous simulation. For the controller f_1, the damping matrix $K_2 = 10I$ slows down the approach to the desired goal, eliminating any oscillations, as seen in the "energy" plot.

The damping matrix K_2 and the Lyapunov function μ are the design parameters for the control of equation (4.38).

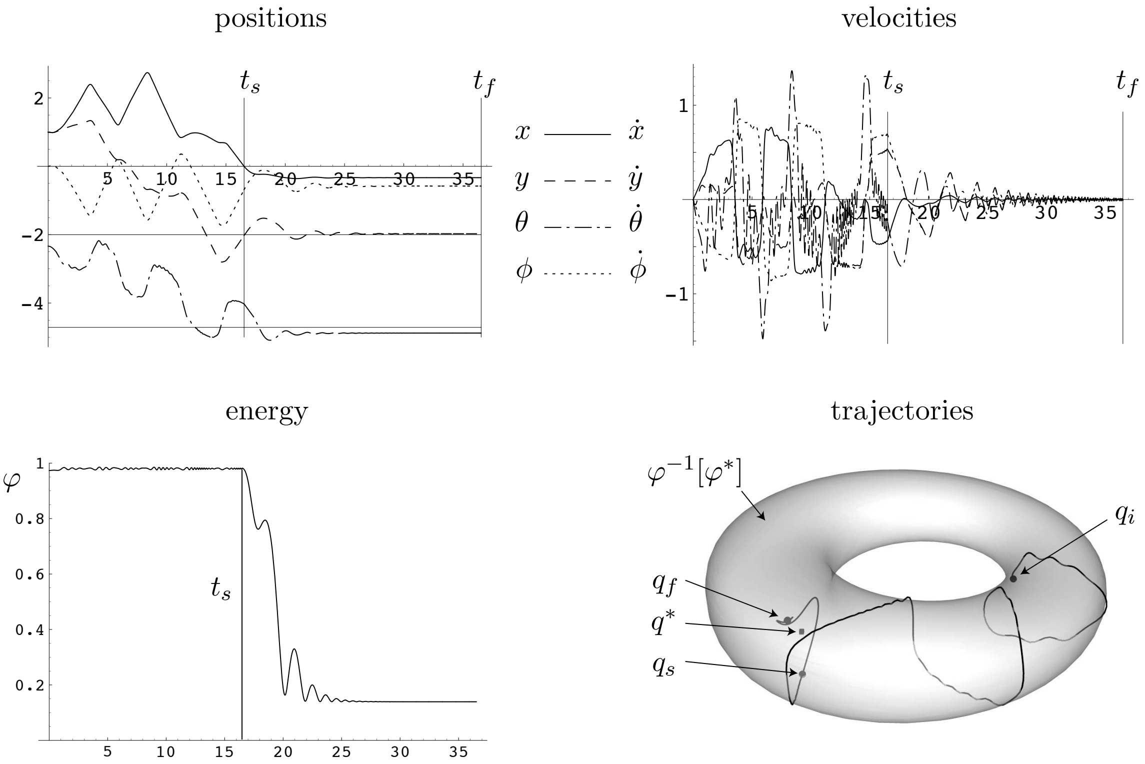
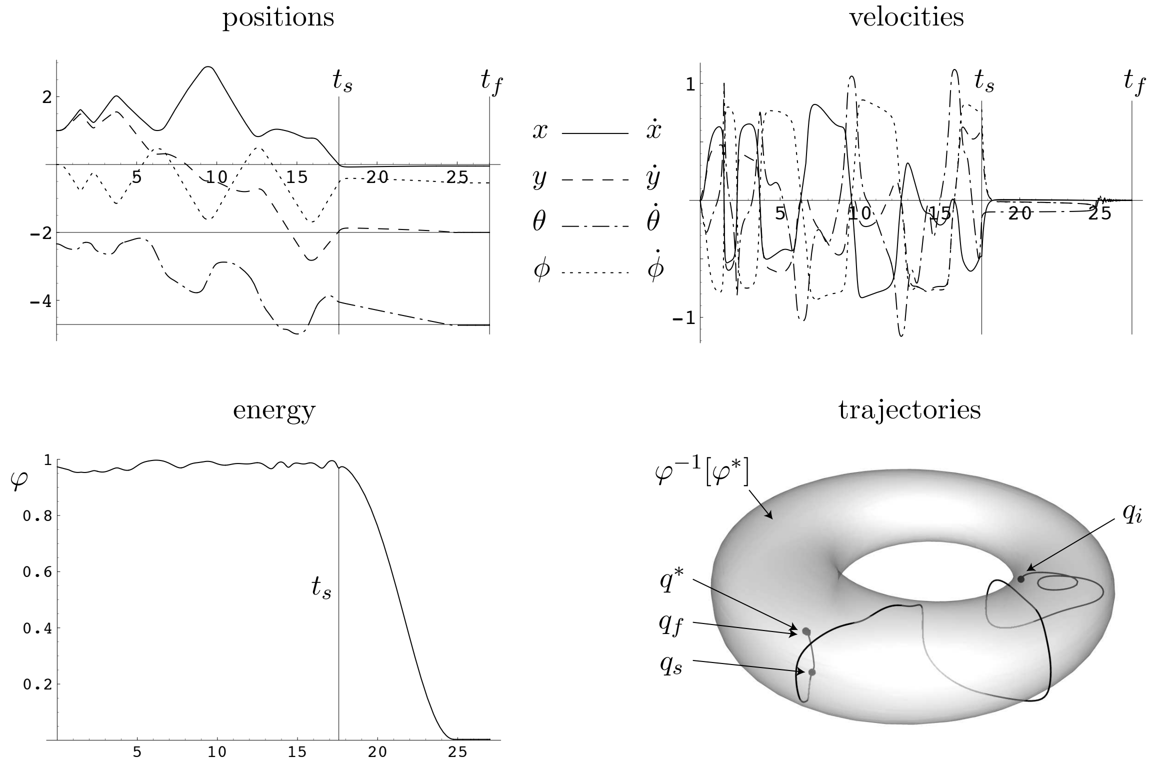


Figure 4.3: Dynamic simulation of the vertical rolling disk.



positions

velocities

energy

trajectories

Figure 4.4: Dynamic simulation of the vertical rolling disk.

CHAPTER 5

Visual Servoing for fully actuated robots

We describe in this chapter a visual sensor designed to observe the position of three known artificial beacons that comprise a landmark. In Chapter 6 these ideas are applied to the hexapedal robot RHex considered as operating in the (three degree of freedom) plane.

We use the term "beacon" to denote any perceptually reliable marker that offers fixed bearing information. Our beacons are "artificial" — brightly colored objects that easily stand out against natural outdoor settings, depicted in figure 5.1 — because we seek to avoid the well known problems of early vision that lie outside the intended scope of present work. We use the term "landmark" to denote the composition of three beacons into a source of sensory information sufficient to extract and regulate full relative pose on the plane. This point of view represents an adaptation and slight generalization of the fixed camera, moving beacon visual servoing algorithms introduced by Cowan *et al* [CWK02]. In contrast, we address the "inside out" version of that problem arising from the task of registering a mobile robot vehicle relative to some fixed landmark in the visual field (we assume that the camera is fixed to the robot's frame). The resulting camera map incorporates (a transformed copy of) the full relative pose. Its gradient will be used to generate a servo controller that forces convergence to (some arbitrarily small specified neighborhood of) any desired visible pose along "safe" transients guaranteed to maintain the view along the way.

We find that it is sufficient to identify a beacon with the location of its centroid projected onto the plane. Given three such centroids, without any loss of generality, we define their composed landmark parameter space $\mathcal{B} \subset (\mathbb{R}^+)^2 \times S^1$ by fixing the world frame so that the second beacon is at the origin and the remaining beacons lie along lines going through the origin that define congruent angles (see figure 5.2(a)):

$$\mathcal{B} := \{(\rho_1, \rho_2, \alpha) \in (\mathbb{R}^+)^2 \times S^1 \mid \rho_1 > 0, \rho_2 > 0, 0 \leq \alpha < \pi\} \tag{5.1}$$

The coordinates of each beacon b_i in the world frame are:

$$\begin{bmatrix} b_1 & b_2 & b_3 \end{bmatrix} = \begin{bmatrix} \rho_1 R_\alpha \hat{e}_2 & \mathbf{0} & \rho_2 R_\alpha^T \hat{e}_2 \end{bmatrix}, \tag{5.2}$$

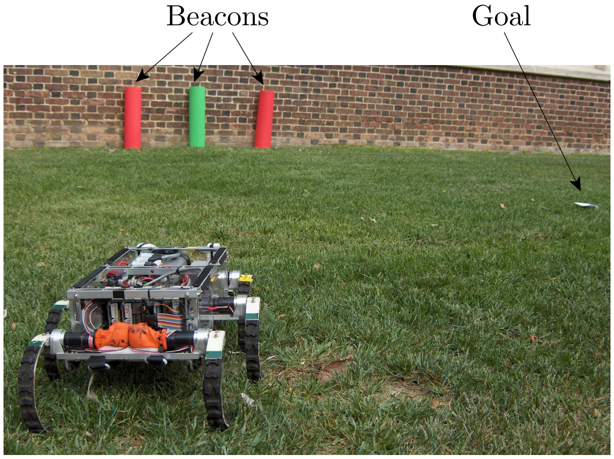


Figure 5.1: The visual servoing problem: the robot RHex aims to reach the goal navigating by use of the colored beacons for visual cues.

where $R_\alpha = [\cos(\alpha)\ -\sin(\alpha); \sin(\alpha)\ \cos(\alpha)]$ is the standard 2×2 rotation matrix and $\hat{e}_2$ is the canonical base vector $[0\ 1]^T$.

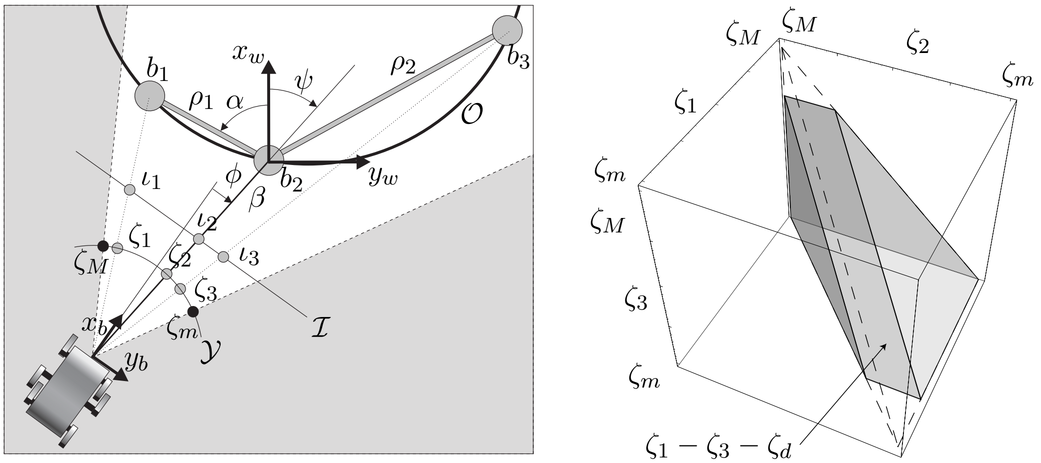
5.1 Camera map

Here we define the camera map to be the transformation that relates the pose (position and orientation) of the robot in the world frame ($\mathsf{SE}(2)$) to the pinhole projection of the beacons in the camera's image plane.

For convenience, we treat the camera image plane as (a subset of) the unit sphere, S^2, and drop the azimuthal component, thereby projecting all pinhole camera readings onto the great circle, $S^1 \subset S^2$, corresponding to bearing in the horizontal plane. In this manner, a beacon's pinhole image is parameterized by the angle of the ray that connects it to the camera center when projected onto the horizontal plane. We denote by $\mathcal{Y} \subset T^3$ this image projection space — the triple of angles of each of the beacons in a landmark. Note that, although a physical camera has a flat image plane, that we denote by $\mathcal{I}$, we prefer to work with a ray's angle computed by the transformation $\zeta_i = \arctan(\iota_i) + \pi/2$, where ζ_i is the ith angle and ι_i is the coordinate measured by the camera in meters (after pre-processing using a lens calibration model), as illustrated in figure 5.2(a).

Because subsequent computations involving robot pose associated with the camera map are most easily expressed in polar coordinates, we find it expedient to introduce a new space[1], $\mathcal{P} \subset T^2 \times \mathbb{R}^+$, diffeomorphic to the robot configuration space with coordinates $w = (\phi, \psi, \beta)$ (see figure 5.2(a)), where T^2 is the 2-dimensional torus. The motivation to introduce such a coordinate system arises from the fact that in $\mathcal{P}$ the set of self-occlusions

[1] The introduction of the intermediate space $\mathcal{P}$ distinguishes the present construction [LK03] from the one implemented in [CWK02].



(a) (b)

Figure 5.2: Simply connected configuration space introduced by Cowan *et al.* **a)** The beacons are represented by the gray circles named (b_1, b_2, b_3). **b)** Configuration space plotted in the Image projection space.

appears as a literal (2 dimensional) torus, providing significant geometrical insight into the self-occlusion problem.

Having adopted a representation for the beacon configuration and the robot configuration space it is now necessary to determine for a given beacon, the set of robot configurations for which occlusion-free servoing can be accomplished. Define the *facing set* $\mathcal{F}$ as the set of configurations for which the robot lies "in front" of the set of beacons, i.e. the beacons appear to face the robot sensor. Intuitively the beacons must keep a certain order in the camera projection line $\mathcal{I}$. Define the function $\varrho_i : \mathcal{P} \to \mathbb{R}^2$ that returns a vector that goes through beacon b_i for a given configuration $w = (\phi, \psi, \beta)$.

$$\varrho_i(w) \quad := \quad R_\phi R_\psi b_i + \beta R_\phi \hat{e}_2 \tag{5.3}$$

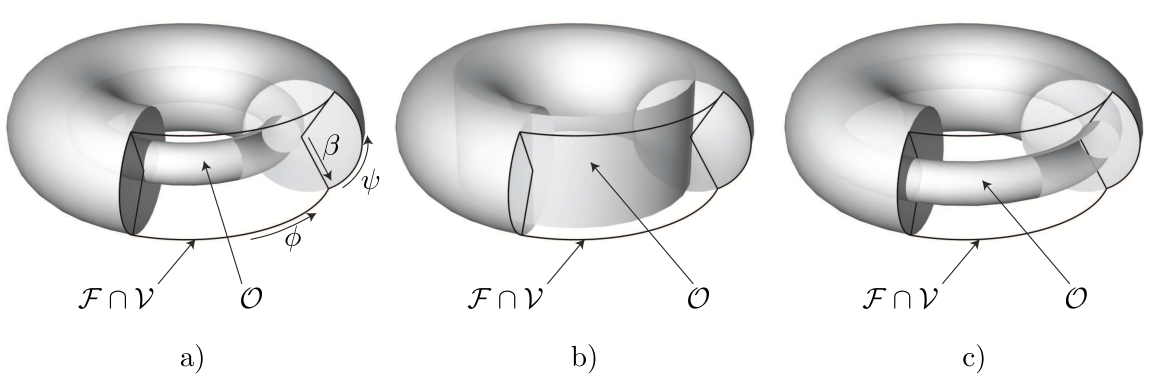
The facing set is then defined by (5.4) where J is a skew symmetric matrix:

$$\mathcal{F} \quad := \quad \{w \in \mathcal{P} \mid \varrho_i(w)^T J \varrho_j(w) > 0; i < j\} \tag{5.4}$$

Define the *visible set* $\mathcal{V}$ as the set of configurations for which the beacons are in the field of view (FOV) of the camera sensor, where ζ_{min}, ζ_{max} are the FOV camera parameters illustrated in Figure 5.2(a) and function arctan is assumed to take into account which quadrant its argument is in.

$$\mathcal{V} := \{w \in \mathcal{P} \mid \zeta_{min} < \arctan(\varrho_i(w)) < \zeta_{max}, i = 1, 2, 3\} \tag{5.5}$$

The previous sets arise from geometrical insight, necessary for the vision implementation, but are not sufficient to fully characterize the set of configurations for which pose



β
ψ
ϕ
$\mathcal{F} \cap \mathcal{V}$
$\mathcal{O}$
a)
$\mathcal{F} \cap \mathcal{V}$
$\mathcal{O}$
b)
$\mathcal{F} \cap \mathcal{V}$
$\mathcal{O}$
c)

Figure 5.3: Illustration of the intermediate space $\mathcal{P}$. The thick black lines represent the intersection of the visible set with the facing set. **a)** convex beacon configuration. **b)** linear beacon configuration. **c)** concave beacon configuration. For the concave configuration the obstacle set $\mathcal{O}$ disconnects the workspace $\mathcal{W}$.

computation can be accomplished. In fact, as shown later, the camera map may not always be injective in $\mathcal{F} \cap \mathcal{V}$. This is due to the generalization of [CLK00] by allowing any beacon configuration. It is shown here that the injectivity is lost at worst on the zero set of the function Θ (a factor in the determinant of the jacobian of the camera map, described next) in which set the inverse image can have cardinality 2. We then introduce the *obstacle set* $\mathcal{O}$ using the function Θ:

$$\begin{aligned} \Theta(w) \quad &: \quad \mathcal{B} \times \mathcal{P} \to \mathbb{R} \\ &\quad\quad w \mapsto \rho_1 \sin(\alpha - \psi) + \rho_2 \sin(\alpha + \psi) + \beta \sin(2\alpha) \end{aligned}$$

$$\mathcal{O} \quad := \quad \{v \in \mathcal{B}, w \in \mathcal{P} \mid \Theta(w) = 0\}$$

If the polar configuration space is understood topologically as a thickened torus, then the obstacle set will in general be a thin torus that disconnects $\mathcal{P}$. If the three beacons are collinear then $\mathcal{O}$ becomes a cylinder. Figure 5.3 illustrates the solid torus sliced by the FOV of the camera sensor. This exemplifies the dependence of the pose computation afforded by a given landmark upon the particular physical configuration of its constituent beacons. For convenience, in this work, we maintain a linear beacon configuration since it proves to result in the largest robot workspace.

Define the workspace $\mathcal{W} \subset \mathcal{P}$ by

$$\mathcal{W} := (\mathcal{F} \cap \mathcal{V}) - \mathcal{O}, \tag{5.6}$$

To reconstruct the pose of the robot in the world frame, with coordinates (x_w, y_w, θ_w), a composition of changes of coordinates is implemented. We denote the change of coordinates from the intermediate space to the image projection space by *intermediate camera map*

$c^B : \mathcal{P} \to \mathcal{Y}$,

$$c^B(w) := \begin{bmatrix} \arctan(\varrho_1(w)) \\ \arctan(\varrho_2(w)) \\ \arctan(\varrho_3(w)) \end{bmatrix}, \tag{5.7}$$

Proposition 5.1. *The intermediate camera map c^B is locally a diffeomorphism.*

Proof. It is easy to see that c^B is locally continuous and differentiable. It is enough to show that $|D_w c^B| \neq 0$ in $\mathcal{W}$:

$$D_w c^B = \begin{bmatrix} \nabla \arctan(\varrho_1)^T . D_w \varrho_1 \\ \nabla \arctan(\varrho_2)^T . D_w \varrho_2 \\ \nabla \arctan(\varrho_3)^T . D_w \varrho_3 \end{bmatrix} \tag{5.8}$$

with

$$\begin{aligned} \nabla \arctan(\varrho)^T &= -\frac{\varrho^T J}{\varrho^T \varrho} && (5.9) \\ D_w \varrho_i &= [R_\phi J R_\psi b_i + \beta R_\phi J \hat{e}_2 \quad R_\phi R_\psi J b_i \quad R_\phi \hat{e}_2] && (5.10) \end{aligned}$$

then:

$$\begin{aligned} \nabla \arctan(\varrho_i)^T . D_w \varrho_i &= \begin{bmatrix} -\frac{\varrho_i^T J J \varrho_i}{\varrho_i^T \varrho_i} & \frac{\varrho_i^T R_\phi R_\psi b_i}{\varrho_i^T \varrho_i} & \frac{\varrho_i^T R_\phi \hat{e}_1}{\varrho_i^T \varrho_i} \end{bmatrix} \\ &= \begin{bmatrix} 1 & \frac{\beta \hat{e}_2^T R_\psi b_i + b_i^T b_i}{\varrho_i^T \varrho_i} & \frac{b_i^T R_\psi^T \hat{e}_1}{\varrho_i^T \varrho_i} \end{bmatrix} && (5.11) \end{aligned}$$

using equation (5.11):

$$\begin{aligned} D_w c^B &= \begin{bmatrix} 1 & \frac{\rho_1 \beta \hat{e}_2^T R_\psi R_\alpha \hat{e}_2 + \rho_1^2}{\varrho_1^T \varrho_1} & \frac{\rho_1 \hat{e}_2^T R_\alpha^T R_\psi^T \hat{e}_1}{\varrho_1^T \varrho_1} \\ 1 & \frac{\rho_2 \beta \hat{e}_2^T R_\psi R_\alpha^T \hat{e}_2 + \rho_2^2}{\varrho_2^T \varrho_2} & \frac{\rho_2 \hat{e}_2^T R_\alpha R_\psi^T \hat{e}_1}{\varrho_2^T \varrho_2} \\ 1 & 0 & 0 \end{bmatrix} && (5.12) \\ &= \begin{bmatrix} \|\varrho_1\|^{-2} & 0 & 0 \\ 0 & \|\varrho_2\|^{-2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \|\varrho_1\|^2 & \rho_1 \beta \cos(\psi + \alpha) + \rho_1^2 & -\rho_1 \sin(\psi + \alpha) \\ \|\varrho_2\|^2 & \rho_2 \beta \cos(\psi - \alpha) + \rho_2^2 & -\rho_2 \sin(\psi - \alpha) \\ 1 & 0 & 0 \end{bmatrix} \end{aligned}$$

therefore:

$$|D_w c^B| = \|\varrho_1\|^{-2} \|\varrho_2\|^{-2} \det \begin{bmatrix} \rho_1 \beta \cos(\psi + \alpha) + \rho_1^2 & -\rho_1 \sin(\psi + \alpha) \\ \rho_2 \beta \cos(\psi - \alpha) + \rho_2^2 & -\rho_2 \sin(\psi - \alpha) \end{bmatrix} \tag{5.13}$$

$$
\begin{aligned}
&= \rho_1\rho_2\|\varrho_1\|^{-2}\|\varrho_2\|^{-2}\left(\beta\sin(2\alpha)+\rho 1\sin(\alpha-\psi)+\rho_2\sin(\alpha+\psi)\right) && (5.14)\\
&= \frac{\rho_1\rho_2\Theta}{\|\varrho_1\|^2\|\varrho_2\|^2} && (5.15)
\end{aligned}
$$

□

Proposition 5.2. *The cardinality of the inverse map $c^{B^{-1}}$ is unique.*

Proof. Consider the projected points $w_1 = [\phi_1\ \psi_1\ \beta_1]$ and $w_2 = [\phi_2\ \psi_2\ \beta_2]$. Then:

$$
\left(R_{\phi_1}R_{\psi_1}b_i+\beta_1R_{\phi_1}\hat{e}_2\right)^T J\left(R_{\phi_2}R_{\psi_2}b_i+\beta_2R_{\phi_2}\hat{e}_2\right)=0,\quad i=1,...,3 \tag{5.16}
$$

which is equivalent to:

$$
\begin{cases}
\left(\rho_1R_{\phi_1}R_{\psi_1}R_\alpha\hat{e}_2+\beta_1R_{\phi_1}\hat{e}_2\right)^T J\left(\rho_1R_{\phi_2}R_{\psi_2}R_\alpha\hat{e}_2+\beta_2R_{\phi_2}\hat{e}_2\right)=0\\
\left(\beta_1R_{\phi_1}\hat{e}_2\right)^T J\left(\beta_2R_{\phi_2}\hat{e}_2\right)=0\\
\left(\rho_2R_{\phi_1}R_{\psi_1}R_\alpha^T\hat{e}_2+\beta_1R_{\phi_1}\hat{e}_2\right)^T J\left(\rho_2R_{\phi_2}R_{\psi_2}R_\alpha^T\hat{e}_2+\beta_2R_{\phi_2}\hat{e}_2\right)=0
\end{cases} \tag{5.17}
$$

Using the second equation:

$$
\begin{aligned}
\left(\beta_1R_{\phi_1}\hat{e}_2\right)^T J\left(\beta_2R_{\phi_2}\hat{e}_2\right)=0 \quad&\Leftrightarrow\quad \beta_1\beta_2\sin(\phi_1-\phi_2)=0\\
&\Rightarrow\quad \phi_1=\phi_2+k\pi, k\in\mathbb{N} && (5.18)
\end{aligned}
$$

Then,

$$
\begin{cases}
\rho_1(\beta_2\sin(\psi_1+\alpha)-\beta_1\sin(\psi_2+\alpha)+\rho_1\sin(\psi_1-\psi_2))=0\\
\rho_2(\beta_2\sin(\psi_1-\alpha)-\beta_1\sin(\psi_2-\alpha)+\rho_2\sin(\psi_1-\psi_2))=0
\end{cases} \tag{5.19}
$$

Let $\beta_i\sin(2\alpha)+\rho_1\sin(\alpha-\psi_i)+\rho_2\sin(\alpha+\psi_i) = \omega_i \neq 0$. Substituting in the previous equation we obtain:

$$
\begin{cases}
\omega_2\sin(\psi_1+\alpha)-\omega_1\sin(\psi_2+\alpha)=0\\
\omega_2\sin(\psi_1-\alpha)-\omega_1\sin(\psi_2-\alpha)=0
\end{cases} \quad ,\sin(2\alpha)\neq 0 \tag{5.20}
$$

Eliminating ω_1 and ω_2 from the previous equantion results in:

$$
\sin(2\alpha)\sin(\psi_1-\psi_2)=0,\omega_1\neq 0,\omega_2\neq 0,\sin(2\alpha)\neq 0
$$

$$
\Rightarrow \psi_1=\psi_2+k\pi,\quad k\in\mathbb{N} \tag{5.21}
$$

Going back to equation (5.19) with $\psi_1 = \psi_2$ we obtain:

$$\left\{ \begin{array}{l} (\beta_2 - \beta_1)\rho_1 \sin(\alpha + \psi) = 0 \\ (\beta_2 - \beta_1)\rho_2 \sin(\alpha - \psi) = 0 \end{array} \right. \Rightarrow \beta_1 = \beta_2 \tag{5.22}$$

□

Lemma 5.3. c^B *is a smooth and smoothly invertible map from the workspace* $\mathcal{W}$ *into* $\mathcal{Y}$.

Proof. Propositions 5.1 and 5.2. □

The use of the intermediate space $\mathcal{P}$ provides a simple closed form expression for $(c^B)^{-1}$, the camera map inverse valid in $c^B(\mathcal{P}-\mathcal{O})$. To find its inverse the same constructive method is used as in [Cow01]. Knowing that $\varrho^T J \varrho = 0$ we have:

$$\varrho_i^T J R_\phi (\rho_i R_{\alpha_i} R_\psi + \beta I)\hat{e}_2 = 0 \tag{5.23}$$

in particular:

$$\alpha_1 = -\alpha_3 = \alpha; \quad \rho_2 = 0 \tag{5.24}$$

$$\begin{aligned} & \beta \varrho_2^T J R_\phi \hat{e}_2 = 0 \Leftrightarrow \\ \Leftrightarrow \quad & R_\phi = \delta_1 \left[\begin{array}{cc} J\varrho_2 & -\varrho_2 \end{array} \right] \end{aligned} \tag{5.25}$$

The constant $\delta_1 = \pm 1$ is chosen so that $-\frac{\pi}{2} < \phi < \frac{\pi}{2}$ resulting in:

$$\phi = \zeta_2 + \frac{\pi}{2} \tag{5.26}$$

Let $Y' = \left[\begin{array}{ccc} \rho_1 \varrho_1^T R_{\alpha_1} & \mathbf{0} & \rho_3 \varrho_3^T R_{\alpha_3} \end{array} \right]$ and $Y = \left[\begin{array}{ccc} \varrho_1 & \varrho_2 & \varrho_3 \end{array} \right]$. Then:

$$\begin{aligned} & \rho_i \varrho_i^T R_{\alpha_i} J R_\psi R_\phi \hat{e}_2 + \beta \varrho_i^T J R_\phi \hat{e}_2 = 0 \\ \Leftrightarrow \quad & \left[\begin{array}{cc} Y'^T & Y^T \end{array} \right] \left[\begin{array}{c} J R_\phi R_\psi \hat{e}_2 \\ \beta J R_\phi \hat{e}_2 \end{array} \right] = 0 \end{aligned} \tag{5.27}$$

Let $Y_\perp^\dagger$ be the orthogonal complement of the subspace generated by the lines of $Y^\dagger$,i.e. $Y_\perp^\dagger$ lives in the null space of $Y^\dagger$, with $Y^\dagger = (YY^T)^{-1}Y$ the pseudo-inverse of Y^T. Since $\left[\begin{array}{cc} Y^{\dagger^T} & Y_\perp^{\dagger^T} \end{array} \right]$ is full rank then the previous expression is equivalent to:

$$\begin{aligned} & \left[\begin{array}{c} Y^\dagger \\ Y_\perp^\dagger \end{array} \right] \left[\begin{array}{cc} Y'^T & Y^T \end{array} \right] \left[\begin{array}{c} J R_\phi R_\psi \hat{e}_2 \\ \beta J R_\phi \hat{e}_2 \end{array} \right] = 0 \\ \Leftrightarrow \quad & \left[\begin{array}{cc} Y^\dagger Y'^T & I \\ Y_\perp^\dagger Y'^T & 0 \end{array} \right] \left[\begin{array}{c} J R_\phi R_\psi \hat{e}_2 \\ \beta J R_\phi \hat{e}_2 \end{array} \right] = 0 \end{aligned} \tag{5.28}$$

Solving for R_ψ we get:

$$\begin{aligned} x &= \delta_2 \frac{Y' {Y_\perp^\dagger}^T}{\|Y' \varrho_\perp\|} & (5.29) \\ R_\psi &= R_\phi^T \left[\begin{array}{cc} Jx & -x \end{array} \right] & (5.30) \end{aligned}$$

Simplifying we obtain ψ:

$$\psi = \arctan(\delta_2 R_\phi^T J Y' {Y_\perp^\dagger}^T) \qquad (5.31)$$

Again $\delta_2 = \pm 1$ is chosen so that $-\frac{\pi}{2} < \psi < \frac{\pi}{2}$. Finally solving for β in (5.28) we get:

$$\begin{aligned} & Y^\dagger Y'^T J \left[\begin{array}{cc} Jx & -x \end{array} \right] \hat{e}_2 + \beta J R_\phi \hat{e}_2 = 0 \\ \Leftrightarrow \quad & \beta \| J R_\phi \hat{e}_2 \| = \| Y^\dagger Y'^T J \left[\begin{array}{cc} Jx & -x \end{array} \right] \hat{e}_2 \| \\ \Leftrightarrow \quad & \beta = \frac{\| Y^\dagger Y'^T J Y' {Y_\perp^\dagger}^T \|}{\| Y' {Y_\perp^\dagger}^T \|} & (5.32) \end{aligned}$$

In the obstacle set $\mathcal{O}$ we have $\|Y' {Y_\perp^\dagger}^T\| = 0$. In summary, the inverse camera map results in the following expressions:

$$\begin{aligned} \phi &= \zeta_2 + \frac{\pi}{2} \\ \psi &= \arctan(\delta R_\phi^T J Y' {Y_\perp^\dagger}^T) & (5.33) \\ \beta &= \frac{\| Y^\dagger Y'^T J Y' {Y_\perp^\dagger}^T \|}{\| Y' {Y_\perp^\dagger}^T \|} \end{aligned}$$

Numerical simulations in [LK03] suggest that in general it is possible to accomplish global convergent occlusion-free navigation even in the presence of a disconnecting obstacle set $\mathcal{O}$. In particular, for a concave beacon configuration it is expected that in a small neighborhood of $\mathcal{O}$ the camera map jacobian is small, potentially introducing large errors in the pose computation. Figure 5.4 illustrates the pose computation error by adding a random noise vector with gaussian distribution to the computation of the inverse camera map, hence simulating noise from a camera. One can notice that the error increases with distance from the beacons and the proximity to $\mathcal{O}$. Figure 5.5 illustrates the deformation obtained from "pushing" spheres centered at different initial conditions in $\mathcal{I}$ through the inverse camera map, giving a sense of how a small perturbation in $\mathcal{I}$ may result in a very large uncertainty region in $\mathsf{SE}(2)$.

The final camera map from world coordinates $\mathcal{Q} \subset \mathsf{SE}(2)$ to image projection space $\mathcal{Y}$

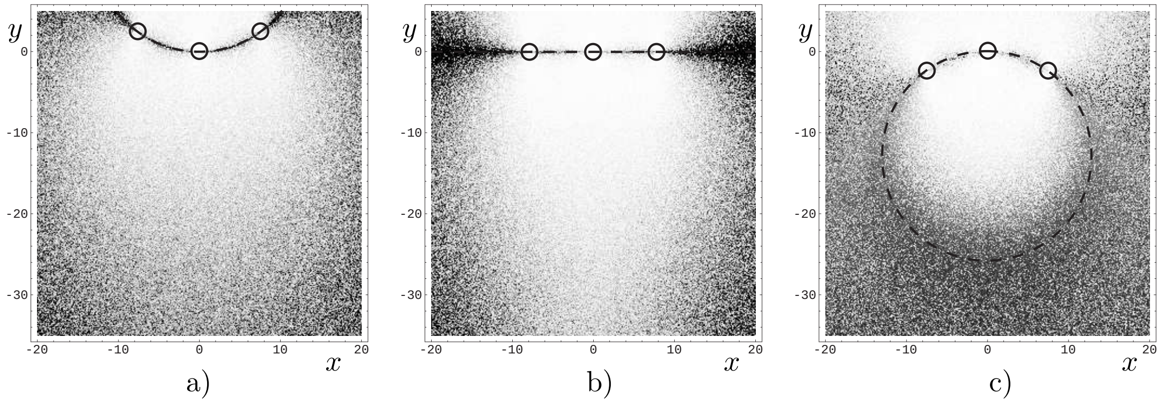


Figure 5.4: Pose computation error sampled distribution.

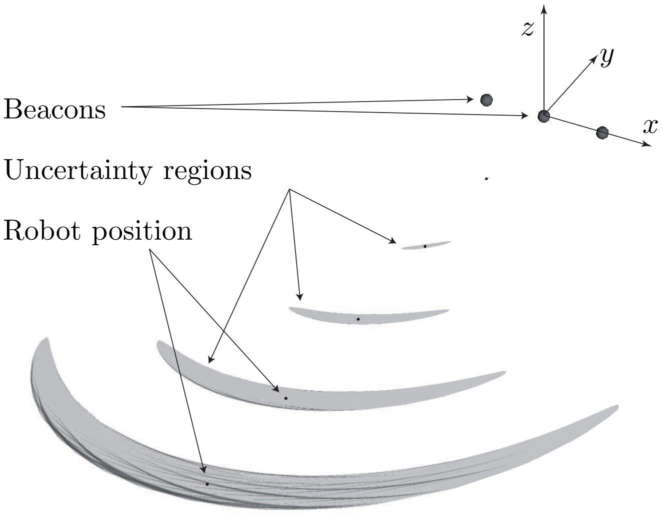


Figure 5.5: Shape of uncertainty regions for a ball neighborhood in the image projection line $\mathcal{I}$.

is:

$$c := c^B \circ c^C, \tag{5.34}$$

where $c^C : \mathcal{Q} \to \mathcal{P}$ maps the world frame, with coordinates (x_w, y_w, θ_w), to the intermediate space $\mathcal{P}$.

$$\begin{aligned} c^C \ &: \ \mathcal{Q} \to \mathcal{P} \\ (x_w, y_w, \theta_w) &\mapsto \begin{bmatrix} \arctan(\dfrac{y_w \sin(\theta_w) - x_w \cos(\theta_w)}{x_w \sin(\theta_w) + y_w \cos(\theta_w)}) \\ \\ \theta_w - \arctan(\dfrac{y_w \sin(\theta_w) - x_w \cos(\theta_w)}{x_w \sin(\theta_w) + y_w \cos(\theta_w)}) \\ \\ \sqrt{x_w^2 + y_w^2} \end{bmatrix} \end{aligned} \tag{5.35}$$

Let $c^A : \mathcal{I} \to \mathcal{Y}$ be a map from the camera image line to the image projection space:

$$\begin{aligned} c^A \ &: \ \mathcal{I} \to \mathcal{Y} \\ (\iota_1, \iota_2, \iota_3) &\mapsto \begin{bmatrix} \arctan(\iota_1) + \pi/2 \\ \arctan(\iota_2) + \pi/2 \\ \arctan(\iota_3) + \pi/2 \end{bmatrix} \end{aligned} \tag{5.36}$$

The following diagram summaries the composition of all the maps to achieve the robot's pose in $\mathsf{SE}(2)$:

$$\mathcal{I} \xrightarrow{c^A} \mathcal{Y} \xrightarrow{c^{B^{-1}}} \mathcal{W} \xrightarrow{c^{C^{-1}}} \mathcal{Q} \tag{5.37}$$

5.2 Navigation function on the image projection space

Building a navigation function that imposes the boundaries of $\mathcal{F}$ and $\mathcal{V}$ is best accomplished in $\mathcal{Y}$ space. Let $\bar{\varphi}$ be a potential function:

$$\begin{aligned} \bar{\varphi} \ &: \ \mathcal{Y} \to [0, 1] \\ \zeta &\mapsto \frac{\bar{\bar{\varphi}}(\zeta)^k}{\epsilon + \bar{\bar{\varphi}}(\zeta)^k} \end{aligned} \tag{5.38}$$

with:

$$\bar{\bar{\varphi}}(\zeta) := \frac{\left((\zeta_1 - \zeta_1^*)^2 + (\zeta_2 - \zeta_2^*)^2 + (\zeta_3 - \zeta_3^*)^2\right)^m}{(\zeta_{max} - \zeta_1)(\zeta_1 - \zeta_2)(\zeta_2 - \zeta_3)(\zeta_3 - \zeta_{min})} \tag{5.39}$$

By construction, the function $\bar{\varphi}$ equals unity on the boundary of $\mathcal{Y}$ and has a global minima at the goal $(\zeta_1^*, \zeta_2^*, \zeta_3^*)$. $\bar{\varphi}$ is continuous and differentiable, and it can be shown that it is a navigation function in $\mathcal{Y}$ as defined in [KR90]. The parameters ϵ, k and m shape the function $\bar{\varphi}$ to allow fine tuning of the resultant velocity vector field. $(\zeta_{min}, \zeta_{max})$ are the FOV constraints described in Figure 5.2(a). For more details see Section 3.5.4. The navigation function written in the configuration space is:

$$\varphi := \bar{\varphi} \circ c^B \circ c^C \tag{5.40}$$

Assuming a fully actuated body, the previous navigation function can be implemented in $\mathcal{Q} \subset \mathsf{SE}(2)$ by the following differential equation, with the projection of the beacon centroids in the image plane as measurements v:

$$\dot{q} = -\nabla\varphi = -Dc^{C^T} Dc^{B^T} \nabla\bar{\varphi} \circ c^A(v) \tag{5.41}$$

CHAPTER 6

Empirical validation

In this chapter we present the results of the implementation of the visual servoing algorithm developed in the previous chapters for the RHex robot. In Section 6.1, we outline the hardware and software components that comprise the image processing pipeline. It is important to keep in mind that this perceptual apparatus must remain fairly simple since it is located entirely onboard the robot and runs in real time as we detail below. In Section 6.2 we describe the controller implementation, emphasizing the two extensions to the simple version of the algorithm presented in Section 3.1.1 that compensate for the significant sensor noise and limitations in control authority inherent in this physical setting. We explain why the resulting closed loop (hybrid) behavior is still governed by the correctness results of Chapter 3, notwithstanding these real world adjustments. Finally, we present two applications: return home and auto-tuning. We include tables and figures of data summarizing our extensive experimental results for both indoor and outdoor implementations of the complete system.

6.1 The Perceptual Hardware and Software

The entire visual sensor suite is implemented on a second dedicated onboard 300MHz PC104 stack, running Linux, connected by local ethernet to the (QNX based) motor control stack documented in [SBK01]. We implement the following computational pipeline on this second stack at a 10 Hz update rate:

1. Video acquisition: is accomplished by a Sony DFW300 camera via a firewire connection.
2. Image processing library: Early vision is accomplished using the SVision library that I developed for the RHexLib infrastructure code. It is inspired by Hager's XVision [HT98], albeit considerably stripped down in comparison. We implement the following image processing methodology:
   - color calibration (this step is executed only at startup): A lookup table is used

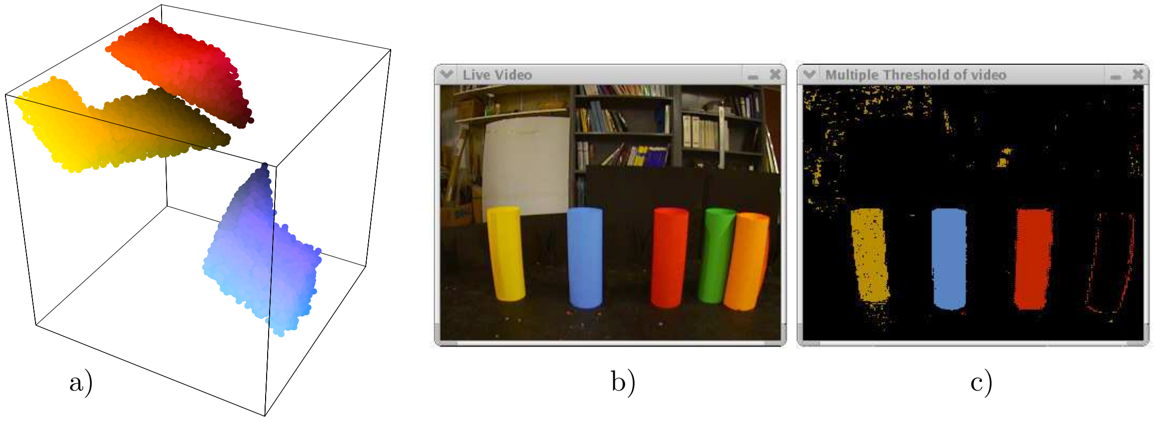


Figure 6.1: **a)** Color threshold classes in YUV color space; **b)-c)** Preview of the multi-color blob tracker.

for color classification in the YUV color space (the standard TV NTSC color space) with size 256 × 256 × 256. Different color classes are acquired by selecting different objects in the GUI's camera view. After a color class is acquired its size is increased[1] by a pre-defined amount in the luminance direction of the HLS color space (Hue, Luminance and Saturation) so as to maximize robustness to daylight changes, specifically, switching from shade to direct sun exposure.

- blob extraction: the standard 4-neighbor connected components algorithm is used as presented in [BB82]. A vector of mass, centroid and labeling class is returned per blob found.
- lens correction: the standard Heikkilä [HS97] lens model is used. Calibration is performed at startup using a flat checkerboard surface.

3. Image stabilization: The centroid information provided by the image processing library follows a post-processing roll correction. Since it assumed that the beacons project into a line, following Figure 5.2(a), roll correction is accomplished by fitting a line to the 2D centroid of the 3 blobs (chosen by size and class) and attaching a frame to it. The beacon coordinates are defined in relation to that frame. The following simplified expression is used in the experimental implementation, where (X_i, Y_i) are the centroids of the three beacons in the image plane:

$$\zeta_i = \arctan\left(\frac{X_i + \delta Y_i}{1+\delta^2}\right) + \pi/2, \tag{6.1}$$

with,

$$\delta := \frac{\sum X_i \sum Y_i - 3\sum X_i Y_i}{(\sum X_i)^2 - 3\sum X_i^2}. \tag{6.2}$$

[1] The color's acquired simply-connected volume is projected into the Hue and Saturation plane and then spread over an interval in the Luminance axis.

In the simulations developed in Section 3.5.4 the robot is assumed to live in the plane and therefore no pitch is considered. However, in the experimental implementation there can be large disturbances that pitch the robot enough for the beacons to leave the field of view either from the bottom or from the top of the image. We coded a state machine, described below, that in case of "emergency" will stop and rotate the robot in place until it relocates the beacons. This simple procedure corrected for all the temporary failures that occurred due to excessive pitching.

4. Supervisory state machine: The transitions between the controllers f_1 and $\bar{f}_2$ are implemented using a standard state machine formulation. The robot is initiated with controller $\bar{f}_2$. A transition occurs if the robot crosses the stable manifold approximation switching to controller f_1. If f_1 fails to bring the robot to a pre-defined neighborhood of the goal location, i.e. reaches the center manifold outside the goal's neighborhood and a fixed amount of time has passed, then another transition occurs, switching back to controller $\bar{f}_2$. The robot stops when it reaches the goal's neighborhood. As mentioned before, the state machine will also deal with particular emergency situations.

6.2 Controller Implementation

The control algorithms use the camera map exactly as defined above in Section 5.1. However the substantial perceptual noise and limitations in control authority associated with our physical RHex environment require two additional complications in the controller implementation.

First, although the horizontal plane behavior of the robot RHex is reasonably well approximated by the unicycle mechanics presented in Section 1.2.4, the limited number of gaits available for any given terrain [WLB$^+$04] typically dictate that the available fore-aft speed control be limited to a few discrete velocity magnitudes. Thus, a more accurate model of control authority would replace u_1 in equation (1.34) with a variable taking its values in a discrete set. Fortunately, gradient vector fields can be scaled in an arbitrary (albeit sign definite) manner with no change in steady state behavior. Namely, for any gradient field, $f(x) = -\nabla\varphi$ and any positive scalar valued function, $\sigma(x)$, observe that φ remains a Lyapunov function for the scaled field $\sigma(x)f(x)$. Our implementation using a discrete magnitude field can now be modeled by $\sigma(x) := \sigma_0/||f(x)||$.

Second, in systems where noise is introduced via imperfect perception or actuation, the vector field f_2 looses its φ-invariance. Although a thorough-going treatment of the stochastic version of our problem lies well beyond the scope of this paper, the reliance on gradient vector fields once again affords an intuitively simple "regulator" against these undesirable (and, ultimately, dangerous) fluctuations in proximity to the obstacles. Namely, suppose that the noise is additive and zero mean. Rewrite equation (3.30) as:

$$\dot{q} = \bar{f}_2(q) + v(t) \tag{6.3}$$

Define the new input $\hat{f}_2$ as:

$$\begin{aligned} \hat{f}_2 &:= \bar{f}_2(q) + \beta\left(\varphi^* - \varphi(q)\right) f_1(q) \\ &= \sigma J(A)\nabla\varphi + \beta\left(\varphi^* - \varphi\right) H\nabla\varphi, \end{aligned} \tag{6.4}$$

where β is a positive scalar and φ^* is the desired *target level set*, normally chosen to be slightly less then 1. The dynamics of φ for $\dot{q} = \hat{f}_2(q) + v(t)$ are:

$$\begin{aligned} \dot{\varphi} &= \underbrace{\nabla\varphi^T \bar{f}_2}_{=0} + \beta(\varphi^* - \varphi)\underbrace{\nabla\varphi^T H \nabla\varphi}_{\gamma} + \underbrace{\nabla\varphi^T v}_{w} \\ &= \beta\gamma(\varphi^* - \varphi) + w \end{aligned} \tag{6.5}$$

As $q(t)$ evolves over time, $\varphi(q(t))$ converges to a neighborhood of φ^* if $\gamma > 0$ and w is small in proportion. In practice this means that the robot will stay in the proximity of the target level set φ^* while it is in motion, escaping the center manifold. The experiments performed on RHex, described next, revealed that adding the second term to the vector field (6.4) is indeed necessary. The robot was unable to follow a particular level set when $\bar{f}_2$ was solely used. In contrast, note that f_1 is energy dissipative, hence standard arguments from Lyapunov theory establish its robustness against these sorts of perturbations without the requirement of any further modification.[2] Although formal robustness analysis is generally not available for nonlinear systems, the nondegenerate gradient systems of the kind introduced in this paper are structurally stable, hence "small" perturbations away from the nominal model are guaranteed to result in only "small" perturbations in the limit set.

The resulting modified input of (3.41) used in the experiments, before applying the scaling required for RHex's discrete actuation presented in the beginning of this section, is:

$$u_p := B^\dagger \left[\sigma J(A) + \beta\left(\varphi^* - \varphi\right) I\right] \nabla\varphi \tag{6.6}$$

As a final note we would like to remind the reader that throughout this thesis we consider only the problem of point stabilization and avoid the tracking problem. In the experimental implementation the robot eventually "tracks" a level set of the navigation function but still does not track any particular fixed trajectory. Tracking changes completely the structure of the problem since in general, time-invariant vector fields can no longer be used for control.

[2] Specifically, the Lie derivative of φ along $\hat{f}_1 := f_1 + v$ is "usually" negative — except possibly in a small neighborhood of the center manifold whose size is regulated by the relative magnitude of f_1 and the variance of v. It follows that this neighborhood remains an attractor "on average".

6.3 Return home

The set of experiments presented in this section relates to the problem of having the robot "return home", in effect, an implementation of point stabilization. We divide the experimental results into two parts:

The first data set, a trace of the visually perceived pose and energy level resulting from application of controller f_2, depicted in figures 6.2–6.6, gives a feeling for the robustness of these gradient style controllers as the robot roughly but reliably traces out the desired trajectory in the face of notable sensor noise, the inevitable perturbations from uneven ground, as well as the very severe parametric uncertainty arising from the crudeness of the unicycle model used to describe the horizontal plane behavior of RHex. At a far distance from the beacons, the pose estimation performs poorly, as seen in the high variance of the data. Three set of experiments, depicted in figures 6.2–6.4, are conducted outdoors using RHex's onboard camera only, according to the procedure documented above in Section 6.2, for two different target levels. The remaining two experiments, figures 6.5–6.6, were conducted indoors. Here the ground truth trajectory is presented side by side with the robot's perceived trajectory. One clearly observes the regularity of the robot's real trajectory in the face of high noise pose measurements.

The second data set — a graphical and tabular summary of convergence from several different initial configurations — portrays the nature of "practical stability" [MS03] assuring convergence to a small neighborhood of the goal pose with the guarantee of maintaining visibility (never losing sight of the triple-beacon landmark) along the way. This experiment was conducted indoors with the ground truth data acquired by an overhead camera running at 30Hz. Quantitatively, the interesting parameter to measure is the *mean arc-length ratio* of the path, defined in discrete time by:

$$\text{mean arc-length ratio} := \text{mean}_i \left[\frac{\sum_k \|q^i_k - q^i_{k-1}\|}{\|q^i_0 - q^*\|} \right], \tag{6.7}$$

where k spans the indexes of the samples for the ith experiment. Table 6.1 compiles the experimental results and Figure 6.7 illustrates the robot's real trajectory obtained via a ground truth system.

No chattering effect was observed in any of these experiments. This is due to the state machine formulation (that prevent f_1 and $\bar{f}_2$ to switch in an "incoherent" fashion) and RHex's actuation model, realizing discrete steps. Note however that in wheeled vehicles chattering may occur when controller $\bar{f}_2$ is used very close to the goal, i.e. with a very small energy. Since $\bar{f}_2$ will reside on a very small level set of the navigation function, this results in very small oscillations around the goal.

Table 6.2 presents a summary of all of the experiments presented in this section.

Table 6.1: Comparison between simulations of a unicycle and experiments executed using the robot RHex for the three beacon problem.

		#	Mean error position	Mean arc-length ratio
Simulations		368	5.3 cm	2.9
Experiments	a)	5	17.6 cm	9.3
	b)	5	17.8 cm	6.2
	c)	5	17.6 cm	6.5
	d)	5	26.1 cm	5.2
	e)	5	11.5 cm	5.5
	f)	5	27.9 cm	4.9

Table 6.2: Summary of experimental data for the return home application.

Controller	Figure	Ground	Location	Target energy	Data source
f_2	6.2	grass	outdoors	$\varphi^* = 0.9$	onboard
	6.3	packed dirt	outdoors	$\varphi^* = 0.9$	onboard
	6.4	packed dirt	outdoors	$\varphi^* = 0.8$	onboard
	6.5,6.6	cement	indoors	$\varphi^* = 0.8$	ground truth, onboard
f_1 and f_2	6.7	cement	indoors	$\varphi^* = 0.8$	ground truth

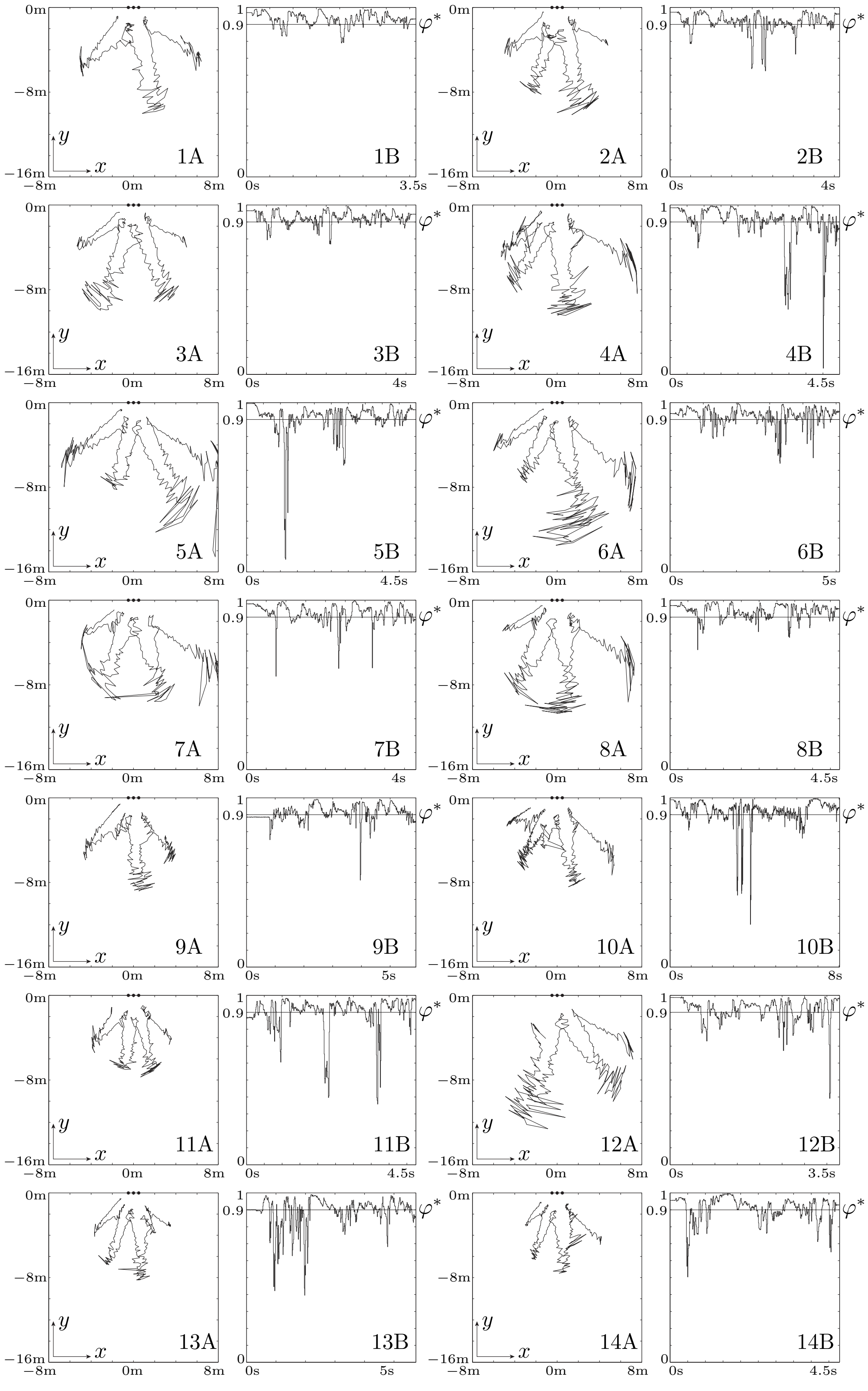


Figure 6.2: Subset of representative experiments on grass. Target energy is $\varphi^* = 0.9$

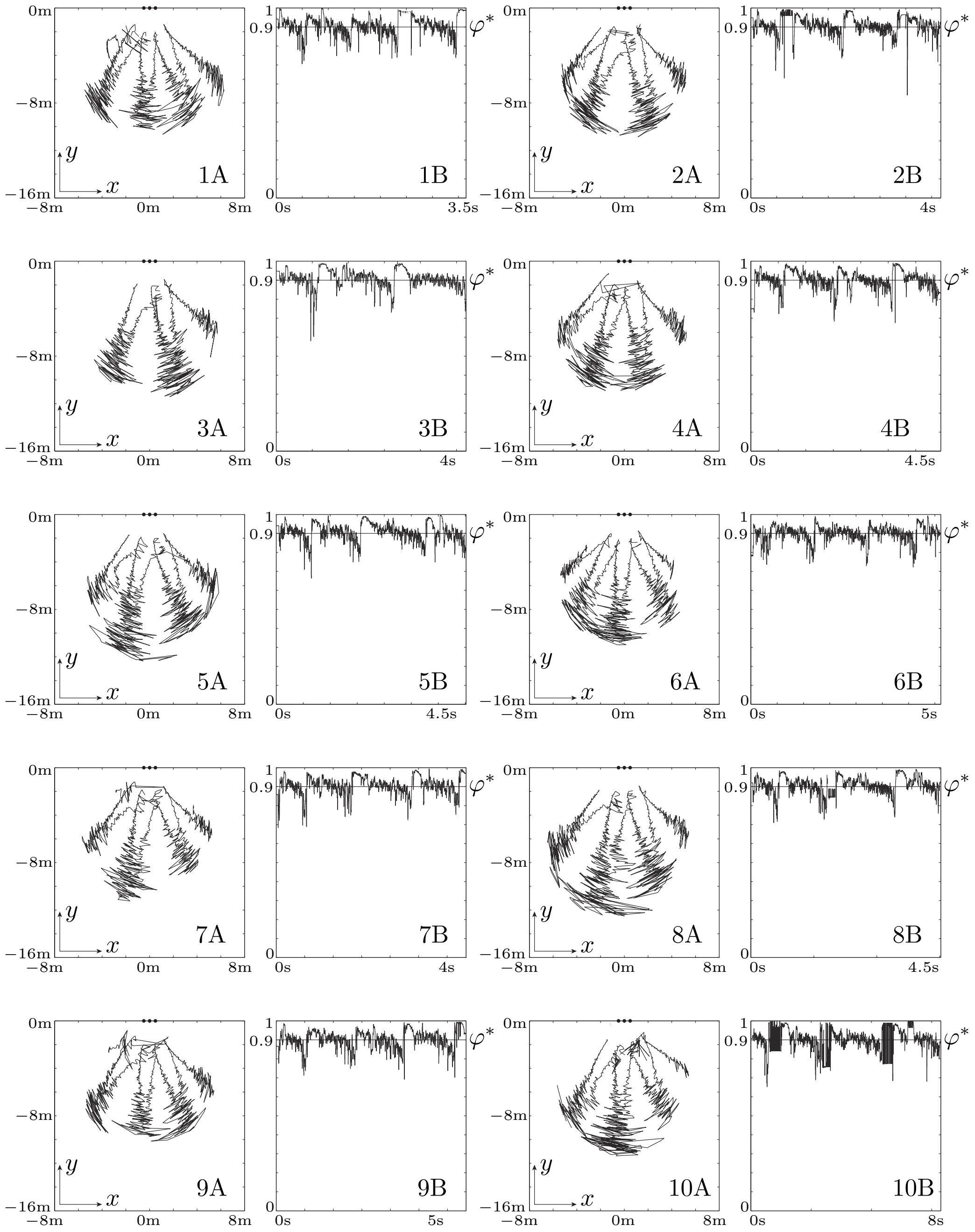


Figure 6.3: Subset of representative experiments on packed dirt. Target energy is $\varphi^* = 0.9$

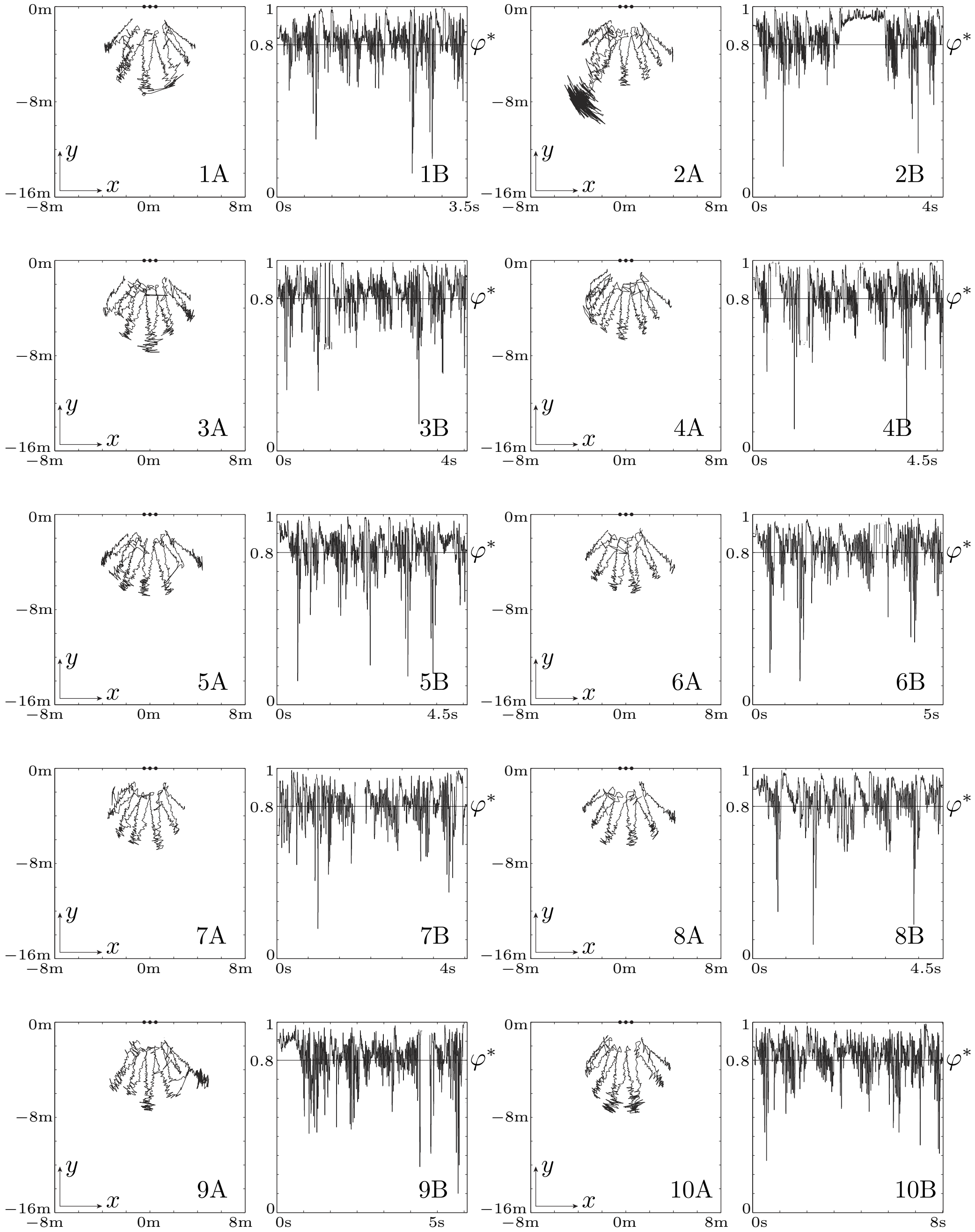


Figure 6.4: Subset of representative experiments on packed dirt. Target energy is $\varphi^* = 0.8$

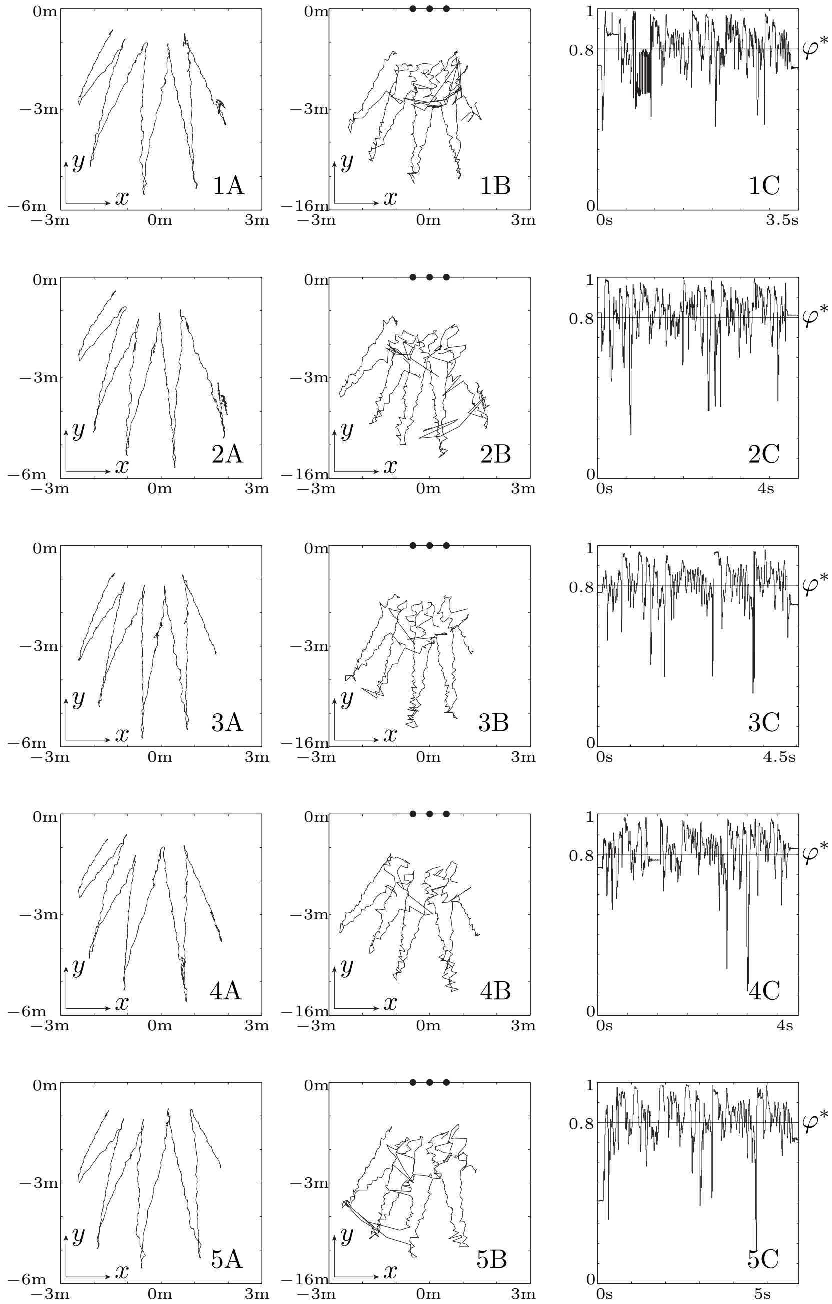


Figure 6.5: Subset of representative experiments indoor using ground truth. Target energy is $\varphi^* = 0.8$

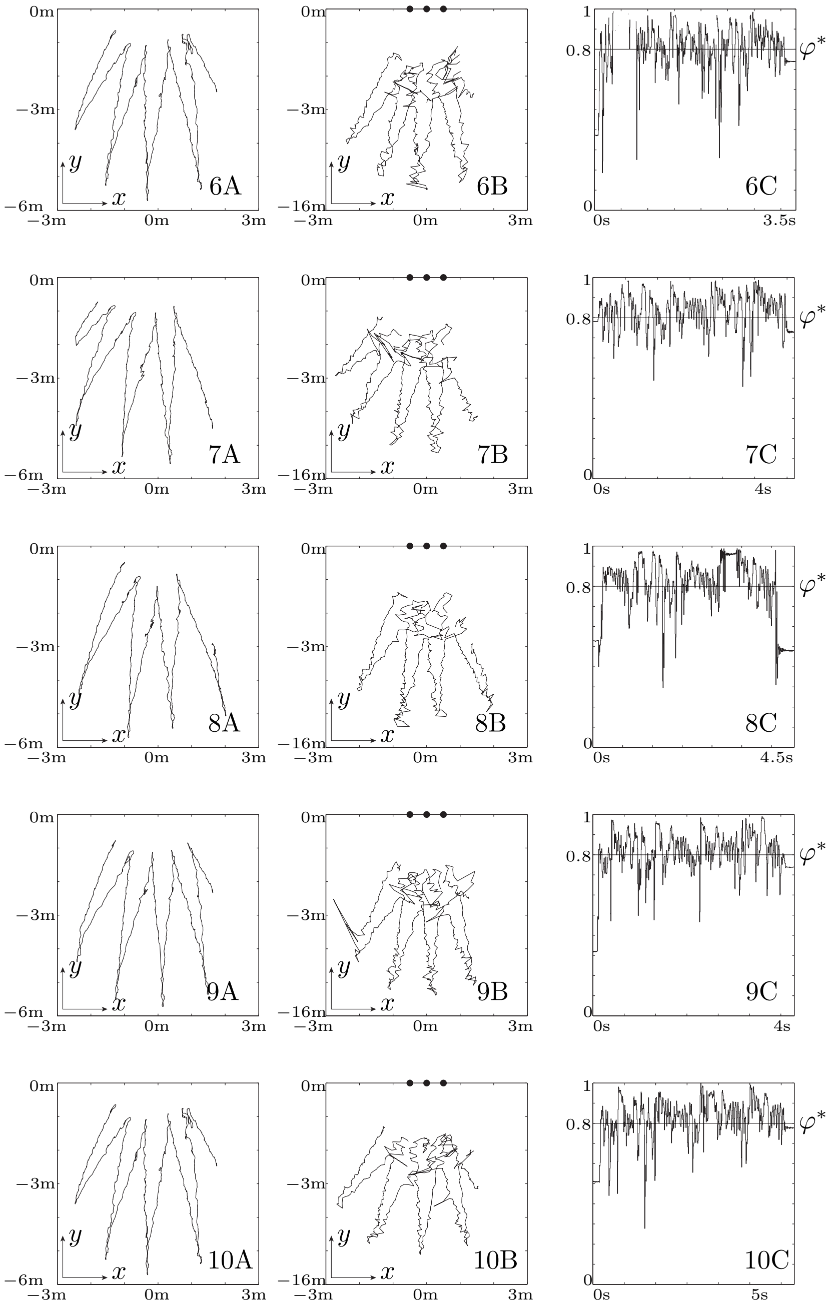


Figure 6.6: Subset of representative experiments indoor using ground truth. Target energy is $\varphi^* = 0.8$

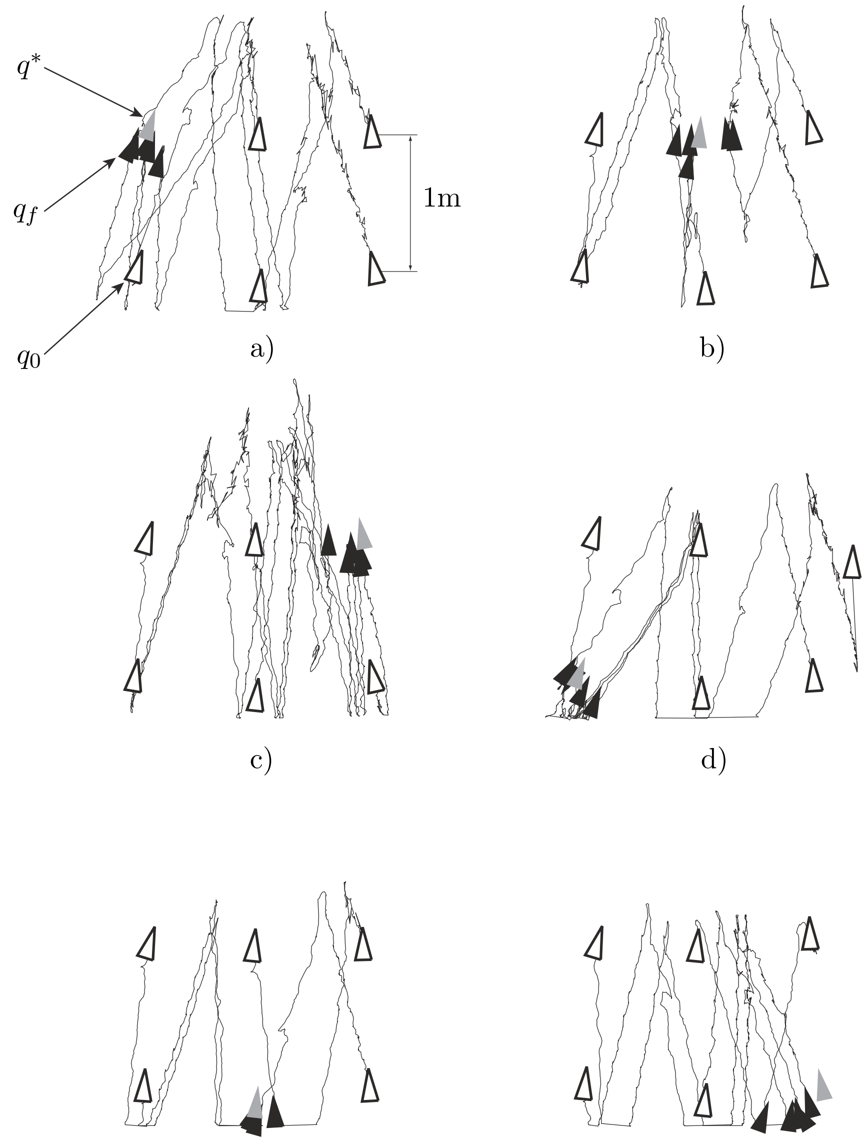


Figure 6.7: RHex's ground truth measurement experiments. Different goal locations q^* are represented by the gray triangles. The initial configurations q_0 are represented by the thick line white triangles and the final configurations q_f by the solid black triangles.

6.4 Auto tuning

Legged robots, due to the intermediate ground contact motion, have in general an infinite dimensional space of leg trajectories that can be chosen for different applications. Such types of trajectories include different gates such as walking, jogging, running, prancing, bouncing, etc. or specific applications such as gates for optimal endurance, speed, slopes, etc. Saranli et al. [SBK01] developed a piecewise continuous leg trajectory model for the RHex robot. Later, I participated in an effort with Weingarten [WLB$^+$04] to implement a self-tuning algorithm based on the Nelder-Mead descent [NM65] on the parameter set introduced by Saranli. In this work we addressed optimization for speed, endurance and walking slope angles of RHex. I describe in this section my contribution to this project.

We have integrated the controller presented in Section 6.2 into a parameter tuning infrastructure by recourse to engineered beacons, bright red vertically striped panels as depicted in Figure 6.8. The sequential composition of the constituent continuous controllers (see Section 2.3) is implemented by a supervisor defined by the standard finite state machine illustrated in Figure 6.9. Transition events between discrete supervisor states occur when the robot reaches (or, via surrogate means, supervisor states "believes" itself to have reached, in the cases noted below wherein it lacks the sensory modality to measure the relevant aspects of its state directly) its goal inside the domain. In critical situations the user can also trigger events.

The three primary supervisor states in an optimization trial are: servo-home, stabilize and experiment. Additional states are added to deal with undesired events. The numbered states illustrated in Figure 6.8 are described next:

- **Servo home** *Domain of attraction*: Entire workspace. The controller assumes that the robot is in any upright configuration inside the optimization area (in all experiments reported here we have used a 15×2m corridor). If no beacons appear on the robot's FOV then it rotates in place until it finds an appropriate constellation of 3 beacons. The corridor is so engineered with beacons that for every location therein, some interval of heading angles is guaranteed to afford a clear view of an appropriate constellation. It is for this reason that the domain of attraction arising from the servo home controller includes the entire workspace.
  Goal: Move the robot into a predefined home location preparing itself to start a new trial. A navigation function drives the robot to the home position while guaranteeing that the beacons stay in the FOV at all times.

- **Stabilizing phase** *Domain of attraction*: Locations in which the robot is behind the home line illustrated in Figure 6.8 and a set of beacons is centered in the FOV.
  Goal: Cross start line illustrated in Figure 6.8. This stage of the composition is introduced to eliminate the transient response of the gait being tested. The controller used is the same as in the experiment phase described next. Since we have no sensor

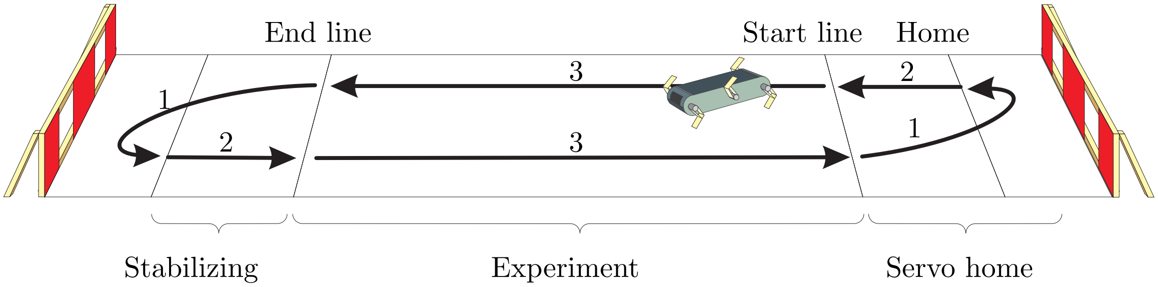


Figure 6.8: Illustration of a typical setup used for automated gait adaptation. A set of 3 beacons is placed at each side of a corridor. The robot moves back and forward registering against the beacons. The lines perpendicular to the corridor represent the start location for the stabilizing phase and the start/stop for the experiment phase.

capable of measuring directly when the transient response has ended, the goal in this state is triggered by distance. Thus while we adhere to the formal definition of sequential composition [BRK99] with respect to a surrogate projection of greatly reduced dimension (a projection of the robot's three degree of freedom configuration in the horizontal plane), this is only a coarse substitute for the more refined goal that would need to be defined in the underlying state space of the robot's full 12 dimensional rigid body position and velocity.

- **Experiment phase** *Domain of attraction*: The robot must be over the start line. *Goal*: Cross end line illustrated in Figure 6.8. The experiment phase drives the robot in a straight line for a fixed length. The controller maintains a constant forward velocity and steers the robot through the corridor so that in stays on a line as much as possible. In order to eliminate disturbances introduced by the steering leg offsets a dead zone is added to the yaw controller resulting in a 90% no steering motion on slow gaits.

The recovery states illustrated in Figure 6.9 are activated when the robot temporarily loses the beacons during a trial. Heuristically, the robot turns in the direction in which the beacons are spotted last. If the recovery does not bring to robot back into track within a couple of seconds then the trial is aborted and the robot returns home using the previously described servo home controller.

The efficiency of the automatic leg parameter tuning algorithm is judged by the amount of human interaction. Although the state machine described above accounts for almost every foreseeable contingency, there still exist situations that require human intervention, notably when the robot flips over, tho motor temperatures reach the point of incurring damage or the batteries need to be changed. Thus, while not entirely displaced, the burden on the human operator is substantially reduced, allowing useful attention to other work while tuning progresses, thereby allowing for longer and more accurate tuning sessions.

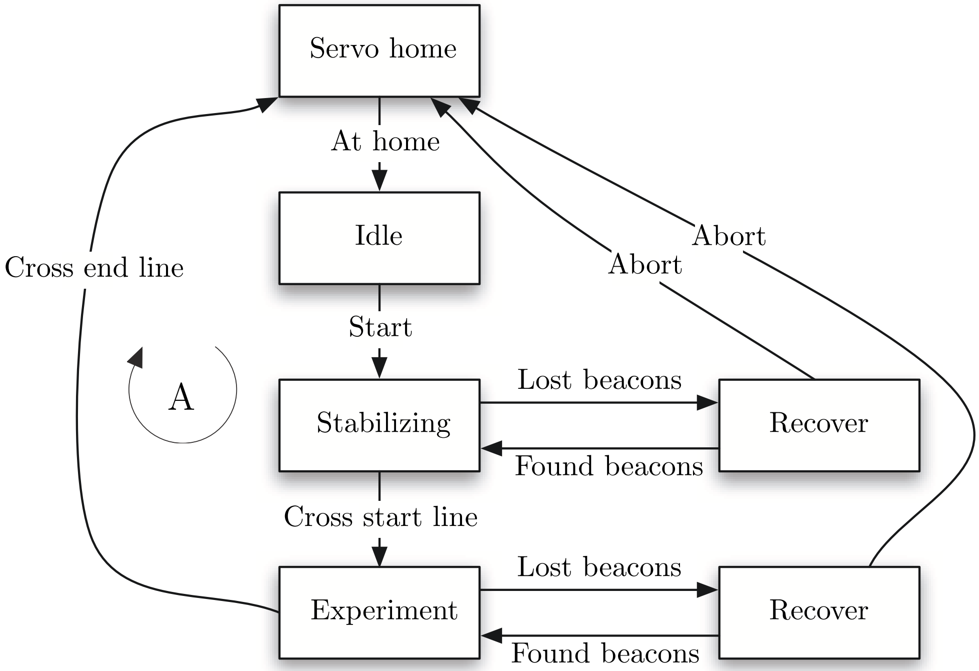


Figure 6.9: Illustration of state machine used for automated gait optimization. A trial is considered successful if loop A occurs. Other loops occur if the robot loses sight of the beacons for longer then a predefined time or a critical situation occurs. Only critical situations require human intervention, in general the robot is able to recover by rotating in place until the beacons appear in the field of view of the onboard camera.

In addition to greatly relieving operator burden, the automation system is both more reliable and accurate than the human operated version at speeds less than 1.3m/s. To test the attributes of the vision system we ran trials at three constant speeds over an 8m linoleum course. Table 6.3, borrowed from Weingarten's article [WLB$^+$04], documents the better than doubled performance the robot achieved while significantly reducing the percentage of the run where steering inputs are used to keep the robot on course. Furthermore, the percentage of experiments that need to be re-evaluated (redo rate = unsuccessful runs/total runs) is greatly reduced. At lowest speeds (approximately 0.5m/s) the vision system proved to work entirely without human assistance as opposed to every run without vision. As the velocity of the robot increases, it becomes more prone to flipping and thus the experimenter had to intervene to right the robot. The automation system fails at high speeds (greater than 1.5 m/s) because of the low frame rate returned by the vision system and image blur due to a long exposure time.

Table 6.3: Accuracy and reliability of the vision based automated parameter tuning system.

Test	Speed	Automated	Human	Trials
Trial Timing	0.5 m/s	∼ 0.2 s	∼ 0.5 s	50
(std. dev.)	1.0 m/s	∼ 0.3 s	∼ 0.5 s	50
	2.0 m/s	—	∼ 0.5 s	20
Steering Rate	0.5 m/s	∼ 10%	∼ 70%	10
(percentage)	1.0 m/s	∼ 40%	∼ 85%	10
	2.0 m/s	—	∼ 90%	10
Run Redo Rate	0.5 m/s	∼ 2%	∼ 10%	50
(percentage)	1.0 m/s	∼ 10%	∼ 10%	50
	2.0 m/s	—	∼ 20%	20
Human Assisted	0.5 m/s	0%	100%	50
(percentage)	1.0 m/s	∼ 3%	100%	30
	2.0 m/s	—	100%	5

CHAPTER 7

Conclusions and future directions

In this thesis we present a robust visual servo suitable for registering a legged robot with limited perception relative to engineered landmarks over rugged outdoor terrain. At the heart of our algorithm is a provably correct hybrid controller that reuses navigation functions developed for fully actuated bodies on kinematically constrained systems. It is straightforward to extend the guarantee of obstacle avoidance. Verifiable assumptions are given for convergence to an arbitrarily small neighborhood of the goal. We identify the parameters that govern the efficiency of the control law and suggest different methods for computing these. We present various simulations for different perceptual and motion models, and various sets of controller parameters.

We then address a large class of dynamical systems by redeveloping the proposed hybrid controller on a mechanical systems framework. We suggest how the vector fields developed for kinematic systems can be lifted to the dynamic setting with the introduction of damping and proportional gain type constants. The simulations presented suggest this lifting can be readily realized in real applications, by proper choice of the damping and gain.

We summarize the results of an extensive empirical implementation on the legged robot RHex that comprise of two relevant applications: set point stabilization on the plane that we denote by return home, and automated gait parameter tuning.

- The return home application attests for the robustness of the proposed hybrid controller combined with the visual sensor by conducting experiments both indoor and outdoor as well as in different types of terrains, such as grass, dirt, or cement. We conclude that in the presence of unfiltered high variance visual sensor noise, a crude robot motion model, and ground perturbations, point stabilization is achievable with very stable robot trajectories.

- We present a real world autonomous robotic application by composing a set of the proposed hybrid controllers with carefully geometrically engineered beacons that allow the robot to implement a gait parameter tuning procedure in a nearly autonomous manner. We present the supervisory state-machine formulation that encodes almost all conceivable recoverable contingencies and guides the robot through each step of the leg parameter tuning experiment. We conclude that the vision-based automated tuning runs exhibit markedly increased accuracy and improved steering relative to human-controlled experiments.

7.1 Short term applications

The work developed in this thesis can generate a range of possibilities for near and long future exploration and study. We focus here on a possible extension to a contemporary approach based on Lie algebra techniques, that we review in Section 1.2.3. Interpreting nonholonomic systems from a geometric control perspective where the configuration space has a fiber bundle structure brings many advantages due to the decoupling of the actuated (base) variables from the unactuated (group) states. As an example, Shammas [Sha06] developments in open-loop gait generation illustrate the ability to tackle higher dimensional systems recurring to lower dimensional base spaces.

One branch of ideas that I plan to follow in the near future includes results from this thesis on building "molded" energy conserving vector fields, the stabilizing and tracking algorithm developed by Morin et al [MS03], and the principally kinematic representation coined by Koiller [Koi92]. A class of stabilizing (and tracking) controllers for nonholonomic systems is based on tracking reference dynamics (or isolated trajectories) by adding a perturbation term that deals with the directions that are not spanned by the system actuated vector fields. Two sample frameworks based on this principle were introduced independently by Walsh [Wal94] and Morin [MS03]. We focus on the latter here. Consider the affine nonholonomic system:

$$\dot{g} = \sum_{i=1}^{m} X_i(g) u_i + P(g,t), \tag{7.1}$$

As revisited in Section 1.2.3, Morin [MS03] shows that if the vector fields X_i are left invariant then one can find a perturbation term f_ϵ with a specially crafted input that renders the system $\dot{z} = Z(z)$ asymptotically stable with $z = g f_\epsilon^{-1}$, also stabilizing g in the "practical sense", since the perturbation term f_ϵ is made small. One possible idea consists in changing the definition of f_ϵ by equipping it with a navigation function that encodes shape and possible group obstacles. Since the function $Z(z)$ can be chosen at will, one can imagine generating a reference vector field based on a second navigation function that deals exclusively with group obstacles. This may result in a framework that can simultaneously and efficiently encode shape obstacles, such as joint limits, with group obstacles, such as collisions, is a decoupled fashion.

A fundamental condition for Morin's algorithm to stabilize system (C.2) is that the vector fields X_i be left-invariant. In general, not all systems will adhere to this property. In

Appendix C we describe a possible way of utilizing Morin's feedback stabilizing controller in principally kinematic systems. We end with a list of open questions for this interesting problem:

- Can we systematically impose obstacles in the shape and in the group by manipulating f_ϵ and $Z(z)$, guaranteeing the conditions of Morin's stabilizing algorithm?
- Can we adaptively change the "gains" of the perturbation term f_ϵ so that one can execute maneuvers with large amplitudes of motion (and little oscillations) and yet reach arbitrary small neighborhoods of the goal?

7.2 Long term applications

The successful fully autonomous navigation application presented in Section 6.4 motivates the exploration of new navigation applications both in the scientific and commercial world. Being able to have your legged robot teach itself how to walk without requiring engineered landmarks or even having a pet robot navigating autonomously at your home brings great opportunities for the continuation of the framework developed in this thesis. Improvements can be done in the image processing component, allowing for natural landmarks, or in the supervisory state machine, by using different topology-based algorithms for navigation. Although we have at this point a clear idea of the limitations and conditions of applicability of the class of hybrid feedback controllers developed in this thesis, we believe there is still room for new developments such as, for example, specific high dimensional nonholonomic systems and new motion and perception models.

APPENDICES

APPENDIX A

Fečkan's extension of the Bendixon's criteria

Definition A.1 (Fečkan [Fec01b]). Let $M \subset \mathbb{R}^l$ be an m-dimensional compact smooth orientable submanifold with a nonempty border ∂M. Hence ∂M is an $m-1$-dimensional compact smooth orientable sub-manifold. Assume that $m \geq 2$. Let $V \subset \mathbb{R}^n$ be a k-dimensional smooth submanifold of $\mathbb{R}^n$ with empty border $\partial V = \emptyset$. Let $\beta \in \mathrm{Lip}(\partial M, \mathbb{R}^n)$ be such that $\beta(M) \subset V$ and $\tau = \beta/\partial M$ satisfy:

I τ is injective on ∂M.

II The inverse $\tau^{-1} : \tau(\partial M) \to \mathbb{R}^l$ is Lipschitz on the set $\tau(\partial M) \subset \mathbb{R}^n$.

We call the set $S = \tau(\partial M)$ an $m-1$-V-*L-boundary* of V. It is a generalization of smooth submanifolds of V.

Theorem A.2 (Fečkan [Fec01b]). *Let $g_1, g_2, \cdots, g_p \in C^2(\mathbb{R}^n, \mathbb{R})$ be first integral of (3.10). If $V = G^{-1}[0]$ is a nondegenerate level set of the mapping $G = (g_1, g_2, \cdots, g_p)$ and in addition $\mathrm{div} f_2 \neq 0$ on V, then there is no $n-p-1$-V-L-boundary S of V which is invariant for (3.10).*

APPENDIX B

Reference of common nonholonomic systems

We present in this appendix a collection of common kinematic drift-free nonholonomic systems. It is not our intent to be extensive. Instead we focus on the most common models that have a robotics application. We start, however, with a small number of abstract models that have been studied by the nonlinear control community, due to the insight they have brought to the development of efficient robot control algorithms. We purposely do not include common systems such as the Roller Racer [KT95], the Snakeboard [LOMB94] or the recent RoboTrikke [CCFK05] since they cannot be described as kinematic drift-free systems.

B.1 Abstract models

B.1.1 "Mod Jacobi" system

The "Mod Jacobi" system, presented by Sastry [Sas99] has the property of requiring exactly 2 levels of bracketing for spanning $T\mathbb{R}^n$. Its general form defined in $\mathbb{R}^n$ is

$$\begin{aligned} \dot{x}_i &= u_i, && i = 1, \ldots, m \\ \dot{x}_{ij} &= x_i u_j, && 1 \leq i < j \leq m \\ \dot{x}_{ijk} &= x_{ij} u_k, && 1 \leq i, j, k \leq m \end{aligned} \tag{B.1}$$

where,

$$n = m + \frac{m(m-1)}{2} + \frac{(m+1)m(m-1)}{3} = \frac{m(1+3m+2m^2)}{6} \tag{B.2}$$

For the particular case for $m = 2$ we get $\dot{q} = X_1(q)u_1 + X_2(q)u_2$ with $X_1 = [1\ 0\ 0\ x_3\ 0]^T$ and $X_2 = [0\ 1\ x_1\ 0\ x_3]^T$. The largest filtration can be found by using 2 levels of Lie brackets:

$$\begin{bmatrix} X_1 & X_2 & [X_1, X_2] & [X_1, [X_1, X_2]] & [X_2, [X_1, X_2]] \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & x_1 & -1 & 0 & 0 \\ x_3 & 0 & x_1 & -2 & 0 \\ 0 & x_3 & 0 & 0 & -1 \end{bmatrix}$$

Sastry in [Sas99] proposes a stabilization algorithm based on sinusoid control where each set of variables $\{x_i, x_{ij}, x_{ijk}\}$ are stabilized in sequence, beginning with x_i and ending with x_{ijk}.

B.1.2 One Chain

The one chain system [Sas99] has the interesting property of being controllable and having a degree of nonholonomy $n-2$, therefore increasing linearly with the dimension of the system and being the "worst case" since it requires the highest number of Lie brackets to span $T\mathbb{R}^n$. It general form is:

$$\begin{aligned} \dot{x}_1 &= u_1 \\ \dot{x}_2 &= u_2 \\ \dot{x}_3 &= x_2 u_1 \\ &\vdots \\ \dot{x}_n &= x_{n-1} u_1 \end{aligned} \tag{B.3}$$

In vector field notation we get $\dot{q} = X_1(q)u_1 + X_2(q)u_2$ with $X_1 = [1\ 0\ x_2\ \cdots\ x_{n-1}]^T$ and $X_2 = [0\ 1\ 0\ \cdots\ 0]^T$. Using the notation $\mathrm{ad}_X Y = [X, Y]$ and $\mathrm{ad}_X^k Y = [X, \mathrm{ad}_X^{k-1} Y] = [X, [X, \ldots, [X, Y] \cdots]]$ one can show that for the vector fields in equation (B.3), with $k \leq 1$, we obtain

$$\mathrm{ad}_{X_1}^k X_2 = \begin{bmatrix} 0 & \cdots & (-1)^k & \cdots & 0 \end{bmatrix}^T, \tag{B.4}$$

where the nonzero entry is in the $(k+2)$-th entry. The matrix

$$\begin{bmatrix} X_1 & X_2 & \mathrm{ad}_{X_1} X_2 & \cdots & \mathrm{ad}_{X_1}^{k-2} X_2 \end{bmatrix} \tag{B.5}$$

is full rank and therefore system (B.3) is controllable. This system has been addressed by many authors, resulting in stabilizing controllers using time-varying feedback [Pom92, Sam95], sinusoidal and polynomial control [MS93, TMS93], flatness [FLMR95], backstepping [JP95], piecewise control [HM99, MNC92] and sliding mode control [FBP00].

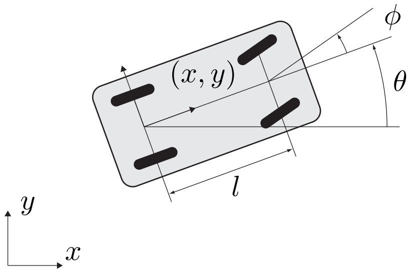


Figure B.1: The car.

B.2 Physical systems

B.2.1 Car

The car, illustrated in figure Figure B.1, is an extension of the unicycle by adding an extra constraint equation resulting from the back set of wheels. It takes the form

$$\begin{aligned} \dot{x} &= \cos(\theta)\cos(\phi)u_1 \\ \dot{y} &= \sin(\theta)\cos(\phi)u_1 \\ \dot{\phi} &= u_2 \\ \dot{\theta} &= \frac{1}{l}\sin(\phi)u_1 \end{aligned} \tag{B.6}$$

This system is controllable and has a degree of nonholonomy 2. The work on the car system has been very active until today. Some of the applications include Fuzzy controllers [YLML04, BMVSSO04, CLL05, DK06], trajectory tracking [WX03, TWC06], adaptive control [DK05, BB05], dealing with obstacles [QWP04, GVLS06], continuous curvature paths [FS04] and feed-forward parking [MDG07]. Since the car system can be converted into a one-chain form system a large body of work on stabilization is readily available.

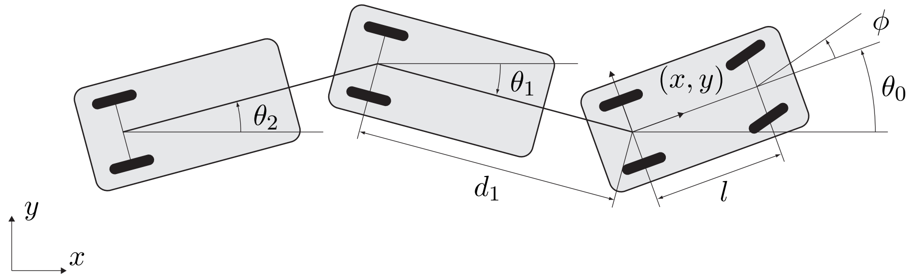


Figure B.2: The car with n trailers.

B.2.2 Car with n trailers

The car can be seen as a particular case of the car with n trailers. Following the notation of Figure B.2, it takes the following form

$$\begin{aligned}
\dot{x} &= \cos(\theta_0)u_1 \\
\dot{x} &= \sin(\theta_0)u_1 \\
\dot{\phi} &= u_2 \\
\dot{\theta}_0 &= \frac{1}{l}\tan(\phi)u_1 \\
&\vdots \\
\dot{\theta}_i &= \frac{1}{d_i}\left(\prod_{j=1}^{i-1}\cos(\theta_{j-1}-\theta_j)\right)\sin(\theta_{j-1}-\theta_j)u_1
\end{aligned}$$

Sørdalen [rda93] showed that there exists a change of coordinates that can transform the car with n trailers system into a one-chain form system. Other authors have presented stabilization algorithms including Tilbury [TMS93] and Samson [Sam95].

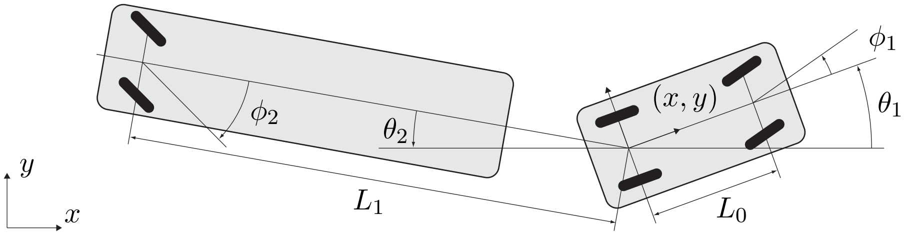


Figure B.3: The firetruck.

B.2.3 Fire truck

The fire truck was initially presented as an example for a problem of multi-input chain form stabilization by Bushnell et al. [BTS95]. Later Michalska and Rehman developed feedback stabilization algorithms based on model decomposition [Reh01] and discontinuous inputs [MR96]. The system equation take the form:

$$\begin{aligned} \dot{x} &= \cos(\theta_1)u_1 \\ \dot{y} &= \sin(\theta_1)u_1 \\ \dot{\phi}_1 &= u_2 \\ \dot{\theta}_1 &= \frac{1}{L_0}\tan(\phi_1)u_1 \\ \dot{\phi}_2 &= u_3 \\ \dot{\theta}_2 &= -\frac{1}{L_1}\sec(\phi_2)\sin(\phi_2 - \theta_1 + \theta_2)u_1 \end{aligned} \tag{B.7}$$

Rewriting equation (B.7) using vector fields $\dot{q} = X_1(q)u_1 + X_2(q)u_2 + X_3(q)u_3$ one can compute the degree of nonholonomy 2 since the following matrix is full rank:

$$\begin{bmatrix} X_1 & X_2 & X_3 & [X_1, X_2] & [X_1, X_2] & [X_1, [X_1, X_2]] \end{bmatrix} \tag{B.8}$$

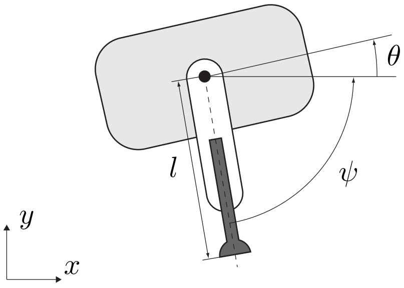


Figure B.4: The hopping robot.

B.2.4 Hopping robot in flight

The hopping robot in flight was initially studied by Li et al. [LMR89] and later revisited by Murray et al. [MS93]. The goal of this system is to reorient the body in midair to achieve a specific leg approach angle and leg extension. Given the conservation of angular momentum, and ignoring the ballistic trajectory during flight, the relevant system equations are

$$\begin{aligned} \dot{\psi} &= u_1 \\ \dot{l} &= u_2 \\ \dot{\theta} &= f(l)u_1 \end{aligned} \tag{B.9}$$

where the scalar $f(l)$ is a function of the masses, inertias and the leg length l. This system has degree of nonholonomy 1. Murry et al. present a solution for this system based on sinusoid control.

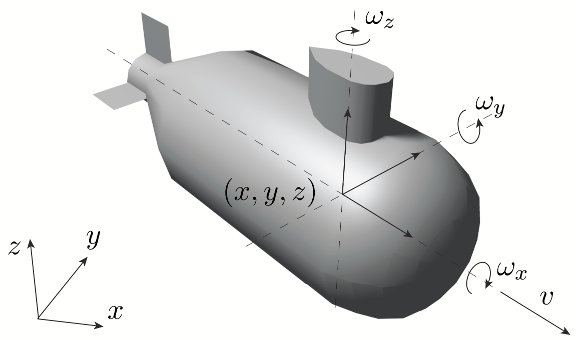


Figure B.5: The submarine.

B.2.5 Submarine

The submarine, studied by Nakamura et al. [NS91], can be interpreted as a 3-dimensional unicycle. Using the standard notation for roll, pitch and yaw, represented by the variables (ϕ, θ, ψ), takes the following system equations:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \cos(\psi)\cos(\theta) & 0 & 0 & 0 \\ \sin(\psi)\cos(\theta) & 0 & 0 & 0 \\ -\sin(\theta) & 0 & 0 & 0 \\ 0 & 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & 0 & \cos(\phi) & -\sin(\phi) \\ 0 & 0 & \sin(\phi)\sec(\theta) & \cos(\phi)\sec(\theta) \end{bmatrix} \begin{bmatrix} v \\ \omega_x \\ \omega_y \\ \omega_x \end{bmatrix} \tag{B.10}$$

This system is controllable and writing equation (B.10) as $\dot{q} = X_1(q)v + X_2(q)\omega_x + X_3(q)\omega_y + X_4(q)\omega_z$ results in the following full rank matrix:

$$\begin{bmatrix} X_1 & X_2 & X_3 & X_4 & [X_1, X_3] & [X_1, X_4] \end{bmatrix} \tag{B.11}$$

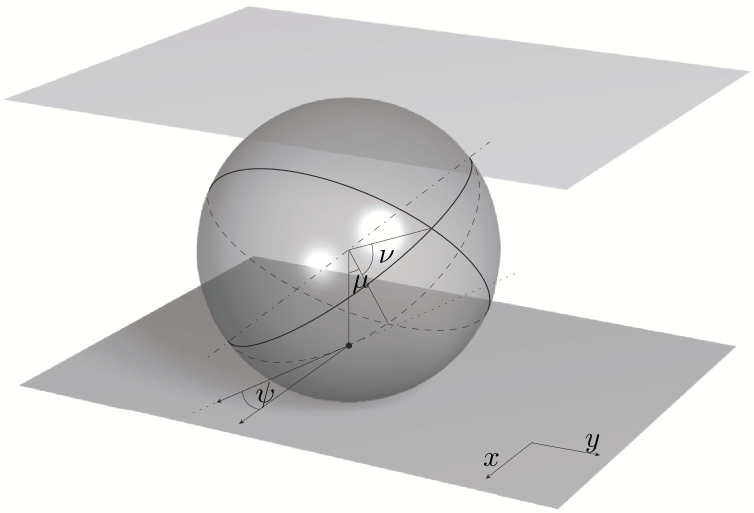


Figure B.6: The ball-place system.

B.2.6 Ball-Plate

The ball-plate mechanism was initially presented by Montana [Mon88]. Later feedback stabilization algorithms were presented by Sampei [SMS$^+$99], Oriolo [OV01, OV05] and Date [DSIK04]. Matsuo [MTS99] presented a Fuzzy logic approach and Svinin [SH06,SH07] specializes the stabilization problem by assuming only limited contact. The system equations following Oriolo's notation are:

$$\begin{aligned}
\dot{x} &= u_1 \\
\dot{y} &= u_2 \\
\dot{\mu} &= \frac{\cos(\psi)}{\rho}u_1 - \frac{\sin(\psi)}{\rho}u_2 \\
\dot{\nu} &= -\frac{\sin(\psi)}{\rho\cos(\mu)}u_1 - \frac{\cos(\psi)}{\rho\cos(\mu)}u_2 \\
\dot{\psi} &= \frac{\tan(\mu)\sin(\psi)}{\rho}u_1 + \frac{\tan(\mu)\cos(\psi)}{\rho}u_2
\end{aligned}$$

APPENDIX C

Notes on principally kinematic systems.

We collect in this appendix some notes relating the work of Morin *&* Samson on feedback stabilization of affine systems, with the classification of principally kinematic systems. We hope to use these results, tied closely together with the class of algorithms presented in this thesis, for near future studies.

Definition C.1. A vector field X on a Lie group G is left-invariant if $\forall g, h \in G$,

$$T_h L_g X(h) = X(gh) \tag{C.1}$$

Let G be a group and consider the affine system [MS03]:

$$\dot{g} = \sum_{i=1}^{m} X_i(g) u_i + P(g,t), \tag{C.2}$$

where $g \in G$, the vector fields X_i are left-invariant and $P(g,t)$ is a drift term. Consider also the reduced principal kinematic mechanical system equations [Ost96, SCR05]:

$$\begin{aligned} T_g L_{g^{-1}} \dot{g} &= A(r)\dot{r} \\ \dot{r} &= u \end{aligned} \tag{C.3}$$

Let $X^r(g) = T_e L_g A(r)$. Using the property $(T_g L_{g^{-1}})^{-1} = T_e L_g$ we verify that the vector field $X^r(g)$ is left invariant for any r:

$$T_g L_h X^r(g) = T_g L_h T_e L_g A(r) = T_e L_{hg} A(r) = X^r(hg) \tag{C.4}$$

Using this fact we can rewrite system (C.3) as:

$$\begin{aligned} \dot{g} &= X^r(g) u \\ \dot{r} &= u \end{aligned} \tag{C.5}$$

Equation (C.5) now resembles (C.2) up to the input integral parameterized $X^r(g)$ vector fields. Following Morin's notation, let f be parameterized by r such that

$$f^r(\theta) = f^r_1(\theta_1) f^r_2(\theta_2) \dots f^r_{n-m}(\theta_{n-m}), \tag{C.6}$$

with $f^r_j(\theta_j) = \exp(\epsilon_1 \sin(\theta_j) X^r_{\lambda(j)} + \epsilon_2 \cos(\theta_j) X^r_{\rho(j)})$, where $X^r = X^r(e) = T_e L_e A(r) = A(r)$. Let $z = g f^{r\,-1}_\epsilon$ with $z \in G$. Following the same constructive steps as in [MS03] we obtain

$$\dot{z} = T_g R_{f^{-1}} T_f L_z \left[\sum_{i=1}^{m} \left(X^r_i(f) - \frac{\partial f^r}{\partial r_j}(\theta) \right) u_i - \sum_{j=1}^{n-m} \dot{\theta}_j \frac{\partial f^r}{\partial \theta_j}(\theta) \right], \tag{C.7}$$

with $P(g,t) = 0$. Let $H_r(\theta)$ be:

$$H_r(\theta) = [X^r_1(f) - \frac{\partial f^r}{\partial r_1}(\theta) \quad \dots \quad X^r_m(f) - \frac{\partial f^r}{\partial r_m}(\theta) \;\; \frac{\partial f}{\partial \theta_1}(\theta) \quad \dots \quad \frac{\partial f}{\partial \theta_{n-m}}(\theta)] \tag{C.8}$$

If one can show that $H_r(\theta)$ is full rank then Morin *&* Samson's feedback stabilizing algorithm can be readily applied to the class of pricipally kinematic systems.

BIBLIOGRAPHY

BIBLIOGRAPHY

[AdRMF99] Angelo Arleo, José del R. Millán, and Dario Floreano, *Efficient learning of variable-resolution cognitive maps for autonomous indoor navigation*, IEEE Transactions Robotics and Automation **15** (1999), no. 6, 990–1000.

[Arn73] V. I. Arnold, *Ordinary differential equations*, The MIT press, 1973.

[Arn89] ———, *Mathematical methods of classical mechanics*, Springer-Verlag, 1989.

[ASO05] C.M. Anupoju, C.Y. Su, and M. Oya, *Adaptive motion tracking control of uncertain nonholonomic mechanical systems including actuator dynamics*, IEE Control Theory and Applications **152** (2005), no. 5, 575–580.

[Aus67] Louis Auslander, *On a problem of philip hall*, The Annals of Mathematics **86** (1967), no. 1, 112–116.

[AVK66] A. A. Andronov, E. A. Vitt, and S. E. Khaiken, *Theory of oscillators*, Pergamon Press: Oxford, 1966.

[Bak02] Andrew Baker, *Matrix groups, an introduction to lie group theory*, Springer-Verlag, 2002.

[BB82] D. Ballard and C. Brown, *Computer vision*, Prentice-Hall, 1982.

[BB05] Lotfi Beji and Yasmina Bestaoui, *Motion generation and adaptive control method of automated guided vehicles in road following*, IEEE Intelligent Transportation Systems **6** (2005), no. 1, 113–123.

[BC06] A. Birk and S. Carpin, *Merging occupancy grid maps from multiple robots*, Proceedings of the IEEE **94** (2006), no. 7, 1384–1397.

[BD97] Anthony Bloch and Sergey Drakunov, *Stabilization of brockett's generalized canonical driftless system*, IEEE Conference on Decision and Control, 1997.

[BDK00] Anthony Bloch, Sergey Drakunov, and Michael Kinyon, *Stabilization of nonholonomic systems using isospectral flows*, SIAM Jounal on Control and Optimization **38** (2000), no. 3, 855–874.

[BDW06] T. Bailey and H. Durrant-Whyte, *Simultaneous localization and mapping (slam): part ii*, IEEE Robotics & Automation Magazine **13** (2006), no. 3, 108–117.

[Bel65] R. Bellman, *Introduction to matrix analysis*, McGraw Hill, New York, 1965.

[BKMM96] A.M. Bloch, PS Krishnaprasad, J.E. Marsden, and R.M. Murray, *Nonholonomic mechanical systems with symmetry*, Archive for Rational Mechanics and Analysis **136** (1996), no. 1, 21–99.

[BL05] Francesco Bullo and Andrew D. Lewis, *Geometric control of mechanical systems*, Springer-Verlag, 2005.

[Blo03] A. M. Bloch, *Nonholonomic mechanics and control*, Springer-Verlag, 2003.

[BMCH04] Sourabh Bhattacharya, Rafael Murrieta-Cid, and Seth Hutchinson, *Path planning for a differential drive robot: Minimal length paths-a geometric approach*, Proc. of IEEE/RSJ International Conference on Intelligent Robotics and Systems, 2004, pp. 2793–2798.

[BMVSSO04] I. Baturone, F. J. Moreno-Velo, S. Sanchez-Solano, and A. Ollero, *Automatic design of fuzzy controllers for car-like autonomous robots*, IEEE Transactions on Fuzzy Systems **12** (2004), no. 4, 447–465.

[BR92] Anthony Bloch and Mahmut Reyhanoglu, *Control and stabilization of nonholonomic dynamic systems*, IEEE Transactions on Automatic Control **37** (1992), no. 11.

[BRK99] R. Burridge, A. Rizzi, and D. E. Koditschek, *Sequential composition of dynamically dexterous robot behaviors*, The International Journal of Robotics Research **18** (1999), no. 6, 534–555.

[Bro82] R. W. Brockett, *Control theory and singular riemannian geometry*, New Directions in Applied Mathematics, Springer-Verlag, New York, 1982.

[Bro83] ______, *Asymptotic stability and feedback stabilization*, Differential Geometric Control Theory, pp. 181–191, R. W. Brockett, R. S. Millman, and H. J. Sussmann, Birkhauser, Boston, 1983.

[BTS95] LG Bushnell, DM Tilbury, and SS Sastry, *Steering three-input nonholonomic systems: The fire truck example*, The International Journal of Robotics Research **14** (1995), no. 4, 366.

[CC05] Yeong-Chan Chang and Bor-Sen Chen, *Intelligent robust tracking controls for holonomic and nonholonomic mechanical systems using only position measurements*, IEEE Transactions on Fuzzy Systems **13** (2005), no. 4, 491–507.

[CCFK05] S. Chitta, Peng Cheng, E. Frazzoli, and V. Kumar, *Robotrikke: A novel undulatory locomotion system*, Robotics and Automation, 2005. ICRA 2005., April 2005, pp. 1597–1602.

[CH06] F. Chaumette and S. Hutchinson, *Visual servo control. i. basic approaches*, IEEE Robotics & Automation Magazine **13** (2006), no. 4, 82–90.

[CH07] ______, *Visual servo control. ii. advanced approaches*, IEEE Robotics & Automation Magazine **14** (2007), no. 1, 109–118.

[Cha97] S. A. Chaplygin, *On the motion of a heavy body of revolution on a horizontal plane (russian)*, Physiscs Section of the Imperial Society of Friends of Physics, Anthropology and Ethnographics, Moscow **9** (1897), 10–16.

[Chw04] Dongkyoung Chwa, *Sliding-mode tracking control of nonholonomic wheeled mobile robots in polar coordinates*, IEEE Control Systems Technology **12** (2004), no. 4, 637–644.

[CLK00] Noah J. Cowan, Gabriel A. D. Lopes, and Daniel E. Koditschek, *Rigid body visual servoing using navigation functions*, Conference on Decision and Control (Sydney, Australia), IEEE, 2000, pp. 3920–3926.

[CLL05] Chian-Song Chiu, Kuang-Yow Lian, and P. Liu, *Fuzzy gain scheduling for parallel parking a car-like robot*, IEEE Control Systems Technology **13** (2005), no. 6, 1084–1092.

[CLLH07] H. Jacky Chang, C. S. George Lee, Yung-Hsiang Lu, and Y. Charlie Hu, *P-slam: Simultaneous localization and mapping with environmental-structure prediction*, IEEE Transactions on Robotics **23** (2007), no. 2, 281–293.

[CN05] P. Coelho and U. Nunes, *Path-following control of mobile robots in presence of uncertainties*, IEEE Transactions on Robotics **21** (2005), no. 2, 252–261.

[Cow01] Noah J. Cowan, *Vision-based control via navigation functions*, Ph.D. thesis, University of Michigan, 2001.

[Cur70] M. Curtis, *Matrix groups*, Springer Verlag, New York, 1970.

[CWK02] Noah J. Cowan, Joel D. Weingarten, and Daniel E. Koditschek, *Visual servoing via navigation functions*, Trans. on Rob. and Aut. **18** (2002), no. 4, 521–533.

[DG81] Fred C. Dyer and James L. Gould, *Honey bee orientation: A backup system for cloudy days*, Science **214** (1981), no. 4524, 1041–1042.

[DK99] J.J. Duistermaat and J.A.C. Kolk, *Lie groups*, Springer-Verlag, 1999.

[DK05] Wenjie Dong and K. D. Kuhnert, *Robust adaptive control of nonholonomic mobile robot with parameter and nonparameter uncertainties*, IEEE Transactions on Robotics **21** (2005), no. 2, 261–266.

[DK06] Tamoghna Das and Indra Narayan Kar, *Design and implementation of an adaptive fuzzy logic-based controller for wheeled mobile robots*, IEEE Control Systems Technology **14** (2006), no. 3, 501–510.

[DSIK04] Hisashi Date, Mitsuji Sampei, Masato Ishikawa, and Masanobu Koga, *Simultaneous control of position and orientation for ball-plate manipulation problem based on time-state control form*, IEEE Transactions Robotics and Automation **20** (2004), no. 3, 465–480.

[DWB06] H. Durrant-Whyte and T. Bailey, *Simultaneous localization and mapping: part i*, IEEE Robotics & Automation Magazine **13** (2006), no. 2, 99–110.

[Elf89] A. Elfes, *Using occupancy grids for mobile robot perception and navigation*, Computer **22** (1989), no. 6, 46–57.

[EM92] S. Engelson and D. McDermott, *Error correction in mobile robot map learning*, Proceedings of the 1992 IEEE International Conference on Robotics and Automation (Nice, France), May 1992, pp. 2555–2560.

[EW06] Karin Erdmann and Mark J. Wildon, *Introduction to lie algebras*, Springer-Verlag, 2006.

[FBP00] T. Floquet, J. Barbot, and W. Perruquetti, *One-chained form and sliding mode stabilization for a nonholonomic perturbed system*, American Control Conference, 2000. (Chicago, IL, USA), vol. 5, 2000, pp. 3264–3268.

[FC05] David Folio and Viviane Cadenat, *A controller to avoid both occlusions and obstacles during a vision-based navigation task in a cluttered environment*, Proc. of 44th IEEE Conference on Decision and Control, December 2005.

[Fec01a] Michal Feckan, *Criteria on the nonexistence of invariant lipschitz submanifolds for dynamical systems*, Journal of Differential Equations (2001), no. 174, 392–419.

[Fec01b] ______, *A generalization of bendixson's criterion*, Proceedings of the American Mathematical Society, vol. 129, 2001, pp. 3395–3399.

[Fen79] Neil Fenichel, *Geometric singular perturbation theory for ordinary differential equations*, Journal of Differential Equations **31** (1979), 53–98.

[FLCP07] E. L. Faulring, K. M. Lynch, J. E. Colgate, and M. A. Peshkin, *Haptic display of constrained dynamic systems via admittance displays*, IEEE Transactions on Robotics **23** (2007), no. 1, 101–111.

[FLMR95] M. Fliess, J. Lévine, P. Martin, and P. Rouchon, *Flatness and defect of non-linear systems: introductory theory and examples*, International Journal of Control **61** (1995), no. 6, 1327–1361.

[FS04] T. Fraichard and A. Scheuer, *From reeds and shepp's to continuous-curvature paths*, IEEE Transactions on Robotics **20** (2004), no. 6, 1025–1035.

[FTBD01] D. Fox, S. Thrun, W. Burgard, and F. Dellaert, *Particle filters for mobile robot localization*, ch. Sequential Monte Carlo Methods in Practice, pp. 499–516, Springer Verlag, 2001.

[GaABD06] Jawhar Ghommam, Faïçal Mnif abd Abderraouf Benali, and Nabil Derbel, *Asymptotic backstepping stabilization of an underactuated surface vessel*, IEEE Control Systems Technology **14** (2006), no. 6, 1150–1157.

[GG00] Luis Giraldo and Francisco G. Gascón, *New proof and generalizations of the demidowitsch-schneider criterion*, Journal of Mathematical Physics **41** (2000), no. 9.

[GH83] J. Guckenheimer and P. Holmes, *Nonlinear oscillations, dynamica systems, and bifurcations of vector fields*, Verlag, New York, 1983.

[Gol50] H. Goldstein, *Classical mechanics*, Addison-Wesley, Reading Mass., 1950.

[Gou86] James L. Gould, *The locale map of honey bees: Do insects have cognitive maps?*, Science **232** (1986), no. 4752, 861–863.

[Gou98] ______, *Sensory bases of navigation*, Current Biology **8** (1998), 731–738.

[GSB07] Giorgio Grisetti, Cyrill Stachniss, and Wolfram Burgard, *Improved techniques for grid mapping with rao-blackwellized particle filters*, IEEE Transactions on Robotics **23** (2007), no. 1, 34–46.

[GVLS06] Paolo Robuffo Giordano, M. Vendittelli, Jean-Paul Laumond, and Philippe Souères, *Nonholonomic distance to polygonal obstacles for a car-like robot of polygonal shape*, IEEE Transactions on Robotics **22** (2006), no. 5, 1040–1047.

[Hal03] Brian C. Hall, *Lie groups, lie algebras, and representations: An elementary introduction*, Springer-Verlag, 2003.

[Her94] H. R. Hertz, *Der prinzipien der mechanik in neuem zusammenhange dargestellt*, Band III, Barth, Leipzig, 1894.

[Hir93] Shigeo Hirose, *Biologically inspired robots: Serpentile locomotors and manipulators*, Oxford University Press, 1993, Translator-Peter Cave and Translator-Charles Goulden.

[HK77] Robert Hermann and Arthur J. Krener, *Nonlinear controllability and observability*, IEEE Transactions on Automatic Control **22** (1977), no. 5, 728–740.

[HM99] J.P. Hespanha and A.S. Morse, *Stabilization of nonholonomic integrators via logic-based switching*, Automatica **35** (1999), no. 3, 385–394.

[HS97] Janne Heikkila and Olli Silven, *A four-step camera calibration procedure with implicit image correction*, Proc. of IEEE Conference on Computer Vision and Pattern Recognition, 1997, p. 1106.

[HT98] Gregory D. Hager and Kentaro Toyama, *Xvision: A portable substrate for real-time vision applications*, Computer Vision and Image Understanding **69** (1998), no. 1, 023–037.

[IKC$^+$06] R. J. Webster III, J. S. Kim, N. J. Cowan, G. S. Chirikjian, and A. M. Okamura, *Nonholonomic modeling of needle steering*, International Journal of Robotics Research **25** (2006), no. 5/6, 509–526.

[INMA99] Takayuki Ikeda, Taek-Kun Nam, Tsutomu Mita, and Brian Anderson, *Variable constraint control of underactuaated free flying robots*, Proc. of Conference on Decision and Control, no. 2539–2544, 1999.

[Jea01] Frédéric Jean, *Complexity of nonholonomic motion planning*, International Journal of Control **74** (2001), no. 8, 776–782.

[JP95] Z. Jiang and J. Pomet, *Backstepping-based adaptive controllers for uncertain nonholonomic systems*, Decision and Control, 1995., Proceedings of (New Orleans, LA, USA), vol. 2, December 1995, pp. 1573–1578.

[KB91] B. Kuipers and Y.-T. Byun, *A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations*, Journal of Robotics and Autonomous Systems **8** (1991), 47–63.

[KB01] D. E. Koditschek and H. I. Bozma, *Assembly as a noncooperative game of its pieces: analysis of 1d sphrere assemblies*, Robotica **19** (2001), no. 1, 93–108.

[KBK04] C.S. Karagoz, H.I. Bozma, and D.E. Koditschek, *Feedback-based event-driven parts moving*, IEEE Transactions on Robotics and Automation **20** (2004), no. 6, 1012–1018.

[Kha86] Oussama Khatib, *Real time obstacle avoidance for manipulators and mobile robots*, International Journal Robotics Research **5** (1986), no. 1, 90–99.

[Kha96] H. K. Khalil, *Nonlinear systems*, Prentice Hall, 1996.

[KM78] O. Khatib and J.-F. Le Maitre, *Dynamic control of manipulators operating in a complex environment*, Proceedings Third International CISM-IFToMM Symposium (Udine, Italy), Sep 1978, pp. 267–282.

[Kod87a] Daniel E. Koditschek, *Adaptive techniques for mechanical systems*, Fifth Yale Workshop on Applications of Adaptive Systems Theory (New Haven, CT), Center for Systems Science, Yale University, May 1987, pp. 259–265.

[Kod87b] ______, *Encyclopedia of artificial inteligence*, ch. Robot Control Systems, pp. 902–923, John Wiley and Sons, Inc., 1987.

[Kod87c] ______, *Exact robot navigation by means of potential functions: Some topological considerations*, IEEE International Conference on Robotics and Automation (Raleigh, NC), Mar 1987, pp. 1–6.

[Kod89a] ______, *The application of total energy as a lyapunov function for mechanical control systems*, Dynamics and control of multibody systems (Brunswick, ME, 1988), Amer. Math. Soc., Providence, RI, 1989, pp. 131–157. MR 91d:58237

[Kod89b] ______, *Robot planning and control via potential functions*, The Robotics Review (O. Khatib, J. Craig, and T. Lozano-Pérez, eds.), MIT Press, 1989, pp. 349–368.

[Kod91] D. E. Koditschek, *The control of natural motion in mechanical systems.*, ASME Journal of Dynamic Systems, Measurement, and Control **113** (1991), no. 4, 547–551.

[Kod92] ______, *Task encoding: Toward a scientific paradigm for robot planning and control*, Journal of Robotics and Autonomous Systems **9** (1992), 5–39.

[Koi92] J. Koiller, *Reduction of some classical non-holonomic systems with symmetry*, Archive for Rational Mechanics and Analysis **118** (1992), no. 2, 113–148.

[Kor99] D. Korteweg, *Über eine ziemlich verbreitete unrichtige behandlungsweise eines problemes der rollenden bewegung und insbesondere über klein rollende schwingungen um eine gleichgewichtslage*, Nieuw Archiefvoor Wiskunde **4** (1899), 130–155.

[KR90] Daniel E. Koditschek and Elon Rimon, *Robot navigation functions on manifolds with boundary*, Advances in Applied Mathematics **11** (1990), 412–442.

[KR03] G. Kantor and A. Rizzi, *Sequential composition for control of underactuated systems*, Tech. Report CMU-RI-TR-03-23, Robotics Institute, Carnegie Mellon, 2003.

[KT95] P. S. Krishnaprasad and D. P. Tsakiris, *Oscillations, se(2)-snakes and motion control*, Decision and Control, 1995., Proceedings of (New Orleans, LA, USA), vol. 3, December 1995, pp. 2806–2811.

[KT02] ByungMoon Kim and Panagiotis Tsiotras, *Controllers for unicycle-type wheeled robots: Theoretical results and experimental validation*, IEEE Transactions Robotics and Automation **18** (2002), no. 3, 294–307.

[KW94] D. Kortenkamp and T. Weymouth, *Topological mapping for mobile robots using a combination of sonar and vision sensing*, Proceedings of the Twelfth National Conference on Artificial Intelligence (Menlo Park), July 1994, pp. 979–984.

[Lau93] Jean-Paul Laumond, *Nonholonomic Motion Planning*, ch. 5, pp. 149–199, Springer, 1993.

[Lau98] ______, *Robot motion planning and control*, Springer-Verlag New York, 1998.

[LC93] Z. Li and J.F. Canny, *Nonholonomic Motion Planning*, Springer, 1993.

[LGM07] Zhijun Li, Shuzhi Sam Ge, and Aiguo Ming, *Adaptive robust motion/force control of holonomic-constrained nonholonomic mobile manipulators*, Systems, Man and Cybernetics, Part B, IEEE **37** (2007), no. 3, 607–616.

[LHC05] W. S. Lin, C. L. Huang, and M. K. Chuang, *Hierarchical fuzzy control for autonomous navigation of wheeled robots*, IEE Control Theory and Applications **152** (2005), no. 5, 598–606.

[Li00] Michael Y. Li, *Dynamics of differential equations on invariant manifolds*, Journal of Differential Equations (200), no. 168, 295–320.

[LK03] Gabriel A. D. Lopes and Daniel E. Koditschek, *Visual registration and navigation using planar features*, International Conference in Robotics and Automation (Taipei, Taiwan), IEEE, 2003.

[LK04] ______, *Level sets and stable manifold approximations for perceptually driven nonholonomically constrained navigation*, IEEE/RSJ International Conference on Intelligent Robots and Systems (Sendai, Japan), 2004.

[LK05] Gabriel Lopes and Daniel Koditschek, *Level sets and stable manifold approximations for perceptually driven non-holonomically constrained navigation*, Advanced Robotics **19** (2005), no. 10, 1081–1095.

[LK06] Gabriel A. D. Lopes and Daniel E. Koditschek, *Navigation functions for dynamical, nonholonomically constrained mechanical systems*, ch. I, pp. 135–155, Springer, 2006.

[LK07] Gabriel A. D. Lopes and Daniel E. Koditschek, *Visual Servoing for Nonholonomically Constrained Three Degree of Freedom Kinematic Systems*, The International Journal of Robotics Research **26** (2007), no. 7, 715–736.

[LMR89] Z. Li, R. Montgomery, and M. Raibert, *Dynamics and optimal control of a legged robot in flight phase*, IEEE International Conference on Robotics and Automation, 1989, pp. 1816–1821.

[LOMB94] Andrew Lewis, Jim Ostrowski, Richard Murray, and Joel Burdick, *Nonholonomic mechanics and locomotion: The snakeboard example*, Proc. of IEEE International Conference on Robotics and Automation, 1994.

[LR96] Jean-Paul Laumond and Jean-Jacques Risler, *Nonholonomic systems: Controllability and complexity*, Theoretical Computer Science **157** (1996), 101–114.

[LS93] G. Lafferriere and H. Sussmann, *Nonholonomic motion planning*, ch. A differential geometric approach to motion planning, pp. 235–270, Kluwer, 1993.

[Mac46] C. C. MacDuffee, *The theory of matrices*, (reprint of first edition) ed., Chelsea Publishing Company, New York, 1946.

[Mat90] M. J. Matari, *A distributed model for mobile robot environment-learning and navigation*, Master's thesis, MIT, Cambridge, MA, January 1990.

[MBF04] J. A. Marshall, M. E. Broucke, and B. A. Francis, *Formations of vehicles in cyclic pursuit*, IEEE Transactions on Automatic Control **49** (2004), no. 11, 1963–1974.

[MDG07] Bernhard M uller, Joachim Deutscher, and Stefan Grodde, *Continuous curvature trajectory design and feedforward control for parking a car*, IEEE Control Systems Technology **15** (2007), no. 3, 541–553.

[MFB04] P. Murrieri, D. Fontanelli, and A. Bicchi, *A hybrid-control approach to the parking problem of a wheeled vehicle using limited view-angle visual feedback*, International Journal of Robotics Research **23** (2004), no. 4-5, 437–448.

[MGS$^+$05] R. Menzel, U. Greggers, A. Smith, S. Berger, R. Brandt, S. Brunke, G. Bundrock, S. Hülse, T. Plümpe, F. Schaupp, E. Schüttler, S. Stach, J. Stindt, N. Stollhoff, and S. Watzl, *Honey bees navigate according to a map-like spatial memory*, Proceedings of the National Academy of Sciences of the USA (2005).

[MJ07] N. Moshtagh and A. Jadbabaie, *Distributed geodesic control laws for flocking of nonholonomic agents*, IEEE Transactions on Automatic Control **52** (2007), no. 4, 681–686.

[MLS94] Richard M. Murray, Zexiang Li, and S. Shankar Sastry, *A mathematical introduction to robotic manipulation*, CRC Press, Reading, Mass., 1994.

[MNC92] S. Monaco and D. N-Cyrot, *An introduction to motion planning using multirate digital control*, Proc. of IEEE Conference on Decision and Control, 1992, pp. 1780–1785.

[Mon88] D. J. Montana, *The kinematics of contact and grasp*, International Journal of Robotic Research **7** (1988), no. 3, 17–32.

[MR96] H. Michalska and F. U. Rehman, *Discontinuous feedback stabilization using guiding functions: the fire truck example*, Decision and Control, 1996., Proceedings of (Kobe, Japan), vol. 2, December 1996, pp. 1712–1713.

[MR99] Jerrold E. Marsden and Tudor S. Ratiu, *Introduction to mechanics and symmetry*, Springer-Verlag, 1999.

[MS93] R. M. Murray and S. S. Sastry, *Nonholonomic motion planning: steering using sinusoids*, IEEE Transactions on Automatic Control **38** (1993), no. 5, 700–716.

[MS03] Pascal Morin and Claude Samson, *Practical stabilization of driftless systems on lie groups: The transverse function approach*, IEEE Transactions on Automatic Control **48** (2003), no. 9, 1496–1508.

[MTS99] Takami Matsuo, Kenshi Tsuruta, and Haruo Suemitsu, *Fuzzy adaptive identification method based on riccati equation and its application to ball-plate control system*, Systems, Man, and Cybernetics, 1999. IEEE (Tokyo, Japan), vol. 6, 1999, pp. 162–167.

[NM65] J. A. Nelder and R. Mead, *A simplex method for function minimization*, Computer Journal **7** (1965), 308–313.

[NS91] Y. Nakamura and S. Savant, *Nonholonomic motion control of an autonomous underwater vehicle*, Intelligent Robots and Systems '91. (Osaka, Japan), November 1991, pp. 1254–1259.

[Ols00] C. F. Olson, *Probabilistic self-localization for mobile robots*, IEEE Transactions Robotics and Automation **16** (2000), no. 1, 55–66.

[OSK03] Masahiro Oya, Chun-Yi Su, and Ryozo Katoh, *Robust adaptive motion/force tracking control of uncertain nonholonomic mechanical systems*, IEEE Transactions Robotics and Automation **19** (2003), no. 1, 175–181.

[Ost96] James Ostrowski, *The mechanics and control of undulatory robotic locomotion*, Ph.D. thesis, California Institute of Technology, 1996.

[OV01] Giuseppe Oriolo and Marilena Vendittelli, *Robust stabilization of the plate-ball manipulation system*, Proc. of IEEE International Conference on Robotics and Automation, 2001, pp. 91–96.

[OV05] ______, *A framework for the stabilization of general nonholonomic systems with an application to the plate-ball mechanism*, IEEE Transactions on Robotics **21** (2005), no. 2, 162–175.

[PK94] D. Pierce and B. Kuipers, *Learning to explore and build maps*, Proceedings of the Twelfth National Conference on Artificial Intelligence (Menlo Park), July 1994, pp. 1264–1271.

[Pom92] J-B. Pomet, *Explicit design of time-varying stabilizing control laws for a class of controllable systems without drift*, Systems and Control Letters **18** (1992), 147–158.

[QWP04] Zhihua Qu, Jing Wang, and C. E. Plaisted, *presence of moving obstacles*, IEEE Transactions on Robotics **20** (2004), no. 6, 978–993.

[rda93] O. J. Sørdalen, *Conversion of the kinematics of a car with n trailers into a chained form*, Proc. of IEEE International Conference on Robotics and Automation, 1993, pp. 382–387.

[Reh01] F. Rehman, *Feedback stabilization of a fire truck by using model decomposition*, Multi Topic Conference, 2001. IEEE INMIC, 2001, pp. 195–199.

[Ris96] Jean-Jacques Risler, *A bound for the degree of nonholonomy in the plane*, Theoretical Computer Science **157** (1996), 129–136.

[RK91] E. Rimon and D. E. Koditschek, *The construction of analytic diffeomorphisms for exact robot navigation on star worlds*, Transactions of the American Mathematical Society **327** (1991), no. 1, 71–115.

[RK92] Elon Rimon and D. E. Koditschek, *Exact robot navigation using artificial potential fields*, IEEE Transactions on Robotics and Automation **8** (1992), no. 5, 501–518.

[Rol76] D. Rolfsen, *Knots and links*, Publish or Perish Press, 1976.

[SA07] Alessio Salerno and Jorge Angeles, *A new family of two-wheeled mobile robots: Modeling and controllability*, IEEE Transactions on Robotics **23** (2007), no. 1, 169–173.

[Sam95] C. Samson, *Control of chained systems - application to path following and time-varying pointstabilization of mobile robots*, IEEE Trans. Automation and Control **40** (1995), no. 1, 64–77.

[Sas99] S. Sastry, *Nonlinear systems – analysis, stability and control*, Springer, 1999.

[SBK01] U. Saranli, M. Buehler, and D. E. Koditschek, *Rhex: A simple and highly mobile hexapod robot*, The International Journal of Robotics Research **20** (2001), no. 7, 616–631.

[SCR05] Elie Shammas, Howie Choset, and Alfred Rizzi, *Natural gait generation techniques for principally kinematic mechanical systems*, Proceedings of Robotics: Science and Systems (Cambridge, USA), 2005.

[SH06] Mikhail Svinin and Shigeyuki Hosoe, *Simple motion planning algorithms for ball-plate systems with limited contact area*, 2006. ICRA 2006. Robotics and Automation (2006), 1755–1761.

[SH07] M. Svinin and S. Hosoe, *On motion planning for ball-plate systems with limited contact area*, Robotics and Automation, 2007 IEEE (Roma,), April 2007, pp. 1820–1825.

[Sha06] Elie Shammas, *Generalized motion planning for underactuated mechanical systems*, Ph.D. thesis, Carnegie Mellon University, 2006.

[SJ72] H. Sussmann and V. Jurdjevic, *Controllability of nonlinear systems*, Journal of Differential Equations **12** (1972), 95–116.

[SKP05] P. Stepan, M. Kulich, and L. Preucil, *Robust data fusion with occupancy grid*, Systems, Man and Cybernetics, Part C: **35** (2005), no. 1, 106–115.

[SMS$^+$99] M. Sampei, S. Mizuno, M. Segawa, M. Ishikawa, and H. Date, *A feedback solution to ball-plate problem based on time-scale control form*, American Control Conference, 1999. (San Diego, CA, USA), vol. 2, June 1999, pp. 1203–1207.

[Son98] Eduardo D. Sontag, *Mathematical control theory*, Springer-Verlag, 1998.

[Tan06] H. G. Tanner, *Mobile manipulation of flexible objects under deformation constraints*, IEEE Transactions on Robotics **22** (2006), no. 1, 179–184.

[Thr98] S. Thrun, *Learning metric-topological maps for indoor mobile robot navigation*, Artificial Intelligence **99** (1998), no. 1, 21–71.

[TMS93] D. Tilbury, R. Murray, and S. Sastry, *Trajectory generation for the n-trailer problem using goursat normal form*, Decision and Control, 1993., Proceedings of (San Antonio, TX, USA), December 1993, pp. 971–977.

[TMS95] Dawn Tilbury, Richard M. Murray, and S. Shankar Sastsy, *Trajectory generation for the n-trailer problem using goursat normal form*, IEEE Transactions on Automatic Control **40** (1995), no. 5, 802–819.

[Tor94] M. C. Torrance, *Natural communication with robots*, Master's thesis, MIT Department of Electrical Engineering and Computer Science, Cambridge, MA, 1994.

[TT86] Sir W. Thompson and P. G. Tait, *Treatise on natural philosophy*, University of Cambridge Press, Cambridge, 1886.

[TWC06] Pu-Sheng Tsai, Li-Sheng Wang, and Fan-Ren Chang, *Modeling and hierarchical tracking control of tri-wheeled mobile robots*, IEEE Transactions on Robotics **22** (2006), no. 5, 1055–1062.

[Vie92] A. Vierkandt, *uber gleitende und rollende bewegung*, Monatshefte der Math. und Phys. **III** (1892), 31–54.

[Vos85] A. Voss, *Ueber die differentialgleichungen der mechanik*, Mathematische Annalen **25** (1885), 258–286.

[VSS04] R. Vidal, O. Shakernia, and S. Sastry, *Following the flock*, IEEE Robotics & Automation Magazine **11** (2004), no. 4, 14–20.

[Wal94] Gregory Walsh, *Planning and feedback control for mechanical systems with nonholonomic constraints*, Ph.D. thesis, University of California at Berkeley, 1994.

[WGL04] Z. P. Wang, S. S. Ge, and T. H. Lee, *Robust motion/force control of uncertain holonomic/nonholonomic mechanical systems*, IEEE/ASME Transactions on Mechatronics **9** (2004), no. 1, 118–123.

[Wig94] Stephen Wiggins, *Normally hyperbolic invariant manifolds in dynamical systems*, Springer-Verlag, 1994.

[WLB$^+$04] J. Weingarten, G. Lopes, M. Buehler, R. Groff, and D. Koditschek, *Automated gait adaptation for legged robots*, Proc. of IEEE International Conference on Robotics and Automation, vol. 3, 2004, pp. 2153–2158.

[WS95] G. C. Walsh and S. S. Sastry, *On reorienting linked rigid bodies using internal motions*, IEEE Transactions Robotics and Automation **11** (1995), no. 1, 139–146.

[WX03] Danwei Wang and Guangyan Xu, *Full-state tracking and internal dynamics of nonholonomic wheeled mobile robots*, IEEE/ASME Transactions on Mechatronics **8** (2003), no. 2, 203–214.

[YB96] B. Yamauchi and R. Beer, *Spatial learning for navigation in dynamic environments*, IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics, Special Issue on Learning Autonomous Robots (1996).

[YLML04] Simon X. Yang, Hao Li, Max Meng, and Peter X. Liu, *An embedded fuzzy controller for a behavior-based mobile robot with guaranteed performance*, IEEE Transactions on Fuzzy Systems **12** (2004), no. 4, 436–446.

[Zim96] U.R. Zimmer, *Robust world-modeling and navigation in a real world*, Neurocomputing **13** (1996), 2–4.