11-1-1998

Motion Planning and Control of Cooperative Robotic Systems

Jaydev P. Desai
University of Pennsylvania

University of Pennsylvania Institute for Research in Cognitive Science Technical Report No. IRCS-98-27.

Motion Planning and Control of Cooperative Robotic Systems

Abstract

This thesis addresses the problem of motion planning for cooperative robotic systems. The problem of motion planning for a robotic system is stated as:

Given initial positions and orientations and goal positions and orientations for a collection, C, of robots, in workspace, W, generate a continuous trajectory for C avoiding contact with the obstacles, O_i, subject to various dynamical constraints of the system.

Because robots are physical systems subject to continuous laws of motion and driven by continuous actuators, we formulate the motion planning problem as an unconstrained variational problem using tools from optimal control and calculus of variations in the first part of the thesis. We develop a general framework for solving motion planning problems involving equality and inequality constraints.

In the second part of the thesis, we study planar human manipulation and develop a computational model for friction-assisted dual arm manipulation tasks incorporating the dynamics of the musculo-skeletal system. We show that our computational model predicts the force distribution and object trajectory in voluntary, relaxed movements. We further study similar tasks in the vertical plane and our experimental findings suggest that there is a great degree of repeatability in trajectories and velocity profiles across trials and subjects.

In the third part, we focus on extending this computational model to plan and control cooperative robotic systems. We solve the dynamic motion planning problem for a system of cooperating robots in the presence of geometric and kinematic constraints, and test the resulting open-loop trajectories on the experimental testbed.

In the last part of the thesis, we explore the application of the motion planning algorithms when the number of robots in *C* is very large. Because of increased computational time, we use optimal sensor-based closed loop motion plans. These are combined with the framework of graph theory and optimal control to guarantee provable measure on the performance of the entire system.

The main contributions of the thesis are: (a) studying trajectory generation and force distribution in human dual arm manipulation; and (b) a set of motion planning algorithms for cooperating robot systems subject to dynamic constraints.

Comments

University of Pennsylvania Institute for Research in Cognitive Science Technical Report No. IRCS-98-27.

MOTION PLANNING AND CONTROL OF COOPERATIVE ROBOTIC SYSTEMS

JAYDEV P. DESAI

A Dissertation in

MECHANICAL ENGINEERING AND APPLIED MECHANICS

Presented to the Faculties of the University of Pennsylvania
in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy.

1998

Professor Vijay Kumar
Supervisor of Dissertation

Professor James P. Ostrowski
Co-Supervisor of Dissertation

Professor Haim Bau
Graduate Group Chairperson

Acknowledgment

I would like to express my gratitude to the people who made an important difference and played a vital role in the successful completion of my Doctoral studies at the University of Pennsylvania.

I am grateful to my advisor Professor Vijay Kumar without whose competence, dedication, generosity, and vision this dissertation would not have been possible. Professor Kumar has been my mentor and an inspiring role model. His energy and relentless support encouraged me towards the completion of my Masters in Mathematics. Professor James Ostrowski, my co-advisor since 1996, provided valuable advice and insight towards my work. I am very thankful for his opinion and guidance during the course of my dissertation.

I am very thankful to Professor Joel Burdick (Caltech), Chairman of the committee, Professor G. K. Ananthasuresh, Professor Ruzena Bajcsy, Professor Vijay Kumar and Professor James Ostrowski for agreeing to be the members of my dissertation committee and for taking the time to read my thesis and provide valuable suggestions.

This dissertation has allowed me to work with people who are both competent and compassionate. I have been fortunate to receive constant guidance and support from Professor Ruzena Bajcsy. She has been a blessing with her wealth of experience and knowledge. Others whose advice and perspective were especially meaningful include Professor P. S. Ayyaswamy and Professor Herman Gluck. Over the years I have come to rely on their advice and have great respect for them. I am also thankful to the Institute for Research in Cognitive Sciences for awarding me the Graduate Student Fellowship during the period 1995-1997.

I would like to thank my past colleagues, Greg Garvin and Milos Žefran, whose experimental setup for planar two arm manipulation tasks set off my initial research in the area of friction assisted planar manipulation. I would also like to thank my past and present colleagues who have made my stay in the GRASP laboratory very enjoyable. Many thanks to my first officemate Robert Mandelbaum for making my initial stay in the laboratory very comfortable, and without whose constant cheer and humor, I would not have known the lighter side of things in the laboratory. My hearty thanks to Matias Atria, my close friend and confidant, for his help and insight in answering some of the difficult mathematical questions. I am also thankful to Marcela Aliaga, wife of Matias, for being a dear friend and encouraging me. Also, appreciation to Robert Andrew Hicks (my second officemate) and Dave Pettey, my good friends, for their help and guidance in mathematics and fun conversations. Thomas Sugar was a very enthusiastic person and I must say, I enjoyed his ever strong opinions on political issues and leading the conversations, inadvertently in all lunch sessions. I enjoyed the 3pm coffee outings with Elli Angelopoulou and Farrukh Azfar who were always willing to come for a quick break. My house mate, Arijitt Borthakur was a very nice person that I came to know during my stay with him. Deepak Tolani, one of my best friends, was always around to listen to my rumblings in times of need. Many thanks to Elizabeth Lai, Betsy Norman, Craig Reynolds, Jeffrey Mendelsohn, Faik Goktas, Raymond McKendall, Peter Kraus, Gerda Kamberova, Hong Zhang, Peng Song and many others who made my stay enjoyable.

My parents, Hansaben and Prataprai Desai, gave me the love, education, skills, and constant support that enabled me to do this work. My wife Anisha provided the love, understanding and support that I needed most during the course of my dissertation. Her optimism and enthusiasm helped me sail through some of the challenging times in my thesis. I am indebted to my sisters, Radha and Anjana, who have always been my very good friends and confidantes. David Rutherford, a very close friend and my father away from home, introduced me to this country and the ways and norms around which served as a valuable guidance for my stay here. Other family members whose encouragement and enthusiasm was especially meaningful included Parimalben and Babubhai Naik, Rohitbhai Desai and Parimalbhai Desai. Finally, I am thankful to Judith and William Hollaender for being very good friends and helping me sail through the very last days of my thesis submission and defense.

Above all, I am deeply grateful to the Almighty for taking me through the whole process and giving me the courage and commitment to achieve this goal.

Abstract

MOTION PLANNING AND CONTROL OF COOPERATIVE ROBOTIC SYSTEMS

Jaydev P. Desai
Vijay Kumar
James P. Ostrowski

This thesis addresses the problem of motion planning for cooperative robotic systems. The problem of motion planning for a robotic system is stated as:

Given initial positions and orientations and goal positions and orientations for a collection, $\mathcal{C}$, of robots, in workspace, $\mathcal{W}$, generate a continuous trajectory for $\mathcal{C}$ avoiding contact with the obstacles, $\mathcal{O}_i$, subject to various dynamical constraints of the system.

Because robots are physical systems subject to continuous laws of motion and driven by continuous actuators, we formulate the motion planning problem as an unconstrained variational problem using tools from optimal control and calculus of variations in the first part of the thesis. We develop a general framework for solving motion planning problems involving equality and inequality constraints.

In the second part of the thesis, we study planar human manipulation and develop a computational model for friction-assisted dual arm manipulation tasks incorporating the dynamics of the musculo-skeletal system. We show that our computational model predicts the force distribution and object trajectory in voluntary, relaxed movements. We further study similar tasks in the vertical plane and our experimental findings suggest that there is a great degree of repeatability in trajectories and velocity profiles across trials and subjects.

In the third part, we focus on extending this computational model to plan and control cooperative robotic systems. We solve the dynamic motion planning problem for a system of cooperating robots in the presence of geometric and kinematic constraints, and test the resulting open-loop trajectories on the experimental testbed.

In the last part of the thesis, we explore the application of the motion planning algorithms when the number of robots in $\mathcal{C}$ is very large. Because of increased computational time, we use optimal sensor-based closed loop motion plans. These are combined with the framework of graph theory and optimal control to guarantee provable measure on the performance of the entire system.

The main contributions of the thesis are: (a) studying trajectory generation and force distribution in human dual arm manipulation; and (b) a set of motion planning algorithms for cooperating robot systems subject to dynamic constraints.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

For centuries people have built mechanisms that imitate human characteristics and have some form of automation. The ancient Egyptians attached mechanical arms to their gods which were operated by priests to convey that they were acting under the inspiration from god. The Greeks built hydraulically operated statues to test the principles of hydraulics. Mechanical puppets built in the eighteenth century in Europe made convincing imitation of humans and animals. Over the years, these mechanical innovations combined with the computing power and advances in technology has translated to the development of robotics. Robotics as we know today, is a huge interdisciplinary field comprised of various topics like vision, sensing, dynamics, motion planning and control, locomotion, and design. Of all the above areas, one of the most basic problems in robotics is planning the motion of a robot for a specified task and controlling it to follow a desired trajectory. Hence the problem of robot motion planning deals with finding a feasible trajectory for a robot moving in an environment with obstacles.

A single robot can manipulate objects only as far as it can reach. Most industrial robots are fixed in place thereby limiting their flexibility for tasks requiring manipulation beyond their workspace. One of the examples of such a scenario is the car industry. Here, for tasks requiring welding, painting, and sealing, the car chassis is moved from one cell to another using automated guided vehicles (AGV's). Since the guidance mechanism is pre-installed, it is expensive to alter it. A solution to this problem is to mount manipulators on mobile robots.

According to Latombe [76], the basic motion planning problem for a robotic system is as follows:

Given an initial position and orientation and a goal position and orientation of robot,

$\mathcal{R}$ in workspace, $\mathcal{W}$, generate a path, σ specifying a continuous sequence of positions and orientations of $\mathcal{R}$ avoiding contact with the obstacles, $\mathcal{O}_i$, starting at the initial position and orientation, and terminating at the goal position and orientation. Report failure if no such path exists.

However, some of the most general problems in motion planning with practical applications (for example, the car industry) require a motion planner which can also incorporate dynamic constraints like constraints on velocities, joint torques and forces. Thus, our definition of the motion planning problem is somewhat broader in scope:

Given a set of initial positions and orientations, $\mathcal{I}$, and a set of goal positions and orientations, $\mathcal{G}$, of a collection, $\mathcal{C}$, of robots, in workspace, $\mathcal{W}$, and a set of dynamic constraints, $\mathcal{D}$, generate a path, σ, specifying a continuous sequence of positions and orientations for all robots in $\mathcal{C}$ avoiding contact with the obstacles, $\mathcal{O}_i$, and satisfying the constraints in $\mathcal{D}$, starting at I and terminating at $\mathcal{G}$.

Cooperative robotic systems appear at first glance to be more troublesome than their usefulness. While we can design one robot to carry out every envisionable task, it is more productive to have a team of robots distributed in the environment for carrying out preassigned tasks. This follows from the classical “divide and conquer” strategy. Such a strategy also greatly simplifies the complexity of each robot, as they are designed to do specific tasks rather than all of them. Such robotic systems can have a wide range of applications, ranging from actual physical coupling through objects transported in the environment to loosely coupled systems like a team of robots moving in a formation.

While the above definition of motion planning is very broad and there is extensive literature on this subject as discussed in later chapters, our focus in this thesis is on a special class of problems. Firstly, we focus on solving the motion planning problem described above with the goal of optimizing a suitable cost function. The choice of the cost function greatly depends on the type of task that is being carried out. Thus, for example, in the case of robot motion planning, if we are interested in finding a smooth trajectory for the kinematic motion planning problem, it is meaningful to minimize the norm of the velocity of the mobile robot. However, most robotic tasks also have dynamic considerations and hence it is important to have the dependence of dynamics of the system in the cost function, otherwise the problem is singular. Secondly, if we desire smooth velocity profiles[1] it is reasonable to minimize the norm of the actuator forces/torques. Thus for most dynamic

[1] Some robotic systems like the TRC-Labmates have wheel angular velocities as control inputs and hence smooth velocity profiles lead to lower slippage and dead-reckoning errors.

motion planning problems such as the one we defined above, it is reasonable to minimize the norm of the actuator forces, if we desire smooth velocity profiles or the rate of change of actuator forces, for smooth force trajectories of the manipulator. Normally smoothness in the force profile is desirable to avoid step changes in the current flowing through the motor coil which generates the required forces. Secondly, we are more interested in cooperating mobile robot systems. The behavior of such systems is governed by kinematic and dynamic constraints, and redundancies in actuation.

As discussed above, the goal of the thesis is to study the motion planning problem in cooperating robot systems. We are motivated by tasks that might be required of a system of autonomous robots such as the one shown in the schematic in Figure 1.1. Multiple robots are organized into a small team to pick up an object and move it from one location to another. The system of robots is subject to many constraints like, (a) There are many obstacles in the environment, (b) There are many ways in which the robots can be organized and the load can be distributed, and (c) There are many possible paths. It may be desirable to find a solution to the problem that requires, for example, the smallest amount of force or involves the shortest distance. Alternatively, because the robust control of a system with rapidly changing contact forces is known to be difficult, the smoothest path in which the contact forces or actuator forces exhibit minimal variation might be of interest. We are motivated by such tasks and the need to plan the optimal motion for such tasks.

In order to better understand the robot motion planning problem, we study human performance with the goal of understanding how humans plan and control motion. The main motivation for this work comes from the close parallel between biological and robotic systems. Biological systems provide us with proofs of concept of optimal systems and performance. However, we have a very poor understanding of these systems. On the other hand, we have a reasonably good understanding of the design and control issues in robotic systems. Unfortunately, the present day robot systems are far from optimal. Thus by studying the two types of systems in parallel we can hope to improve our understanding of the fundamental aspects of human motion planning and control, while being able to design better robot control algorithms.

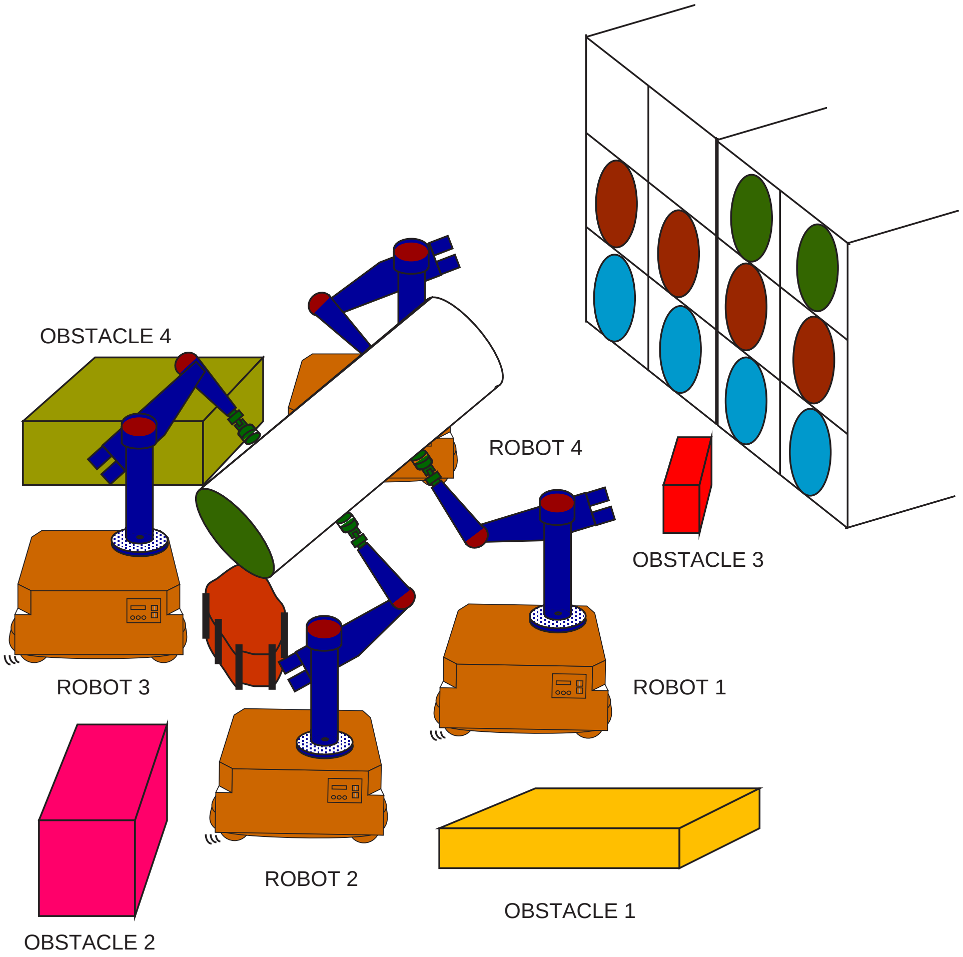

Figure 1.1: Coordination of a multi-robot system in an unstructured environment.

1.1 Approach

There is a vast body of literature for motion planning of holonomic and nonholonomic robots. We will survey them in Chapter 2. While, motion planning can be broadly classified into online and offline strategies, there are problems which rely on a combination of these approaches for planning a trajectory from one position and orientation to another. Some online strategies rely on a sensor based approach like sonar or vision with some hierarchical rules for planning the motion. Such schemes are rarely concerned about optimality issues in finding a feasible path. These methods use knowledge of the current state of the system and its environment to plan the motion for next time instant. Since these strategies do not require any pre-computation, they are suitable for tasks which do not require any specific constraints to be satisfied along the trajectory, *i.e.*, for tasks whose only requirement is a desired goal configuration.

Offline strategies on the other hand, compute motion plans based on the global properties of the workspace. Their output is a set of open-loop trajectories which can be implemented by the system. These methods also take into account the specific constraints that need to be satisfied at every stage of the manipulation task. Offline methods are very well suited for controls since, the controller can be designed to stabilize the motion of the robot along the nominal trajectory, which is the output of the motion planner. However, offline schemes have certain disadvantages like, their inability to deal with uncertainties that can appear during the course of task execution. Another disadvantage is that they are computationally more expensive as the trajectories have to be pre-computed before the onset of motion.

If the goal of building robot manipulators is to mimic human behavior and also the ease and dexterity of human movements, it is reasonable to study motion planning in humans. There is a significant amount of literature which discusses the trajectory planning problem in humans. Humans are highly redundant and yet we are able to perform manipulation tasks with great ease and speed. A simple task of lifting an object or a “handshake”, though effortlessly carried out by humans, seems an enormously complicated problem when we try to understand the underlying control strategies used by the central nervous system (CNS). Since humans seem to be highly adept at manipulation tasks, the goal of this work is to first understand how humans carry out point to point movements. Later, we construct a mathematical model to capture some of the observed behavior.

Most of the studies on understanding human motion planning has been done in the planar single arm or two arm reaching tasks. The minimum jerk criterion (jerk is defined as the third derivative of the position variable with respect to time) suggests that motion planning in humans occurs in task space while the equilibrium trajectory hypothesis suggests that a motion plan is computed at the kinematic level and subsequently transformed to the actuator level. The minimum torque change criterion naturally resolves the indeterminacies as the cost functional explicitly models the dynamics of the system. Thus it gives explicit joint space trajectories which can be transformed through forward kinematics to the task space trajectory. After comparing the results from our computational model and the observed experimental data, we explore a similar methodology to study the cooperative robot motion planning problem in the presence of obstacles.

In the following work, we initially develop offline strategies for motion planning and seek to find locally optimal solutions to motion plans, both in human motion and cooperating robots. Since the extension of this methodology to control a larger ($n > 2$, where n is the number of robots) formation of robots is computationally expensive, we develop strategies which use a blend of online and offline schemes for motion planning. Here we use tools from graph theory to model and control a formation of robots and change formations in the presence of obstacles.

1.2 Statement of the Problem

The main goal of the thesis is to study the problem of motion planning for a system of cooperating robots: *Given initial positions and orientations and goal positions and orientations for a collection, $\mathcal{C}$ of robots, in workspace, $\mathcal{W}$, generate a continuous trajectory for $\mathcal{C}$ avoiding contact with the obstacles, $\mathcal{O}_i$, subject to various dynamical constraints of the system*. Because robots are physical systems subject to continuous laws of motion and driven by continuous actuators, we formulate the motion planning problem as an unconstrained variational problem using tools from optimal control and calculus of variations. We develop a numerical method to solve the continuous motion planning problem for systems subject to dynamical constraints, nonholonomic constraints, and inequality constraints engended by frictional contacts, saturation of actuators, and obstacles. We present many examples of biological and robotic systems to illustrate the application of the methods. These examples include: (a) motion planning for cooperating mobile manipulators; (b) modeling of trajectory generation in human dual-arm cooperative manipulation tasks; and (c) optimal control of locomotion in the snakeboard.

1.3 Contribution

This thesis makes three significant contributions in the area of cooperative robot motion planning. These are summarized below:

- In the first part of the thesis, we propose a general methodology for solving optimal control problems in an unconstrained variational calculus framework. We then study the problem of human motion planning in horizontal and vertical plane in great detail. Here, we develop a mathematical model for motion planning in humans which matches some of the experimentally observed kinematic and dynamic trajectories in planar manipulation tasks.

- In the second part, we focus on extending the general method and studies on human motion planning to cooperative robotic systems. This system has two cooperating robots with manipulators mounted on them grasping an object. We have solved the dynamic motion planning problem for such a system in the presence of obstacles and other kinematic constraints.

- In the last part of the thesis, we focus on extending the methodology of Chapter 5 to a formation of many robots using a combination of centralized and decentralized control. The advantage of this approach is significant reduction in computational time and ease of incorporating uncertainties in the environment. We guarantee the shape of a formation while the robots move from one configuration to another and develop an algorithm for changing formation.

1.4 Organization

In Chapter 2, we review work done in the area of human and robot motion planning. We discuss various methodologies and models proposed in psychophysics and the cognitive sciences that attempt to explain the basic problems related to human point to point movements. We also relate these to problems in robot motion planning and discuss the advantages and disadvantages of various approaches. In Chapter 3, we propose a general mathematical model in the framework of calculus of variations to solve the optimal control problem. We specialize the modeling technique and the numerical method to the existing problem to obtain faster and accurate solutions.

In Chapter 4, we discuss the experimental studies carried out with human subjects for point to point movements in the horizontal and vertical plane. We compare our experimental findings with the results of our computational model for planar motions. We also present the explained and unexplained observations with a mathematical model for planar tasks. We begin by analyzing experimental data collected from our experimental setup. Since the problem of cooperative manipulation of a grasped object by humans is similar in nature to that of two robot manipulators grasping an object, we first try to understand the basic human motion planning problem. We develop a novel numerical method that accommodates both equality and inequality constraint in a unified framework. The mathematical model is formulated in the framework of calculus of variations. We develop our computational model using this approach and it explains some of the observed behavior.

In Chapter 5, we demonstrate that a similar mathematical model can be used to plan the motion of cooperating nonholonomic robots in an environment with obstacles. We only plan the motions of the base TRC-Labmates as the problem of coordination and control of robot manipulators on these platforms has been studied in [131]. If we use the same strategy to solve the motion planning problem for a larger number of robots, the task will be computationally burdensome though centralized motion planning methodology has a great advantage of developing explicit motion planning schemes.

In Chapter 6, we investigate the problem of controlling a large formation of robots using a hybrid of centralized and decentralized motion planning strategies. We assume that the lead robot plans an optimal trajectory and that is used to plan the trajectory of the other follower robots in the formation. To overcome the obstacles in the environment, these formations need to change their shape. In order to accomplish this, we propose a new methodology which uses some of the tools from graph theory to solve the problem. We finally make some concluding remarks in Chapter 7.

Chapter 2

Background

2.1 Human motion planning

The problem of modeling or trajectory generation in humans is difficult since we do not have adequate information about task executions by the central nervous system (CNS). There is a significant amount of literature in the area of single arm manipulation. Most of the research in this area has focussed on, kinematics of multi-joint movements, coordination of hands and fingers for grasping and manipulation tasks and developing mathematical models to explain the observed experimental data. We will investigate each of these areas in greater detail.

2.1.1 Kinematics of multi-joint movements

Some of the earlier works in this area stems from studies of single arm movements in the horizontal or spatial domain. Morasso [91] studied spatial control of arm movements and formulated the hypothesis that the central commands for point to point movements are formulated in terms of the trajectories of the hand in space rather than joint variables. Viviani and Terzuolo [124, 125] gave a detailed kinematic description of hand motion during handwriting and showed that the basic kinematic variables, *i.e.*, tangential velocity and the curvature of trajectory, preserve their temporal pattern when the position and size of movement is changed. Soechting and Lacquaniti [113] studied invariant characteristics for a simple task of reaching forward towards a target and concluded that the trajectory in space was independent of the movement speed. Secondly, they hypothesized that movements can be viewed to consist of two phases, an acceleratory phase and a deceleratory phase. The acceleratory phase is so organized as to maintain an invariant ratio of elbow and shoulder angular velocity with respect to target location in the deceleratory phase. Further, Soechting and Lacquaniti [114] studied the effect of sudden changes in target location to kinematic trajectories of the arm and concluded that the coordination of arms and shoulders in such tasks was achieved by a reduction in the number of degrees of freedom of the movement.

A detailed account and model of a kinematic chain for the asymmetric behavior of the human arms in a manipulation task is proposed by Guiard [51]. Atkeson and Hollerbach [6] carried out a detailed study of kinematic features for unrestrained vertical arm movements. They concluded that though the velocity profiles were bell-shaped, there was a distinct difference in the curvature of the trajectories for upward and downward movements. Downward movements in some subjects were observed to be straight while the upward movements showed curvature. They also concluded that all movements, whether curved or straight, with heavier or lighter objects, showed an invariant tangential velocity profile when normalized. However, the curvature of trajectories for unrestricted vertical arm movements did contradict the model of Flash and Hogan [44].

2.1.2 Coordination of hands and fingers for grasping and manipulation tasks

The coordinated manipulation with fingers has also been extensively studied in the robotics literature. Johansson and Westling [65, 66] have studied the coordination of grip and loading force when small objects are lifted, positioned and replaced by human subjects. They showed that the ratio of the grip force to load force does not vary appreciably with load. They also investigated the adaptation of motor commands to variations in the weight of the object during various phases of the task. The target and probing strategies used by humans suggest that the haptic information may be used to convey the size information which can be combined with the sensory information in programming a precision grip [48]. It is well-known that the grip force is adapted to the friction between the skin and the object. This adaptation depends on the cutaneous afferent input [65]. The relative importance of various sources of sensory feedback has been studied by Srinivasan and Chen [116]. However, we are more interested in the plans for grasping and manipulation, *i.e.*, the feed forward component of the control system.

Sensor driven models for grasping are also discussed in [102, 116]. Reinkensmeyer *et al.* [105] studied how humans control fast, planar, single-degree-of-freedom, wrist movements using a simple two-hand grasp. A similar study was also been done by Lum *et al.* [84] for feedforward stabilization in a bimanual unloading task. Their studies revealed that the load-sharing in the hand-load-hand system is asymmetric. One hand chiefly accelerates while the other brakes the system during the task. Their studies also showed that humans exert a higher internal force than required to carry out the manipulation task. The dependence of internal forces on external disturbances has been studied by Johansson *et al.* [64]. They observed that the grip force in a pinch grip rises about 70 milliseconds after the onset of a ramp increase in load.

Flanagan and Tresilian [42] further explored the relationship between the grip force and the load force for different grips. Their most interesting observation is the strong correlation between the grip force and the acceleration (and therefore the load forces) of the grasped object. They proposed a model for the coordinated control of grasp and reach, in which the grasp controller is strongly coupled to the reach controller. This is in contrast to the paradigm generally used in robotics in which the reaching motion and the grasp forces are controlled independently [92].

Hoff and Arbib [57], presented a model describing the kinematics of hand transport under a variety of circumstances, including target perturbation. They developed a model for temporal coordination of reaching and grasping tasks. They also provided an optimization principle for hand preshaping in grasping tasks. Though these models study finger or hand manipulation in the horizontal or vertical plane, they do not explicitly address the problem of two-arm manipulation considering the dynamics of the entire system for large scale motion. We investigate this problem and describe how the computational model accounts for some of the observed experimental data. This will improve our understanding of human manipulation and aid in designing better robot control algorithms.

2.1.3 Mathematical models

Mathematical modeling is important to understand the dynamics, control and planning process underlying a manipulation task. The planning phase of a manipulation task may be classified into two independent sub-phases [137]. First, the *trajectory generation problem* involves the computation of the best trajectory for the object. Given this trajectory, the second problem is to determine the *force distribution*, or the distribution of the load forces between the different limbs, that will produce the desired trajectory. Trajectory generation is generally formulated as an optimization problem (for example, shortest distance [100], minimum acceleration [94], or minimum time [112]). For a given optimal trajectory, the indeterminacy (called actuator redundancy) in the force distribution problem, is resolved by minimizing a suitable cost function that usually involves some measure of the internal forces [74, 129, 142], or by imposing additional constraints on the grasp forces [99, 134].

Although the optimization criteria or constraints may be meaningful for robot systems, it is not clear if they are useful for modeling human behavior.

An alternative approach is to consider trajectory generation and force distribution problems concurrently. The results of a two-stage approach to planning motions and forces are generally inferior to globally optimal solutions that are obtained by solving for the best trajectory and the best force distribution concurrently [137, 138]. The dynamics of frictional, open-palm grasps has been studied in robotics. Yun [133] studied the control of grasping forces in friction-assisted pushing tasks with a prespecified trajectory. Erdmann [39] addressed planning of manipulation tasks with open-palm grasps in a quasi-static framework. Lynch and Mason [87] studied the control of single arm, planar grasps. The dynamics of multi-limbed frictional grasps are addressed by Murray *et al.* [92]. However, they do not address the planning of trajectories and grasp forces.

Many previous studies have investigated mechanisms that might underlie the generation of single arm trajectories in humans. These include studies on the movement of a single joint [58], and multi-joint reaching motions in the horizontal [44, 91, 121] and vertical [6, 113] planes. Questions relating to the generation of more complex motions are also addressed: for example, a modified hand [43] and eye [53] trajectories. Modeling multi-joint systems has been explored in the work of McIntyre [90]. This model for multi-joint motor systems takes into account the non-linear features of the motor system and is used to represent mechanical interactions with the environment. This can be used to derive the control inputs for a wide variety of motor tasks.

Flash and Hogan [44, 59] studied single arm reaching tasks and suggested that the central nervous system uses an optimality criterion to calculate the trajectory along which to move. In medium speed, large amplitude, unconstrained motions in the horizontal plane for a single arm, the integral of the jerk along the trajectory is minimized. Their model thus involved minimizing the norm of jerk in planar arm movements, *i.e.*,

$$J = min\frac{1}{2}\int_{t_o}^{t_f}(\dddot{x}^2 + \dddot{y}^2)dt$$

where t_o and t_f are the initial and final times and (x, y) represent the coordinates of the end-effector of the manipulandum in the plane. The *minimum-jerk* solution, as it is known in the literature, depends only on the kinematics of the task and is independent of the physical structure and the dynamics of the arm. The solution to the minimum-jerk problem is determined by the boundary conditions at the beginning and termination of movement. Once this information is given, we can uniquely specify the hand trajectory. Thus for example, in the case of zero starting and ending velocities and accelerations, the trajectory is a fifth order polynomial and is given by:

$$\begin{aligned} x(t) &= x_0 + (x_0 - x_f)(15\hat{t}^4 - 6\hat{t}^5 - 10\hat{t}^3) \\ y(t) &= y_0 + (y_0 - y_f)(15\hat{t}^4 - 6\hat{t}^5 - 10\hat{t}^3) \end{aligned}$$

where, t_f is the final time, $\hat{t} = t/t_f$ and (x_0, y_0) are the initial coordinates of the hand at $t = 0$. The final coordinates of the hand are given by (x_f, y_f) at $t = t_f$. According to studies of Garvin *et al.* [45] on coordinated manipulation with two arms by humans, it appears that the minimum-jerk criterion may not adequately explain all aspects of the observed kinematic data in bimanual tasks.

An alternative model that has been employed for single arm reaching tasks in the plane is the so-called minimum-torque-change criterion, first proposed by Uno *et al.* [121], where the task is to minimize the rate of change of actuator torques:

$$J = min\frac{1}{2}\int_{t_o}^{t_f} \|\dot{\tau}\|^2 dt$$

where, t_o and t_f are the starting and ending times of motion and τ represents the vector of actuator torques. The general form of the dynamical equations of motion are given by:

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) = \tau$$

where M represents the inertia of the object, θ represents the generalized coordinates, $C(\theta, \dot{\theta})$ is the coriolis force and τ is the vector of generalized torques. We will use some of these ideas to model the closed kinematic chain for coordinated two arm manipulation tasks by humans in Chapter 4. This model suggests that the planned trajectories and forces are such that they minimize the integral norm of the vector of time rates of change of actuator or muscle forces [121]. Unlike the minimum-jerk model, the minimum-torque-change model also predicts the force distribution in the system [138]. Further, as will be shown later, it appears to explain the kinematics and dynamics of bimanual reaching movements in which the asymmetry of the two arms does not play a significant role [30, 34]. In the current study for modeling friction assisted planar manipulation tasks, we use the minimum torque change model as it predicts some of the observed features of experimental data.

2.2 Cooperative robot motion planning

The area of motion planning has received great attention recently. The motion planning problem involves finding a feasible path from an initial configuration to the final configuration while avoiding collisions with obstacles subject to kinematic and dynamic constraints that may possibly be present in the system/environment. Traditionally, the motion planning problem defined by Latombe, involves finding a feasible path (if there exists one) for a single robot subject to geometric constraints. We have extended this to finding a solution for a more general problem where we also need to satisfy kinematic and dynamic constraints like constraints on the turning radii of the mobile platforms, bounds on the actuator forces and so on. The task becomes much more complicated when we try to find solutions for a system of cooperating robots navigating in an environment with obstacles. This is due to the fact that apart from solving the basic robot motion planning problem defined in Chapter 1, they also need to satisfy other constraints such as contact forces on the grasped object, frictional constraints, etc.

The notion of representing the robot in a transformed space where the robot is a point was first introduced by Udapa [120] though he used only rough approximations to the actual configuration space obstacles or C-space obstacles. The idea of a configuration space obstacle or C-space obstacle is to map the obstacle from the workspace, $\mathcal{W}$, to the configuration space of the robot. This transforms the motion planning problem for a robot with physical dimensions in the workspace, $\mathcal{W}$, to one in which the robot is a point. This notion was formalized for arbitrary degrees of freedom of a robot by Lozano-Perez [82]. With the aid of this formulation, a robot is a point in the configuration space and the forbidden configurations are represented as C-space obstacles. An example of a C-space obstacle is as shown in Figure 2.1. Figure 2.1(a) shows the top view of the envelope of the robot moving around the obstacle at a fixed orientation while Figure 2.1(b) shows the plot of the actual C-space obstacle for various orientations of the robot from 0 to 360 degrees.

Hence, by computing the C-space obstacles, we can construct algorithms for computing the motion plans for a single point in this space. Though this seems to have reduced the computational complexity of the problem by considering the motion planning of a single point, it is however not that straight forward. This is due to increased computational complexity in computing arbitrary C-space obstacles for non-polyhederal obstacles. For example, the problem of computing the configuration space for a robot with polygonal shaped links and polygonal obstacles is a PSPACE hard problem [76].[1] Complexity issues for holonomic systems have been very well understood. In such problems the existence of a path is characterized by the notion of "connectedness" of the configuration space, *i.e.*, its topology. Schwartz and Sharir [110, 111] studied precisely this problem. Since then much work has been done both in theory and practical implementation of motion planning

[1] A PSPACE hard problem is one which requires polynomial order memory allocation for solving the problem.

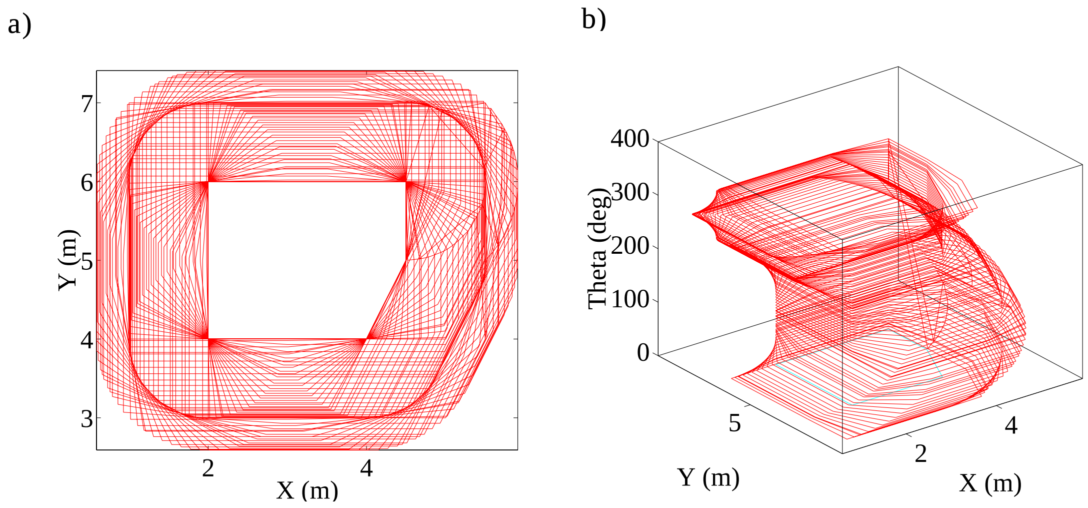

Figure 2.1: C-space obstacle for a pentagonal obstacle and square robot: a) Top view, b) Actual C-space obstacle.

algorithms for holonomic systems [11, 19, 40, 49, 83]. Though the area of nonholonomic motion planning has received significant attention in the recent past, few people have investigated optimality issues in motion planning for nonholonomic systems and of those, few have again implemented practical planners.

Latombe [76] provides a very nice classification of the motion planning methodologies and emphasizes three important methodologies in this area: *road-map*, *cell decomposition* and the *potential field* approach. We will now briefly describe each of these approaches below.

- **Roadmap** is defined as the network of one dimensional curves which captures the free space of the robot. Once these roadmaps are constructed the next task is to find a suitable path between the initial and final points. Variations of this method are the *visibility graph*, *Voronoi diagram*, *freeway net* and *silhouette*.

- **Cell decomposition** is probably one of the most popular methods for motion planning. The aim is to discretize the free space of the robot into *cells* so that the path between the cells can be easily generated. The method uses a non-directed graph as the underlying tool for finding a feasible path. The problem eventually reduces to a findpath problem on a graph.

- **Potential field** approach is probably one of the most popular methods for online motion planning. This method involves construction of artificial potentials which can be attractive or repulsive depending on whether it is a goal or an obstacle in the environment. Thus at each instant, the robot moves in the direction of the negative of the gradient of the potential which is analogous to an artificial force applied to the robot. Though the idea of potential fields first appeared in the dissertation of Khatib [69], it was formally used in the context of motion planning in [70]. Koditschek [71] used the Poincaré-Hopf Index theorem to prove that a global navigation function does not exist in general. More generally, in R^2 if there are q obstacles homeomorphic to a closed unit disc then the potential function, U must have at least q saddle points. He proposed that one possible way to overcome this difficulty is to construct potential functions with their minimum located at the goal and whose domain of attraction is the entire free space of the robot except possibly finitely many points of measure zero. These are the saddle points of the potential function. A generalization of this approach to account for arbitrary star shaped obstacles was developed by Rimon and Koditschek [106], where they constructed such "almost global navigation functions" by successive smooth deformations of the obstacles into spheres and carrying out the motion planning problem in the sphere world.

Though the above approaches work very well in some situations, they are rather impractical methods when it involves motion planning for a nonholonomic robot with additional inequality constraints imposed by, for example, bounds on the minimum and maximum separation between the two robots as shown in the experimental test-bed (Figure 5.2). Secondly, in the case of potential field approaches, constructing the navigating functions is also not an easy task.

Since we deal with continuous systems, we are interested in solving problems of motion planning which yield continuous solutions. Buckley [22] addressed continuum methods for motion planning. He defined a *non-intersection metric*, and its directional derivatives in the configuration space to avoid obstacles. Buckley addressed the minimization of an integral cost function using the calculus of variations. The resulting two-point boundary value problem was solved for many planar motion planning problems. However, he did not consider either kinematic or dynamic constraints.

Initial results for optimal control problems with inequality constraints appeared in the works of Bryson *et al.* [20]. The method is a modification to the extremal solution of the problem of Bolza. They required that the constraints not having an explicit dependence on the input, satisfy certain corner conditions at the entry point of a constraint boundary. These constraints were essentially lower order derivatives of the constraint. Speyer and Bryson [115] further proved that these conditions must also be satisfied at the exit point from a constraint boundary if we use the direct adjoining approach. Jacobsen and Lele [62] combined the previous conditions and defined the highest derivative of the slack variable as a new control input to the system. Thus, there were no discontinuities in the adjoint variables for such constraints. These results are also summarized in [21]. Jacobsen *et al.* [63] further studied the nature of solutions by Bryson *et al.* [20] and showed that in some cases, the conditions of Bryson can lead to non-stationary trajectories. Baillieul [7] developed a new approach for optimal control problems with nonlinear state evolution equations. He showed that for bilinear systems with a right invariant evolution equation, it is possible to obtain closed form solutions for the control inputs where the Legendre condition is satisfied.

One of the primary objectives of a motion planning algorithm avoiding obstacles, involves an efficient distance computation algorithm. The literature on the computation of distances between objects and the rate of change of the distances with respect to configuration space variables is very relevant to this work. In particular, there are two sets of papers that are worth mentioning. Gilbert and Johnson [46] proposed a distance function for motion planning. The main focus of this paper was on the properties of the distance function and the computation of distances and their gradients. Efficient algorithms for computing distances in R^n are described in [47]. A different approach to computing distances between polyhedra is described by Lin and Canny [81]. It is based on a cleverly designed data structure which facilitates the identification and tracking of a pair of features (vertex, edge or face), one on each polyhedron during the motion. Though the algorithm is linear in the number of vertices, it is not very popular due to the storage requirements of the data structure defining the connectivity between different vertices.

The first fundamental result in motion planning for car-like systems was derived by Reeds and Shepp [104]. Building on the work of Dubins [37], they showed that minimum-distance trajectories for a car are composed of straight lines and circular arcs. Laumond [77] addressed motion planning of a nonholonomic cart in a cluttered environment. The focus was on blending straight line segments with arcs so that the resulting path is free of cusps. Barraquand and Latombe [12] described the generation of collision-free paths for a cart with a trailer that minimized the number of reversals in the path. Jacobs and Canny [61] presented an algorithmic approach to path planning for mobile robots subject to nonholonomic constraints. They defined a set of canonical trajectories which satisfied the nonholonomic constraints and an algorithm that found a path satisfying the constraints imposed by the obstacles. In the area of nonholonomic motion planning, Fernandes *et al.* [41] adopted an optimal controls approach to obtain near optimal solutions by choosing the control input as a linear combination of smooth orthogonal basis functions. However, the search space for such functions is infinite dimensional and their approach considers a finite combination of these basis functions to minimize the residue along with the prescribed cost function. This approach does not work very well with unilateral constraints on the state, because the optimal inputs can become discontinuous at points where the trajectory switches from an unconstrained region to a constrained region. Laumond *et al.* [78] proposed an efficient algorithm for planning near minimum distance trajectories for a car moving among obstacles. Their planning problem is divided into three stages: (a) find a holonomic path that avoids all obstacles; (b) find the nearest path that also satisfies the nonholonomic constraints; and (c) optimize the obtained path. The results in these papers cannot be easily extended to systems in which dynamic considerations are important. Different researchers have studied the nonholonomic motion planning problem from different viewpoints: probabilistic road maps by Kavraki *et al.* [67], differential geometric methods by Sussmann [118] and behavior based planning by Manikonda [88] are some of these examples.

There are many approaches to the generation of open-loop trajectories for nonholonomic systems. Brockett [17] derived the optimal control inputs for canonical systems in which the tangent space to the configuration space manifold is spanned by the input vector fields and their Lie brackets. Murray and Sastry [93] studied a more general class of nonholonomic systems that can be converted to the so-called chained form. Their results are applicable to systems which may require more than one level of Lie brackets to span the tangent space of the configuration space manifold. This work also outlines an algorithmic approach to steering such systems from an initial to final configuration using sinusoidal inputs. Bushnell *et al.* [23] studied the motion planning of three-input wheeled systems by reduction to the chained form. Tilbury *et al.* [119] employed methods of exterior differential calculus to reduce the N-trailer system to the chained form. Other interesting methods are described in Egeland *et al.* [38] and Walsh and Sastry [127]. However, the reduction of general systems to the chained form is by no means a simple process. While this approach yields considerable geometric insight into the nature of inputs and trajectories, the inputs are suboptimal and these methods do not lend themselves to incorporating obstacles and other constraints which can be possibly present in a practical system.

It is useful to think of nonholonomic locomotion systems in terms of a set of shape variables that can be independently actuated and a group of rigid body motions that describes the gross motion of the locomotion system [75, 96]. Koon and Marsden [72] discussed the optimal control of nonholonomic mechanical systems that exhibit group symmetries and derived a set of simplified necessary conditions for the control inputs. Ostrowski and Burdick [96] discussed optimal “gaits” for such systems by investigating different sinusoidal input motions of the shape variables. Again these approaches are not suitable for solving motion planning problems with obstacles and other unilateral constraints.

In recent years, many researchers have investigated control of nonholonomic systems. A car or a car with trailers are typical examples of such systems. Other examples include underwater vehicles [38], satellites [127] and variable truss chains [75]. Brockett [18] showed that no smooth state-feedback control law can asymptotically stabilize a nonholonomic system to a specified configuration even if the system is controllable. However, stabilization can be achieved using non-smooth, autonomous feedback laws [16, 38], time-varying feedback laws [108] or non-smooth, time-varying schemes [26].

The approach adopted here and briefly described in [31, 33, 136], is in many ways superior to others discussed in the literature. We solve the general nonholonomic motion planning problem by formulating it as an optimal control problem with equality and inequality constraints on the state and/or inputs. It is then reformulated as an unconstrained variational calculus problem and the general approach in Gregory and Lin [50] is used to solve the problem. The advantage of this method is that constraints such as nonholonomic constraints simply become equality constraints on the state variables and they can be easily dealt with. In addition, it is easy to include constraints on the state, and in particular, unilateral constraints such as those arising from the boundaries of the configuration space. Once the optimal trajectory is known, there are many methods (see, for example, [109, 126]) to efficiently control the system along the trajectory.

As the number of robots in the formation increases ($n > 2$), the computational cost for planning a feasible motion plan increases significantly and hence it is meaningful to construct better control strategies to carry out a particular task. Many researchers have recently investigated controlling multiple (*i.e.,* many) cooperative robots [2, 5, 73, 101]. This research has primarily focused on generating emergent or complex behaviors from organizations of many small, individual, and often heterogeneous robots. Generally, these approaches either require some type of supervisory (centralized) control and process planning, or rely on more evolutionary approaches to generate a pre-specified system behavior without provable bounds on the performance of the entire system. The goal we pursue here to control a formation of robots relies on the idea of decentralized motion planning. Here, each robot has minimal interaction with the other robots in the formation and it is equipped with the sensing and computational hardware that allow it to detect the distance to the nearest obstacle and the relative position and orientation of the neighboring robots in the team. We attempt to generate rigorously provable measures/bounds on the performance of our system. We formulate the problem of control and coordination for many robots moving in formation using decentralized controllers. In this approach, a motion plan for the overall formation is given, for example, using methods from optimal control or some other external source such as human operator. This motion plan is then used to control a single lead robot. We assume that each robot has the ability to measure the relative position of other robots in the formation. Once the motion for the lead robot is given, the remainder of the formation is governed by local control laws based on the relative dynamics of each of the follower robots and the relative positions of the robots in the formation. These control laws have the advantage of providing easily computable, real-time feedback control, with provable performance for the entire system. An important requirement in controlling a formation of robots is the ability to change the shape of a formation to avoid obstacles. We investigate this problem and develop a strategy for changing formations for a team of robots.

One of the main challenges in the practical implementation of the theoretical algorithms is to be able to obtain information about the environment in real-time and using it to plan subsequent motion. Since our proposed strategy is not limited to the shape and size of obstacles, some of the existing techniques in literature to model the environment can be effectively exploited to carry out the required task. Thus, the formation of robots must be capable of navigating in possibly uncertain environments along with a possible uncertainty in its state estimation. Even if the robot has *a priori* a nominal model of the environment there is normally some uncertainty in the state and the model of the robot/environment. State and modeling uncertainties, for example, can be present in the form of position and modeling of the shape and size of the obstacles. Modeling uncertainty can lead to an increase in the state uncertainty while a reduction in state uncertainty can be used with effective identification techniques to reduce the modeling uncertainty. There is extensive literature on motion planning with uncertainty in the environment. Lumelsky *et al.* [85] presented an algorithm for terrain acquisition without any limitations on the size or shape of the obstacle. The algorithm is useful to construct a model of planar terrains with obstacles of arbitrary shapes. The hierarchical generalized Voronoi graph (HGVG) technique of Choset and Burdick [25] is an alternate strategy for motion planning of a single point robot based on sensory information. Some of the main features of this method are: (a) it uses the line of sight distance information to build the map of the environment and (b) it can be constructed without any constraints on the size and shape of obstacles present in the environment. Another advantage of this method is that it is proven to be complete, *i.e.*, it finds the goal if it is reachable and reports the entire map of the environment in the event of a non-existent path. It will be a challenging task to use some of these techniques to execute online motion planning algorithms developed in Chapter 6.

Chapter 3

Unconstrained reformulation of the optimal control problem

3.1 Introduction

We will model our system of cooperating manipulators as a dynamic system in state space, and the constraints on the system (for example, limits on joint positions and forces, and constraints due to obstacles) will be described as state and input based inequality constraints. We will assume that it is meaningful to minimize an integral cost or performance function. In tasks in which dynamic considerations are not important, it may be useful to minimize distance [104]. A minimum jerk functional has been advocated for modeling reaching by humans [44] for generating smooth robot trajectories [140]. When manipulating objects, it is necessary to maintain force closure while minimizing the contact forces. Therefore, there are constraints on the exerted forces and it is often meaningful to minimize the integral of the norm of the actuator forces. In many systems, this also results in minimizing the energy consumption [112]. If we desire a smooth actuator force history, it makes sense to minimize the integral of the norm of the rate of change of actuator forces [121]. An important benefit of such integral cost functions based on actuator forces is that by minimizing them, we can automatically resolve kinematic and actuator redundancy while generating trajectories that will guarantee a desired level of smoothness in the actuator forces and/or trajectories.

In the following section, we will describe the classical optimal control problem and the solution of Pontryagin for various conditions on the state and control vector. In Section 3.3, we will consider an equivalent of the optimal control problem in variational framework, known as the problem of Bolza. We will present first order necessary conditions for the critical solution to both of these problems in the corresponding sections and discuss the their equivalence. We will then reformulate the optimal control problem as an unconstrained variational problem in Section 3.4. The variational problem is formulated in an integral form, in contrast to the more traditional methods that express the necessary conditions as differential equations [50]. In Section 3.5, we will illustrate the procedure with a simple example and finally discuss the numerical method for solving these problems in Section 3.6.

3.2 The optimal control problem

Consider a dynamic system described by:

$$\dot{x} = f(t, x(t), u(t)) \tag{3.1}$$

where $x = [x_1(t), x_2(t), \cdots, x_n(t)]^T$ is the $n \times 1$ state vector and $u = [u_1(t), \cdots, u_m(t)]^T$ is the $m \times 1$ vector of control inputs. The control vector, $u : \mathbb{R} \to U \subset \mathbb{R}^n$ belongs to an arbitrary set U. The set $\mathcal{U}$ of admissible controls, is one that consists of bounded, piecewise continuous functions. Further we assume that $f : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is sufficiently smooth (at least C^1). The cost functional to be minimized is:

$$J(x, u) = \gamma(t_0, x_0) + \int_{t_0}^{t_f} L\left(t, x(t), u(t)\right)\, dt \tag{3.2}$$

where, t_0 and t_f are the initial and final time and $t \in [t_0, t_f]$. $\gamma : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}$ and $L : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ are also assumed to be sufficiently smooth (at least C^1). In addition, we have the following constraints:

$$g_i(t, x, u) = 0 \quad i = 1, \ldots, l_\phi \tag{3.3}$$

$$h_i(t, x, u) \leq 0 \quad i = 1, \ldots, l_\xi \tag{3.4}$$

The equality and inequality constraints may or may not have an explicit dependence on the input, $u(t)$. The optimal control problem is stated as follows [50]:

Problem 3.2.1 The optimal control problem:
Among all controls, $u(t) \in \mathcal{U}$, find the control $u^*(t)$, and the corresponding state $x^*(t)$, satisfying Equation (3.1), which evolves the state from initial state $x(t_0) = x_0$, to the final state $x(t_f) = x_f$, while minimizing the cost functional, $J(x, u)$ given by Equation (3.2).

The solution to the classical optimal control problem was given by Pontryagin, known as the **Pontryagin's minimum principle**. The solution of Pontryagin's minimum principle can be broken down into three cases depending on whether (x^*, u^*) is on the boundary of the admissible set, (x, u). The three cases are enumerated below:

1. In the first case, $(x^*(t), u^*(t))$ is an interior point for all $t \in [t_0, t_f]$.

2. In the second case, $u^*(t)$ is on the boundary of $\mathcal{U}$ for a subinterval of $[t_0, t_f]$ and $x^*(t)$ is an interior point for all $t \in [t_0, t_f]$.

3. Finally, in the third case, both $x^*(t)$ is on the boundary of the admissible set for all $t \in [t_0, t_f]$ and $u \in U$.

The main theorem's relating to Pontryagin's minimum principle for the three cases described above can be summarized below [50, 55].

Theorem 3.2.2 (Case 1: Pontryagin's minimum principle) *[50] Define the Hamiltonian, $H(t, x, u, \lambda)$ as:*

$$H(t, x, u, \lambda) = L(t, x, u) + \lambda^T f(t, x, u)$$

where, λ is an $n \times 1$ vector of multipliers. Then if, (x^*, u^*) gives a minimum to $J(x, u)$, there exists, $\lambda^*(t)$ such that:

$$\begin{aligned} \dot{x}(t) &= H_\lambda(t, x, u, \lambda) \\ \dot{\lambda}(t) &= -H_x(t, x, u, \lambda) \\ H_u(t, x, u, \lambda) &= 0 \end{aligned} \tag{3.5}$$

along $(t, x^*(t), u^*(t), \lambda^*(t))$ for all $t \in [t_0, t_f]$ satisfying the following boundary conditions:

$$x(t_0) = x_0 \quad \textit{and} \quad x(t_f) = x_f$$

In the second theorem, we assume that $x^*(t)$ is an interior point for all $t \in [t_0, t_f]$ in the admissible set and $u^*(t)$ may lie on the boundary of the admissible set, $\mathcal{U}$ for some interval of $[t_0, t_f]$. This can happen when we have a bounded control set such that, $\mathcal{U} = \{u : |u| \leq c_1\}$ where $c_1 > 0$.

Theorem 3.2.3 (Case 2: Pontryagin's minimum principle) *[50] Define the Hamiltonian, $H(t, x, u, \lambda)$ as:*

$$H(t, x, u, \lambda) = L(t, x, u) + \lambda^T f(t, x, u)$$

where, λ is an $n \times 1$ vector of multipliers. Then if, (x^*, u^*) gives a minimum to $J(x, u)$, there exists, $\lambda^*(t)$ such that:

$$\begin{aligned} \dot{x}(t) &= H_\lambda(t, x, u, \lambda) \\ \dot{\lambda}(t) &= -H_x(t, x, u, \lambda) \\ H(t, x^*(t), u^*(t), \lambda^*(t)) &\leq H(t, x^*(t), u(t), \lambda^*(t)) \end{aligned} \tag{3.6}$$

for $u(t) \in U$ for all $t \in [t_0, t_f]$ satisfying the following boundary conditions:

$$x(t_0) = x_0 \quad \textit{and} \quad x(t_f) = x_f$$

Finally, when the optimal control problem involves equality constraints, the state vector, $x^*(t)$ can lie on the boundary of the constrained set. These constraints can be of the following form:

$$g_i(t, x) = 0 \quad i = 1, \ldots, q \tag{3.7}$$

Thus the most general form of the Pontryagin minimum principle gives the following solution for the optimal control problem.

Theorem 3.2.4 (Case 3: Pontryagin's minimum principle) *[50] Define the Hamiltonian, $H(t, x, u, \lambda)$ as:*

$$H(t, x, u, \lambda) = L(t, x, u) + \lambda^T f(t, x, u)$$

where, λ is an $n \times 1$ vector of multipliers. Then if, (x^*, u^*) gives a minimum to $J(x, u)$, there exists, $\lambda^*(t)$ such that:

$$\begin{aligned} \dot{x}(t) &= H_\lambda(t, x, u, \lambda) \\ \dot{\lambda}(t) &= -H_x(t, x, u, \lambda) + \phi^T Q_x \\ H(t, x^*(t), u^*(t), \lambda^*(t)) &\leq H(t, x^*(t), u(t), \lambda^*(t)) \end{aligned} \tag{3.8}$$

where $Q = [Q_1, Q_2, \cdots, Q_q]^T$ and

$$Q_i = \sum_{j=1}^{n} \frac{\partial g_i}{\partial x_j} f(t, x, u) \quad i = 1, 2, \cdots, q$$

for $u(t) \in U$ for all $t \in [t_0, t_f]$ satisfying the following boundary conditions:

$$x(t_0) = x_0 \quad \textit{and} \quad x(t_f) = x_f$$

Clearly, when $q = 0$, the above theorem reduces to Theorem 3.2.2 or Theorem 3.2.3. Clearly, both Theorems 3.2.3 and 3.2.4 cannot be easily solved numerically, particularly due to the inequality in $H(t, x, u, \lambda)$. Also the classical techniques do not incorporate equality constraints having state dependence as in Equation (3.3) and inequality constraints having either state or control input dependence as in Equation (3.4). A general procedure for solving these problems is to rewrite the Hamiltonian incorporating the the equality and inequality constraints. This can be written as [21]:

Theorem 3.2.5 *Define the Hamiltonian, H as:*

$$H(t,x,u,\lambda,\phi,\psi) = L(t,x,u) + \lambda^T f(t,x,u) + \phi^T g(t,x,u) + \psi^T h(t,x,u) \tag{3.9}$$

where ϕ and ψ are $l_\phi \times 1$ and $l_\xi \times 1$ vectors. Thus if the optimal control problem has a solution, $(x^*(t), u^*(t))$ then there exist nonzero multipliers, $\lambda^*(t)$ which are solutions of the equation:

$$\dot{\lambda}(t) = -H_x(t,x,u,\lambda,\phi,\psi)$$

and the optimal inputs satisfy:

$$\frac{\partial H(t,x,u,\lambda,\phi,\psi)}{\partial u} = \frac{\partial L(t,x,u)}{\partial u} + \lambda^T \frac{\partial f(t,x,u)}{\partial u} + \phi^T \frac{\partial g(t,x,u)}{\partial u} + \psi^T \frac{\partial h(t,x,u)}{\partial u} = 0$$

Also the multipliers, $\psi(t)$ need to satisfy the following complimentary conditions:

$$\psi_i \begin{cases} = 0 & \text{if } h_i < 0 \\ \geq 0 & \text{if } h_i = 0 \qquad i = 1, \ldots, l_\xi. \end{cases} \tag{3.10}$$

3.3 The variational problem

The optimal control problem defined in Problem 3.2.1 is related to a problem in the calculus of variations known as the *problem of Bolza*. The general control problem of Bolza can be stated as:

Problem 3.3.1 The problem of Bolza
Of all the arcs, $(x(t), u(t))$ satisfying the differential equation:

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

and the equality and inequality constraints:

$$\begin{aligned} g_i(t,x,u) &= 0 \quad i = 1, \ldots, l_\phi \\ h_i(t,x,u) &\leq 0 \quad i = 1, \ldots, l_\xi \end{aligned}$$

find the one that minimizes the integral:

$$J(x,u) = \gamma(t_0, x_0) + \int_{t_0}^{t_f} L\left(t, x(t), u(t)\right)\, dt$$

with the initial and final conditions:

$$x(t_0) = x_0 \qquad \textit{and} \qquad x(t_f) = x_f$$

respectively.

The inequality constraints can be converted to equality constraints by adding slack variables. Thus the modified inequality constraints are:

$$\hat{h}_i(t,x,u,\xi) = h_i(t,x,u) + {\xi_i}^2 = 0 \quad i = 1, \ldots, l_\xi$$

The optimal control problem defined in Problem 3.2.1 can be reformulated as the control problem of Bolza in the variational calculus framework using the method of Valentine [122] to convert inequality constraints to equality constraints and introducing the slack variables as new control inputs. Details of this procedure and its modification are discussed in greater detail in Section 3.4.1. The first order necessary conditions for solving the problem of Bolza are the Euler-Lagrange equations and the Weierstrass-Erdmann corner conditions.

3.3.1 Euler-Lagrange equations

Let $x(t)$ be an $n \times 1$ vector of functions which are continuous over the interval $[t_0, t_f]$ and let $z(t)$ be a piecewise smooth variation such that $z(t_0) = z(t_f) = 0$ and ϵ is some parameter such that $|\epsilon| < \delta$ where δ is a constant. We define the new set of functions, $x(t,\epsilon)$ as:

$$x(t,\epsilon) = x(t) + \epsilon z(t) \quad \text{for} \quad |\epsilon| < \delta$$

and the functional, $F(\epsilon)$ as:

$$F(\epsilon) = \int_{t_0}^{t_f} H(t, x(t,\epsilon), \dot{x}(t,\epsilon)) dt \tag{3.11}$$

Expanding Equation (3.11) in Taylor series, we get:

$$F(\epsilon) = F(0) + \epsilon \int_{t_0}^{t_f} (H_x z + H_{\dot{x}} \dot{z}) dt + \frac{1}{2}\epsilon^2 \int_{t_0}^{t_f} (H_{xx} z^2 + 2H_{x\dot{x}} z\dot{z} + H_{\dot{x}\dot{x}} \dot{z}^2) dt + O(|\epsilon^3|) \tag{3.12}$$

Thus, the critical condition is given by:

$$F'(0) = \int_{t_0}^{t_f} (H_x z + H_{\dot{x}} \dot{z}) dt = 0 \tag{3.13}$$

If we substitute, $H(t,x,u,\lambda,\phi,\psi,\xi) = L(t,x,u) + \phi^T g(t,x,u) + \psi^T \hat{h}(t,x,u,\xi)$, in Equation (3.13), and make a few manipulations using the Lemma of Du Bois Reymond, we obtain the Euler Lagrange equations which are the first order necessary conditions for the solution to the problem of Bolza. The Euler-Lagrange equations are given by:

$$\frac{d}{dt}\frac{\partial H(t,x,u,\lambda,\phi,\psi,\xi)}{\partial \dot{x}} = \frac{\partial H(t,x,u,\lambda,\phi,\psi,\xi)}{\partial x} \tag{3.14}$$

The above result along with the Weierstrass-Erdmann conditions for corner points is the general solution to the problem of Bolza and can be summarized in the following theorem:

Theorem 3.3.2 *Define the Hamiltonian, $H(t,x,u,\lambda,\phi,\psi,\xi)$ as:*

$$H(t,x,u,\lambda,\phi,\psi,\xi) = L(t,x,u) + \lambda^T f(x,u) + \phi^T g(t,x,u) + \psi^T \hat{h}(t,x,u,\xi)$$

If $x^*(t)$ yields a relative minimum for problem of Bolza defined in Problem 3.3.1 then there exist nonzero multipliers, λ,ϕ and ψ and the Hamiltonian, $H(t,x,u,\lambda,\psi,\phi,\xi)$ defined above so that $x^*(t)$ satisfies the Euler-Lagrange equations given in Equation (3.14) on every smooth portion of the trajectory of $x(t)$ and the Weierstrass-Erdmann corner conditions at every point of discontinuity, $c_i \in [t_0,t_f]$ of $\dot{x}(t)$ and $x(t_0) = x_0$ and $x(t_f) = x_f$. The Weierstrass-Erdmann corner conditions are given by:

$$\begin{aligned} \frac{\partial H}{\partial \dot{x}}|_{c_i^-} &= \frac{\partial H}{\partial \dot{x}}|_{c_i^+} \\ \left(\dot{x}\frac{\partial H}{\partial \dot{x}} - H\right)|_{c_i^-} &= \left(\dot{x}\frac{\partial H}{\partial \dot{x}} - H\right)|_{c_i^+} \end{aligned} \tag{3.15}$$

Remark 3.3.3 *The extremals of the optimal control problem defined in Section 3.2 and the extremals of the variational problem of Bolza given by Theorem 3.3.2 can be summarized as follows [135]:*

Theorem 3.3.4 *Let $x(t)$ be a piecewise smooth function and $u(t)$ be a piecewise continuous function as defined above.*

1. *If $x(t)$ satisfies the first order necessary conditions for the optimal control problem in Theorem 3.2.5, it also satisfies the first order necessary conditions for the problem of Bolza given by Theorem 3.3.2.*

2. *If $x(t)$ satisfies the first order necessary conditions for the problem of Bolza given by Theorem 3.3.2 and if the corresponding Lagrange multipliers, ψ, for the inequality constraints are non-negative for all $t \in [t_0,t_f]$ then $x(t)$ is also the critical solution to the optimal control problem in Theorem 3.2.5.*

3.4 Reformulation of the optimal control problem

Having established the correspondence between the critical solutions of the optimal control problem and its variational counterpart in the previous section, we will focus on reformulating the optimal control problem with equality and inequality constraints as an unconstrained variational calculus problem. This approach is useful since classical solutions to the optimal control problem given by Theorem 3.2.3 and 3.2.4 are difficult to compute numerically. Secondly, solving the constrained problem involves solving a two-point boundary value problem which is difficult to solve numerically.

We will reformulate the problem in Equations (3.1-3.4) as an unconstrained variational problem in which we will minimize the integral of a Hamiltonian, H:

$$J = \int_{t_0}^{t_f} H\left(X(t), \dot{X}(t), t\right)\, dt \tag{3.16}$$

We first treat the case with no inequality constraints[1].

No inequality constraints ($l_\xi = 0$): We construct the Hamiltonian by adjoining the original cost function, L, with the constraints:

$$H(X, \dot{X}) = L(x, u) + \lambda^T(\dot{x} - f(x, u)) + \phi^T g(x, u) \tag{3.17}$$

Here, X is the extended state vector:

$$X = [X_x^T, X_u^T, X_\lambda^T, X_\phi^T]^T \tag{3.18}$$

in which $X_x = x$ is the renamed $n \times 1$ state vector of the original problem, and X_u, X_λ, X_ϕ are defined according to the differential equations:

$$\dot{X}_u = u, \quad \dot{X}_\lambda = \lambda, \quad \dot{X}_\phi = \phi \tag{3.19}$$

λ and ϕ are the multipliers corresponding to the state Equations (3.1) and constraints in Equation (3.3) respectively. We let the inputs and the multipliers be derivatives of the extended state vector, X, so that they can be piecewise continuous while X (including the original state vector, x) is piecewise smooth. The unconstrained variational problem is then to minimize the functional in Equation (3.16) subject to suitable boundary conditions. The initial and final states, $X_x(t_0)$ and $X_x(t_f)$ are given. X_u, X_λ, and X_ϕ can be chosen to be zero at the initial time, since it is their derivatives that are important for our problem.

[1] In our applications, none of our equations will have an explicit dependence on time, t. From this point, for notational convenience we will stop showing the dependence on t.

Since they are otherwise unconstrained, they must satisfy the transversality conditions at t_f [107]:

$$\begin{aligned} &X_x|_{t_0} = x_0, \quad X_u|_{t_0} = 0, \quad X_\lambda|_{t_0} = 0, \quad X_\phi|_{t_0} = 0 \\ &X_x|_{t_f} = x_f, \quad \tfrac{\partial H}{\partial \dot{X}_u}|_{t_f} = 0, \quad \tfrac{\partial H}{\partial \dot{X}_\lambda}|_{t_f} = 0, \quad \tfrac{\partial H}{\partial \dot{X}_\phi}|_{t_f} = 0 \end{aligned} \tag{3.20}$$

Inequality constraints ($l_\xi \neq 0$): Here, we adjoin the inequality constraints by rewriting them as equality constraints. This is achieved by adding slack variables, ξ_i [62, 122]. Each inequality constraint can be converted to an equality constraint and written as:

$$h_i(t, x, u, \xi) \leq 0 \Leftrightarrow \hat{h}_i = h_i(t, x, u) + {\xi_i}^2 = 0, \quad i = 1, \ldots, l_\xi \tag{3.21}$$

We introduce two more variables in the extended state vector, X_ξ and X_ψ, that satisfy:

$$\dot{X}_\xi = \xi, \quad \dot{X}_\psi = \psi$$

where ξ is the vector of slack variables and ψ is the multiplier associated with the vector $\hat{h}$. Once again, we let ξ and ψ be derivatives of the state so that they can be piecewise continuous. The unconstrained variational problem is to minimize the functional in Equation (3.16) where:

$$H(X, \dot{X}) = L(x, u) + \lambda^T(\dot{x} - f(x, u)) + \phi^T g(x, u) + \psi^T \hat{h}(x, u, \xi) \tag{3.22}$$

with the state given by:

$$X = [X_x^T, X_u^T, X_\xi^T, X_\lambda^T, X_\phi^T, X_\psi^T]^T \tag{3.23}$$

subject to the boundary conditions:

$$\begin{aligned} &X_x|_{t_0} = x(t_0), \quad X_u|_{t_0} = 0, \quad X_\lambda|_{t_0} = 0, \quad X_\xi|_{t_0} = 0, \quad X_\phi|_{t_0} = 0, \quad X_\psi|_{t_0} = 0, \\ &X_x|_{t_f} = x(t_f), \quad \tfrac{\partial H}{\partial \dot{X}_u}|_{t_f} = 0, \quad \tfrac{\partial H}{\partial \dot{X}_\lambda}|_{t_f} = 0, \quad \tfrac{\partial H}{\partial \dot{X}_\xi}|_{t_f} = 0, \quad \tfrac{\partial H}{\partial \dot{X}_\phi}|_{t_f} = 0, \quad \tfrac{\partial H}{\partial \dot{X}_\psi}|_{t_f} = 0 \end{aligned} \tag{3.24}$$

The necessary conditions for $x(t)$, $u(t)$ to solve the system of Equations (3.1-3.4), is that there exists a piecewise smooth vector $X(t)$, which is a critical solution to Equation (3.16) with the Hamiltonian given by Equation (3.22) and the boundary conditions given by Equation (3.24).

The procedure outlined in this section so far can be used for kinematic motion planning and is illustrated in Example 1 in Section 5.4. When dynamic considerations are important, the problem becomes more complicated and state constraints require special attention. This is explained in the next section and illustrated in Examples 2-4 in Section 5.5.

3.4.1 Constraints on the state

Equality constraints: When equality constraints are at the state level and do not involve the input u explicitly, we may be faced with numerical difficulties. In order to avoid this, we repeatedly differentiate the constraints $g_i(x)$, with respect to time substituting the state equations for the derivative of the state until the resulting equation explicitly contains u. Let p_i be the lowest order derivative of g_i in which the dependence on u becomes explicit. After differentiation, the original constraint, g_i, is renamed as g_{m_i} and is given by:

$$g_{m_i}(x,u) = g_i^{(p_i)}(x,u) = 0 \tag{3.25}$$

In order to satisfy the original equality constraints, $g_i(x) = 0$, we enforce the following boundary conditions on the lower order derivatives of $g_i^{p_i}(x,u)$. These boundary conditions are given by:

$$\begin{aligned} g_i(x_0) &= 0 \\ \dot{g}_i(x_0) &= 0 \\ &\vdots \\ g_i^{(p_i-1)}(x_0) &= 0 \end{aligned} \tag{3.26}$$

Thus, we adjoin the vector, $g_m = [g_{m_1}, g_{m_2}, \ldots, g_{m_{l_\phi}}]^T$, instead of the vector g to the Hamiltonian in Equation (3.22).

Inequality constraints: The potential difficulties associated with state inequality constraints are discussed in [21, p.117], [62]. When the trajectory goes from an unconstrained segment to a constrained segment, there is a discontinuity in the adjoint variables (multipliers) and this gives rise to numerical problems. This problem is circumvented by differentiating the state inequality constraints until the input becomes explicit [62]. The arcs lying on the state constraint boundary in the original (undifferentiated) optimal control problem now become singular arcs in the transformed (differentiated) problem. The discontinuities in the adjoint variables at the junction of a constrained and an unconstrained arc are eliminated, thus leading to increased convergence speed and more accurate numerical results.

Some of the inequality constraints in Equation (3.21) may need differentiation as they may not have an explicit dependence on the input. These ϱ equations that do not have an explicit dependence on the input need to differentiated until this dependence is obtained.

Instead of adjoining $\hat{h}$ in Equation (3.22), we adjoin $\hat{h}_m$ to the Hamiltonian, where:

$$\hat{h}_{m_i}(x,u) = h_i^{(q_i)}(x,u) + \frac{d^{(q_i)}\xi_i^2}{dt^{(q_i)}} = 0 \tag{3.27}$$

When the i^{th} constraint is inactive, $\xi_i^{(q_i)}$ is non zero and it determines the evolution of the system relative to the constraint. When the constraint is satisfied, $\xi_i^{(q_i)}$ is zero. We accordingly modify X_ξ so that $\dot{X}_\xi$ contains the q_i^{th} derivative of each inequality, and we order the inequalities so that $q_i > 0$ for the first ϱ inequalities:

$$\dot{X}_\xi = [\xi_1^{(q_1)}, \xi_2^{(q_2)}, \ldots, \xi_\varrho^{(q_\varrho)}, \xi_{\varrho+1}, \ldots, \xi_{l_\xi}]$$

These variables can be viewed as additional inputs for the unconstrained system. However, when one or more constraints are satisfied, the corresponding $\xi_i^{(q_i)}$ are zero and the optimal control is singular [62].

Unlike Equation (3.25), Equation (3.27) involves the derivatives of the slack variables. This introduces new dynamics into the system [62]. The slack variable ξ_i and its derivatives up to the $(q_{i-1})^{th}$ order are part of the description of the system dynamics. We will denote the q_i new variables, $\xi_i, \dot{\xi}_i, \cdots, \xi_i^{(q_i-1)}$, by $\eta_{i,1}, \ldots, \eta_{i,q_i}$. Thus,

$$\begin{aligned} \eta_{i,1} &= \xi_i \\ \eta_{i,2} &= \dot{\xi}_i \\ &\vdots \\ \eta_{i,q_i} &= \xi_i^{q_i-1} \end{aligned} \tag{3.28}$$

We define a new vector, X_η, of dimension $\sum_{i=1}^{\varrho} q_i$, that will be part of our state X:

$$X_\eta = [\eta_{1,1}, \eta_{1,2}, \ldots, \eta_{1,q_1}, \eta_{2,1}, \eta_{2,2}, \ldots, \eta_{2,q_2}, \ldots, \eta_{\varrho,1}, \eta_{\varrho,2}, \ldots, \eta_{\varrho,q_\varrho}]^T$$

There are $\sum_{i=1}^{\varrho}(q_i - 1)$ associated state equations:

$$c(X_\eta, \dot{X}_\eta) = 0 \tag{3.29}$$

where the equations take the form:

$$\begin{aligned} \dot{\eta}_{i,1} &= \eta_{i,2} \\ \dot{\eta}_{i,2} &= \eta_{i,3} \\ &\vdots \\ \dot{\eta}_{i,(q_i-1)} &= \eta_{i,q_i} \end{aligned} \tag{3.30}$$

Finally, all the constraints are adjoined to the Hamiltonian in Equation (3.22). We will adjoin the l_ξ constraints in Equation (3.27) and the state equations in Equation (3.29). The modified Hamiltonian is given by:

$$H(X,\dot{X}) = L(x,u) + \lambda^T(\dot{x} - f(x,u)) + \phi^T g_m + \psi^T \hat{h}_m + \zeta^T(c(X_\eta, \dot{X}_\eta)) \tag{3.31}$$

where the vector, $\hat{h}_m = [\hat{h}_{m_1}, \hat{h}_{m_2}, \ldots, \hat{h}_{m_{l_\xi}}]^T$, is used instead of the vector $\hat{h}$. The extended state vector is:

$$X = [X_x^T, X_\eta^T, X_u^T, X_\xi^T, X_\lambda^T, X_\phi^T, X_\psi^T, X_\zeta^T]^T \tag{3.32}$$

where

$$\dot{X}_\zeta = \zeta = [\zeta_{1,1}, \ldots, \zeta_{1,q_1-1}, \zeta_{2,1}, \ldots, \zeta_{2,q_2-1}, \ldots, \zeta_{\varrho,1}, \ldots, \zeta_{\varrho,q_\varrho-1}]$$

and ζ is the $l_\zeta \times 1$ vector of multipliers, where

$$l_\zeta = \sum_{i=1}^{\varrho}(q_i - 1).$$

ψ is an $l_\xi \times 1$ vector of multipliers for the modified inequality constraints in Equation (3.27) and ϕ is an $l_\phi \times 1$ vector of multipliers for the modified equality constraints in Equation (3.25). Thus the Hamiltonian depends on the $r \times 1$ state vector X, where:

$$r = n + \sum_{i=1}^{\varrho} q_i + m + l_\xi + n + l_\phi + l_\xi + \sum_{i=1}^{\varrho}(q_i - 1) \tag{3.33}$$

The boundary conditions are given by:

$$\begin{array}{llllll}
X_x|_{t_0} = x(t_0), & X_\eta|_{t_0} = \eta(t_0), & X_u|_{t_0} = 0, & X_\lambda|_{t_0} = 0, & X_\xi|_{t_0} = 0, & X_\phi|_{t_0} = 0, \\
X_\psi|_{t_0} = 0, & X_\zeta|_{t_0} = 0, & X_x|_{t_f} = x(t_f), & X_\eta|_{t_f} = \eta(t_f), & \frac{\partial H}{\partial \dot{X}_u}|_{t_f} = 0, & \frac{\partial H}{\partial \dot{X}_\lambda}|_{t_f} = 0, \\
\frac{\partial H}{\partial \dot{X}_\xi}|_{t_f} = 0, & \frac{\partial H}{\partial \dot{X}_\phi}|_{t_f} = 0, & \frac{\partial H}{\partial \dot{X}_\psi}|_{t_f} = 0, & \frac{\partial H}{\partial \dot{X}_\zeta}|_{t_f} = 0 & &
\end{array}$$

Table 3.1: The variables in the extended state vector for the variational problem.

Equations	State variables	Inputs	Multipliers	Slack variables
State equations	n	m	n	-
l_ϕ equality constraints	-	-	l_ϕ	-
Inequality constraints	$\sum_{i=1}^{\varrho} q_i$	-	Λ_1	l_ξ
Total	$n + \sum_{i=1}^{\varrho} q_i$	m	Λ_2	l_ξ

Table 3.1 accounts for the various terms in Equation 3.33 where $\Lambda_1 = l_\xi + \sum_{i=1}^{\varrho}(q_i - 1)$ and $\Lambda_2 = n + l_\phi + l_\xi + \sum_{i=1}^{\varrho}(q_i - 1)$.

Remark 3.4.1 *We may substitute for the control inputs in terms of the state vector and its derivatives explicitly in the Hamiltonian. This is always possible for affine systems. This procedure eliminates m control inputs and the associated multipliers. It reduces the dimensionality of the extended state vector and decreases the complexity of the variational problem.*

3.4.2 Difficulties with state constraints

We saw in the previous section that state constraints that are not explicitly dependent on the input need to be differentiated until the dependence on the input becomes explicit. If a constraint $g(t, x)$ has to be differentiated p times to make the dependence on u explicit, we enforce p boundary conditions on g and its derivatives (up to order $(p-1)$ at t_0 as mentioned before and adjoin $g^{(p)}(t, x, u)$ to the Hamiltonian to enforce the constraint. Since the constraint is not explicitly enforced at all mesh points (only the p^{th} derivative is enforced), the solution may drift due to numerical errors and the constraint may not be satisfied at points far away from the initial value.

In order to "stabilize" equality constraints, we use a trick that is routinely used for initial value problems [13]. Instead of adjoining the p^{th} derivative of the state constraint, we adjoin a linear combination of the state constraint and its derivatives up to the p^{th} level choosing the constants k_i so that the solution of the differential equation:

$$g^{(p)}(x, u) + k_{p-1}g^{(p-1)}(x) + \ldots + k_1\dot{g}(x) + k_0g(x) = 0$$

exponentially goes to zero. For example, we can enforce a second order ($p = 2$) constraint $g(t, x) = 0$, by adjoining the term:

$$\ddot{g}(x, u, t) + k_1\dot{g}(x, t) + k_0g(x, t)$$

where the constants k_0 and k_1 can be picked so that $g(t, x)$ exponentially goes to zero. Since inequality constraints are converted to equality constraints in our method, this approach can be used with inequality constraints as well. This approach greatly enhances the convergence and the accuracy of the solutions.

3.5 Example

We now consider a general motion planning problem of a point in R^2. The purpose of this section is to introduce the general technique used in solving some of the other complicated motion planning problems whose solutions appear in this thesis. The methodology adopted in this example will clarify some of the theoretical results of the previous sections. Consider the example of finding the trajectory of a point moving around a circular obstacle

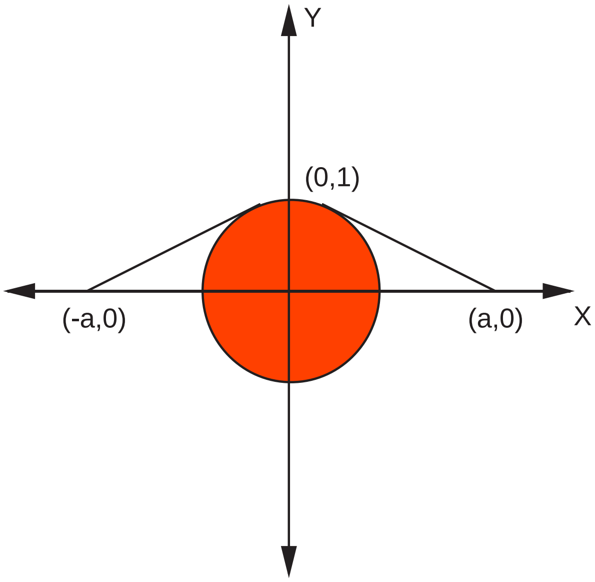

Figure 3.1: A canonical motion planning problem.

as shown in Figure 3.1. The coordinates of the initial and final positions of the point are $(-a, 0)$ and $(a, 0)$ respectively. The radius of the circular obstacle is taken to be unity, *i.e.*, $R = 1$. The cost function to be minimized is given by:

$$J = \int_0^1 \left(\dot{x}_1^2 + \dot{x}_2^2\right) dt \tag{3.34}$$

subject to the following constraint and boundary conditions:

$$x_1^2 + x_2^2 - R^2 \geq 0 \tag{3.35}$$

$$\begin{aligned} x_1(0) &= -a \quad & x_2(0) &= 0 \\ x_1(1) &= a & x_2(1) &= 0 \end{aligned} \tag{3.36}$$

If we apply the method of Gregory and Lin [50] without differentiating the constraints until the input appears explicitly, there will be impulsive jumps in the multipliers as predicted by Jacobsen *et al.* [63]. This in general lead to poor numerical accuracy and increased convergence time in some cases. Hence we resort to differentiation of the constraints and eventually stabilize it for accelerated convergence. In this section, we will outline the procedure used and give the analytical results for the motion planning problem mentioned above. We note that the constraint to be satisfied at all times is not a distance constraint. However, it behaves very much like one. We first subtract the square of the slack variable, x_4, in the original constraint so that the equality holds at all points on the path. Thus the constraint is:

$$x_1^2 + x_2^2 - R^2 - x_4^2 = 0 \tag{3.37}$$

Differentiation of this constraint leads to the new constraint:

$$x_1\dot{x}_1 + x_2\dot{x}_2 - x_4\dot{x}_4 = 0 \tag{3.38}$$

However, x_4 needs to satisfy the initial condition and is given by:

$$x_4(0) = x_1^2(0) + x_2^2(0) - R^2$$

This constraint can be adjoined with the original cost function by a multiplier, $\dot{x}_3$, to form the Hamiltonian and is given by:

$$H(X, \dot{X}) = \dot{x}_1^2 + \dot{x}_2^2 + \dot{x}_3(x_1\dot{x}_1 + x_2\dot{x}_2 - x_4\dot{x}_4) \tag{3.39}$$

where $X = [x_1, x_2, x_3, x_4]^T$ and $\dot{X} = [\dot{x}_1, \dot{x}_2, \dot{x}_3, \dot{x}_4]^T$. We enforce $x_3(0) = 0$ as the initial condition. Based on the Hamiltonian in Equation (3.39) we have:

$$H_X = \begin{pmatrix} \dot{x}_1\dot{x}_3 \\ \dot{x}_2\dot{x}_3 \\ 0 \\ -\dot{x}_3\dot{x}_4 \end{pmatrix} \tag{3.40}$$

$$H_{\dot{X}} = \begin{pmatrix} 2\dot{x}_1 + x_1\dot{x}_3 \\ 2\dot{x}_2 + x_2\dot{x}_3 \\ x_1\dot{x}_1 + x_2\dot{x}_2 - x_4\dot{x}_4 \\ -x_4\dot{x}_3 \end{pmatrix} \tag{3.41}$$

The first order necessary conditions for a minimum (Euler-Lagrange equations) is given by:

$$\frac{d}{dt}H_{\dot{X}} = H_X \tag{3.42}$$

That leads to the following set of equations:

$$\begin{aligned} 2\ddot{x}_1 &= -x_1\ddot{x}_3 \\ 2\ddot{x}_2 &= -x_2\ddot{x}_3 \\ x_1\dot{x}_1 + x_2\dot{x}_2 - x_4\dot{x}_4 &= C_1 \\ x_4\ddot{x}_3 &= C_2 \end{aligned} \tag{3.43}$$

It can be easily shown that C_1 and C_2 are identically 0. Since x_3 and x_4 is not specified at the terminal point, transversality conditions are imposed and they are given by:

$$\begin{aligned} H_{\dot{x}_3} &= 0 \\ H_{\dot{x}_4} &= 0 \end{aligned} \tag{3.44}$$

Thus at $t = 1$,

$$\begin{aligned} x_1\dot{x}_1 + x_2\dot{x}_2 - x_4\dot{x}_4 &= 0 \\ x_4\dot{x}_3 &= 0 \end{aligned}$$

The Weierstrass-Erdmann corner conditions are satisfied at points where $\dot{x}$ becomes discontinuous. The corner conditions at $t = t_1$ are given by:

$$\begin{aligned} H_{\dot{X}}|_{t_1-} &= H_{\dot{X}}|_{t_1+} \\ H_{t_1-} - \dot{X}^T H_{\dot{X}}|_{t_1-} &= H_{t_1+} - \dot{X}^T H_{\dot{X}}|_{t_1+} \end{aligned} \tag{3.45}$$

Thus using the corner conditions and the Euler-Lagrange equations we obtain the following analytical solution for the problem:

$$x_1(t) = \begin{cases} C_3 t - a & 0 \le t \le t_1 \\ \cos(C_4 t + C_5) & t_1 \le t \le t_2 \\ C_6(t-1) + a & t_2 \le t \le 1 \end{cases}$$

$$x_2(t) = \begin{cases} C_7 t & 0 \le t \le t_1 \\ \sin(C_4 t + C_5) & t_1 \le t \le t_2 \\ C_8(t-1) & t_2 \le t \le 1 \end{cases}$$

$$x_3(t) = \begin{cases} C_9 t & 0 \le t \le t_1 \\ C_4^2 t^2 + C_{10} t + C_{11} & t_1 \le t \le t_2 \\ C_{12}(t-1) & t_2 \le t \le 1 \end{cases}$$

$$x_4(t) = \begin{cases} \sqrt{x_1^2 + x_2^2 - R^2} & 0 \le t \le t_1 \\ 0 & t_1 \le t \le t_2 \\ \sqrt{x_1^2 + x_2^2 - R^2} & t_2 \le t \le 1 \end{cases}$$

where

$$\begin{array}{ccc} t_1 = \frac{\sqrt{3}}{2\sqrt{3}+\pi/3} & t_2 = \frac{\sqrt{3}+\pi/3}{2\sqrt{3}+\pi/3} & C_3 = \frac{\sqrt{3}}{2}\{2\sqrt{3}+\pi/3\} \\ C_4 = -\{2\sqrt{3}+\pi/3\} & C_5 = 2\pi/3+\sqrt{3} & C_6 = C_3 \\ C_7 = \frac{2\sqrt{3}+\pi/3}{2} & C_8 = -C_7 & C_9 = -\frac{2\pi}{3}\{2\sqrt{3}+\pi/3\} \\ C_{10} = -2C_4\{\sqrt{3}+\pi/3\} & C_{11} = C_9 t_1 - C_4^2 t_1^2 - C_{10} t_1 & C_{12} = C_4^2 t_2^2 + C_{10} t_2 + C_{11} \end{array} \tag{3.46}$$

Figure 3.2(a) shows the path followed by the particle from the initial to the final configuration. Since the analytical and numerical solutions coincide, we have only one trajectory in Figure 3.2(a). Similar is the case for the multiplier $\dot{x}_3$, in Figure 3.2(b). Based on the results of the equivalence of the critical solutions, we see that the multiplier in Figure 3.2(b) is non-positive as the constraint in Equation (3.35) is opposite to the notation in Equation (3.4).

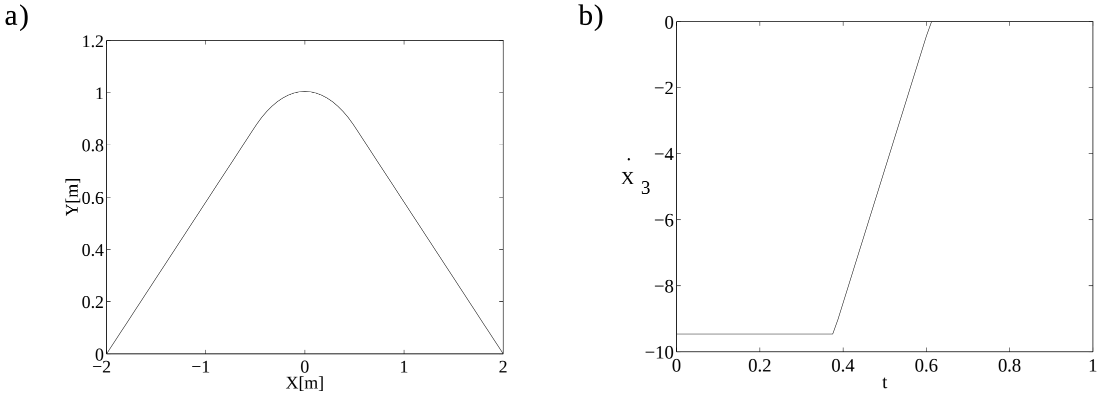

Figure 3.2: Analytical and Numerical solutions for a) the path of the particle and b) Multiplier $\dot{x}_3$.

Clearly, the above example demonstrates the methodology that we have developed in dealing with equality and inequality constraints though the above example did not have any equality constraints. However, for the problems considered in this thesis, the solutions are not easy to compute analytically and hence we have to use numerical techniques. In the following section, we describe the numerical procedure used to compute solutions for a broad class of problems.

3.6 Numerical method

3.6.1 Integral formulation of the necessary conditions

Consider the general variational calculus problem with the $r \times 1$ state vector X:

$$I = \min \int_{t_0}^{t_f} H(t, X, \dot{X})\, dt \tag{3.47}$$

subject to boundary conditions on X at t_0 and t_f. Define an admissible variation, $z(t)$, to be a piecewise smooth function that satisfies the boundary conditions at t_0 and t_f. We will follow the approach proposed by Gregory and Lin [50] in which the admissible variations are approximated by piecewise continuous basis functions. We then consider the first variation of the functional in Equation (3.47) and derive conditions under which it is zero.

If $X(t)$ is a solution of the variational problem in Equation (3.47), then for all admissible variations $z(t)$, the following integral equation must be satisfied as in Section 3.3.1

$$\int_{t_0}^{t_f} [{H_X}^T z + {H_{\dot{X}}}^T \dot{z}]\, dt = 0 \tag{3.48}$$

where H_X and $H_{\dot{X}}$ denote partial derivatives with respect to X and $\dot{X}$ respectively. This necessary condition should be contrasted with the traditional Euler-Lagrange equations. The main disadvantage with using Euler-Lagrange equations is that they are not valid at points called “corners” where the derivative of the state is discontinuous. Thus at these points, the Weirstrass-Erdmann corner conditions apply as stated in Equation (3.15). In most motion planning problems, the corner points are not *a priori* known. By directly solving the integral equation in Equation (3.48), we eliminate the need to patch solutions at the corners. The integral equation is valid at all points, even at points at which $\dot{X}(t)$ is discontinuous.

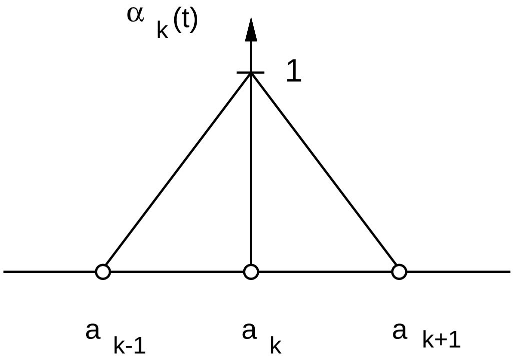

Figure 3.3: Shape functions used in the numerical method.

We represent the piecewise smooth variation $z(t)$ using basis functions. We first discretize the interval $[t_0, t_f]$ so that $t_0 = a_0 < a_1 < \cdots < a_N = t_f$. For simplicity, we will assume that $a_i - a_{i-1} = h$, for $i = 1, \ldots, N$. Since our controls need to be piecewise continuous, the simplest basis function whose derivative is piecewise continuous is the piecewise linear shape function. Thus, we introduce a set of piecewise linear shape functions:

$$\alpha_k(t) = \begin{cases} \frac{t-a_{k-1}}{h} & \text{if} \quad a_{k-1} < t \leq a_k, \\ \frac{a_{k+1}-t}{h} & \text{if} \quad a_k < t \leq a_{k+1}, \\ 0 & \text{otherwise}, \quad (k = 0, \ldots, N) \end{cases} \tag{3.49}$$

The components of the variation $z(t)$ are approximated by the sum:

$$\hat{z}_i = \sum_{j=0}^{N} z_{ij}\alpha_j(t)$$

Next we use the central difference scheme to approximate the derivatives and the mean-value theorem to approximate the integral in Equation (3.48), on each subinterval $[a_{i-1}, a_i]$. This process yields the following set of vector equations in the unknown values X_k at points $k = 1, \ldots, N-1$ [50, pg. 134]:

$$\begin{aligned} 0 = {} & \frac{h}{2}H_X(\frac{a_k + a_{k-1}}{2}, \frac{X_k + X_{k-1}}{2}, \frac{X_k - X_{k-1}}{h})+ \\ & H_{\dot{X}}(\frac{a_k + a_{k-1}}{2}, \frac{X_k + X_{k-1}}{2}, \frac{X_k - X_{k-1}}{h}) + \\ & \frac{h}{2}H_X(\frac{a_{k+1} + a_k}{2}, \frac{X_{k+1} + X_k}{2}, \frac{X_{k+1} - X_k}{h}) - \\ & H_{\dot{X}}(\frac{a_{k+1} + a_k}{2}, \frac{X_{k+1} + X_k}{2}, \frac{X_{k+1} - X_k}{h}) \end{aligned} \tag{3.50}$$

Note that the unknowns, X_k, are vectors. In all, we have $r(N+1)$ unknowns. We also have $r(N-1)$ equations from Equation (3.50). The remaining $2r$ equations are obtained from the boundary conditions.

The resulting system of $r(N+1)$ nonlinear equations is solved using the Newton-Raphson method. Each equation only depends on three adjacent points. The matrix of the system of linear equations solved during the iteration is thus block-tridiagonal and the system can be solved very efficiently.

3.7 Conclusions

In this chapter, we have discussed the general optimal control problem and its corresponding variational problem known as the problem of Bolza. The methodology presented is applicable to general motion planning problems with geometric, kinematic and dynamic constraints. The geometric constraints include constraints imposed by the presence of obstacles while the kinematic constraints may be holonomic or nonholonomic. The dynamic constraints include the equations of motion and constraints on the contact forces. In addition, there may be inequality constraints like constraints on the turning radius for a mobile platform or bounds on the minimal and maximal separation between the platforms. Since the solutions to the general optimal control problem as proved by Pontryagin is not easily implemented in numerical computations, it is reasonable to consider the unconstrained variational formulation. We adopt this strategy for solving general motion planning problems by formulating it as a problem in the calculus of variations and solving it using the integral form of the necessary conditions. There are several advantages to this method:

1. The method is general in the sense that any state or input based constraint (equality or inequality) can be incorporated and any state or input based cost functional can be chosen.

2. In problems with inequality constraints, the system can switch back and forth between constrained and unconstrained states any number of times. The method is transparent to the number of switches and to the corner conditions that characterize the variational problem.

3. When inequality constraints are imposed on the state (for example, due to obstacles or joint limits), the optimal trajectories may not be smooth. This method works with the class of all piecewise smooth trajectories and is therefore more powerful than methods that use a finite number of smooth basis functions to approximate the solution.

4. Although the resulting variational problem is of much higher dimension than the original optimal control problem, the resulting problem can be solved reasonably efficiently using standard off-the-shelf algorithms.

There is one issue pertaining to the numerical method that is not addressed in this chapter. It has to do with the so-called second order necessary conditions for the variational problem. The methods of Section 3.3 guarantee that the first order necessary conditions are satisfied. Second order conditions are checked offline to ensure that we indeed have a locally minimal solution. There is a shortcoming with the method of slack variables. Solutions to necessary conditions to the original problem with inequalities will satisfy the necessary conditions for the corresponding problem with slack variables. However, in general, there may be spurious solutions in problems with slack variables. Fortunately, these are easy to detect off-line. Both these shortcomings are easily overcome with appropriate numerical algorithms [14] that explicitly minimize the cost function as opposed to solving necessary conditions. In the following chapters we will apply the method developed in this chapter to solve complex motion planning problems in human and robotic systems.

Chapter 4

Computational model for human motion

In this chapter, we study the problem of human motion and model the manipulation task in the framework of robotics. The methodology developed in Chapter 3 will be used to mathematically model the system. In this chapter, we study the problem of human dual arm manipulation and develop a computational model that predicts the trajectories and force distribution for the coordination of two arms moving an object between two given positions and orientations in the horizontal plane. We begin by the description of the experimental system and details of the experimental task. Later, we develop a computational model based on the work of Uno *et al.* [121] to model the experimental system. Finally, we will present some experimental results on human vertical arm movements and compare the force and trajectory profiles with planar movements.

4.1 Experimental system

4.1.1 Planar two arm movements

The experimental system for measuring trajectories and force distribution consists of a target system and a passive planar manipulandum with a handle as shown in Figure 4.1. The sides of the handle are parallel plates and simulate a box-like object. Metallic tabs can be added to the handle in order to vary its weight and inertia. Each plate is instrumented with a six-axis force/torque sensor allowing measurement of the forces and torques exerted by the subject during the manipulation task. The handle assembly is attached to the manipulandum with a low-friction, linear bearing. Thus, during the experiment, the weight of the object is supported by the two palms and not by the manipulandum.

The manipulandum consists of three links, connected by revolute joints. The first two links of the manipulandum form a serial kinematic chain capable of locating the distal end of the second link (which coincides with the center of the third link) at any position (x, y) in the horizontal plane (two degrees of freedom) within the manipulandum workspace. By revolving about its center point, the third link (the handlebar) provides a third, rotational degree of freedom (ϕ). Three optical encoders mounted at the joints are used to measure the corresponding angles of rotation at a sampling rate of 200 Hz. During the experiments, the subject sits in front of the manipulandum, and firmly grasps the two flat plates of the handle as shown in Figure 4.1. A rectangular wooden frame supporting a transparent Plexiglas sheet is suspended from the ceiling by nylon cables, such that the Plexiglas sheet is horizontal at the level of the subject's chin (Figure 4.1). Four target sets are mounted on the Plexiglas at different locations, and each set consists of arrays of light emitting diodes (LEDs). A schematic of the top view showing the four targets on the plexiglass sheet and the human holding the force sensors is as shown in Figure 4.2. The task was to move from one lit array of LED's to another. The room was darkened, and a random sequence of target configurations was presented. The duration during which any given target was lit was also randomly determined in the range 1.5 - 3.5 seconds. There was adequate time for the subject to complete the motion and all motions started from complete rest. The geometry and position of the diodes is as shown in Figure 4.3. The subjects were

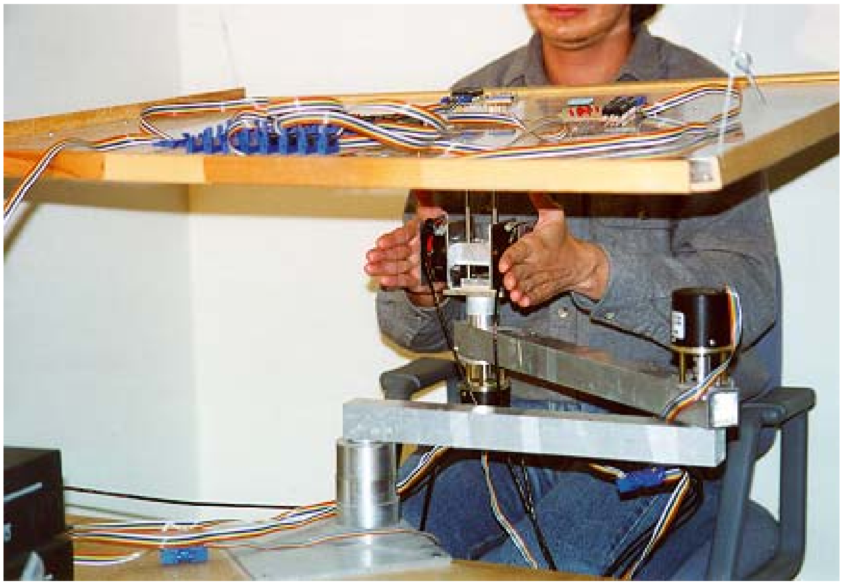

Figure 4.1: Experimental testbed.

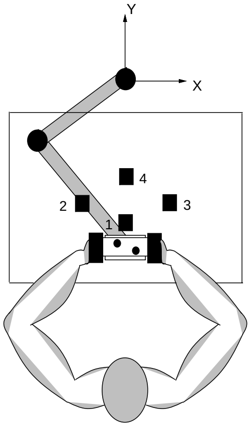

Figure 4.2: Top view of planar manipulation task.

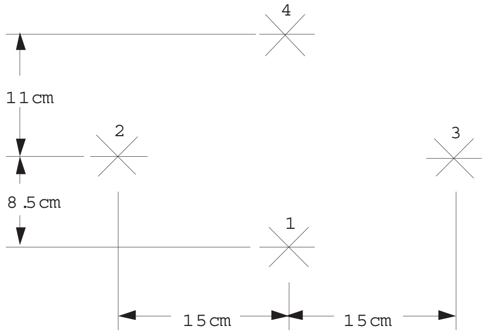

Figure 4.3: Schematic of the target assembly for planar movements.

instructed to position and orient the handlebar at the configuration specified by the lit array of LED's, while maintaining a firm grasp on the side plates with both hands and keeping the elbows in the horizontal plane passing through the shoulders. They were told to move naturally, *i.e.*,at what was in their opinion a comfortable speed. High accuracy was not required from the subjects, as they were instructed not to be unduly concerned about small errors in final position and orientation. Furthermore, since the experiments were performed in the dark, no visual feedback of the arms and the handle was provided to the subjects. Some of the subjects commented that they were able to see the handle after they adapted to the darkness. Each subject was asked to perform five groups of fifty movements. While the subjects carried out the task, force information was captured by the six-axis force/torque sensor (made by ATI : model : 65/5 Gamma F/T sensor) while the encoders gave information about the position of the end-effector of the manipulandum in the plane. Eight healthy subjects participated in the experiments, ages ranging between 24 and 40. Six of these subjects were right handed while two were left handed. The first fifty motions from each subject were discarded allowing the subject to adapt to the experimental task (though they were not informed that the first fifty motions would be discarded). After each set of fifty motions, the subjects were given adequate time (as they required) before the onset of another set of data. Thus the subjects had a chance to relax before they proceeded to the next set thereby removing any effects of excessive fatigue in the experimental data.

4.1.2 Vertical two arm movements

The experimental system for measuring the trajectories and forces in vertical arm movements is different than the one used in planar arm movements. Instead of a spatial manipulandum, we use the OPTOTRAK system to capture the trajectory of the grasped object. We placed four infrared light emitting diodes on a grid as shown in Figure 4.5(a). The metallic grid is mounted on the object being manipulated and the location of the infrared sensors is used to compute the position and orientation of the grasped object. Figure 4.5(a) shows the manipulated object which consists of force sensors on either ends grasped by open palms and four infrared markers which are tracked by the OPTOTRAK during the movement of the object. In vertical arm movements the object has three rotational and three translational degrees of freedom. In our experimental setup, we used a similar target display as in planar studies except that we increased the span of frontal plane motion. A schematic of the target assembly is as shown in Figure 4.4 with the targets numbered from 5 through 8. The target assembly was fixed to the wall at a variable height depending on the comfort of the subject in moving from one target to another. The placement of the display was such that the subjects were able to reach all the targets with ease. The actual experimental setup for vertical arm movements is as shown in Figure 4.5(b) with a subject grasping the object.

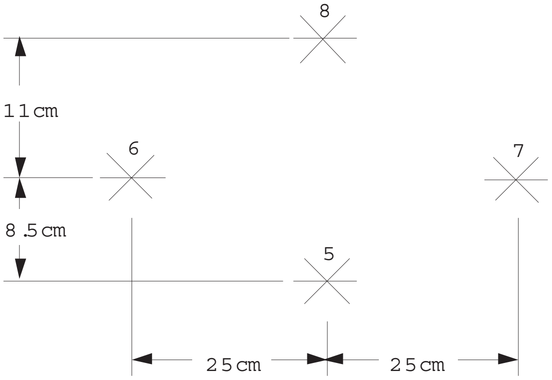

Figure 4.4: Schematic of the target assembly for vertical two arm movements.

The OPTOTRAK system consists of a position sensor unit and a system control unit. The OPTOTRAK system is a infrared-based, non-contact movement measurement system. The three optical sensors in the position sensor unit track a infrared light emitting diode (LED) attached to the moving object. The position sensor unit consists of three one dimensional charged coupled devices (CCD's) paired with three lens cells and mounted on a long stabilized bar. Within each of the three lens cells, light from an infrared source

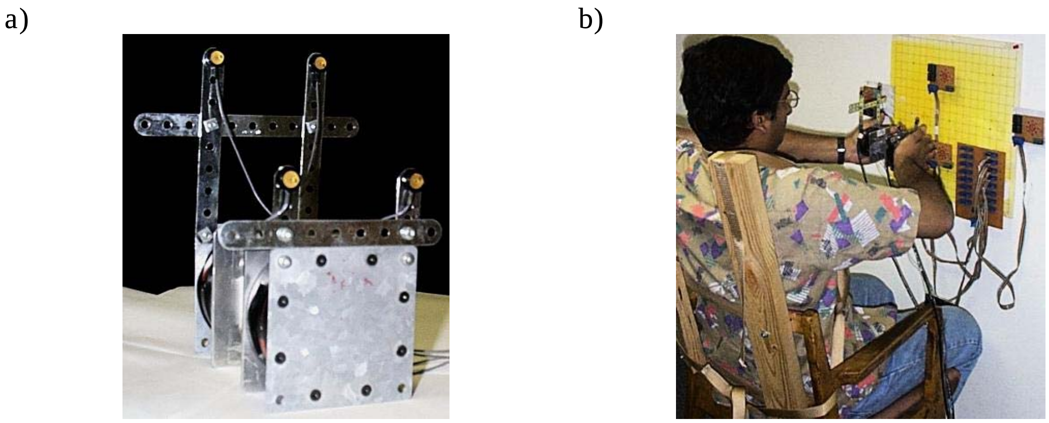

Figure 4.5: a) Force sensors with infrared markers b) Experimental system for vertical movements.

is directed onto the CCD and measured. These measurements determine the location of the marker in space in real time. Since the OPTOTRAK system is pre-calibrated, there is no need for routine calibration. It has the capability of tracking 256 markers in real-time with a peak scan rate of 3000 Hz when a single marker is used. It is also capable of very accurate measurements with accuracies of $0.1mm$ and resolutions of up to $0.01mm$. The system also comes with a system control unit which is the central controlling, interfacing and processing unit of the OPTOTRAK system. The unit is a bridge between the position sensor unit and the host computer and relays data in real time. The operating range is $1.5m$ to $3m$ from the position sensor unit. The actual measurement system with the system control unit is as shown in Figure 4.6.

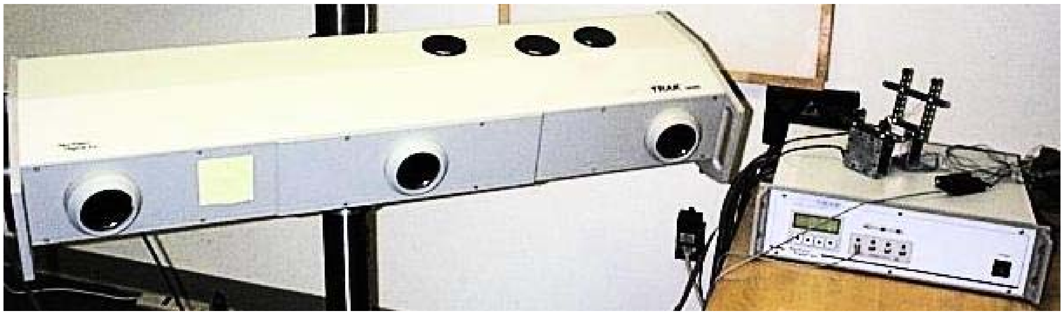

Figure 4.6: The OPTOTRAK system.

As in the case of planar arm movements, the lights in the room were completely shut off so that the subject had no visual feedback of the object being manipulated and they were required to align the object as best as possible with the array of lighted LED's. In this study, we took 5 sets of 30 movements and all the experimental data was analyzed offline. Between each set of movements, the subjects had sufficient time to relax their arms. All other experimental conditions were similar to planar arm movements.

The instructions given to the subject prior to starting the experiment are as stated below:

- Hold the object with your open palms and do not curl your fingers around the object.

- Move as naturally as possible and follow the the lighted array of LED from one target to another.

- There is adequate time to move from one target to another and hence avoid any undue hasty movements or excessive force on the object being manipulated.

- At all times keep the arms and the object in the horizontal plane for planar arm movements.

- There are 5 sets of experiments with a break between each set so that there is enough time to relax the arms.

- In vertical two arm studies, the subjects were asked to align the horizontal bar at the center of the grasped object with the direction of the lighted LED's.

4.2 A computational model for motion generation

The schematic of human holding the grasped object is as shown in Figure 4.7(a) and the associated normal and tangential forces are as shown in Figure 4.7(b). We denote the normal and tangential forces by F_{in} and F_{it} (i=1,2) respectively.

We model the two arms holding an object in the horizontal plane by two planar 3-link manipulators as shown in Figure 4.8. Each arm has 3 degrees of freedom. If the arms rigidly hold the object, the system of the two arms and the object (closed kinematic chain) has three degrees of freedom. The object can therefore be placed at an arbitrary position and orientation in the plane. Hence, the closed kinematic chain is not kinematically redundant. However, since there are 6 joints, there are 6 actuators and thereby the system is overconstrained. Thus, there can in principle be infinite possible force and motion profiles which will guarantee the desired motion of the object from one position and orientation to another. The closed loop is modeled by equality constraints on the position variables and the friction-assisted grasp by inequality constraints on the contact forces. The dy-

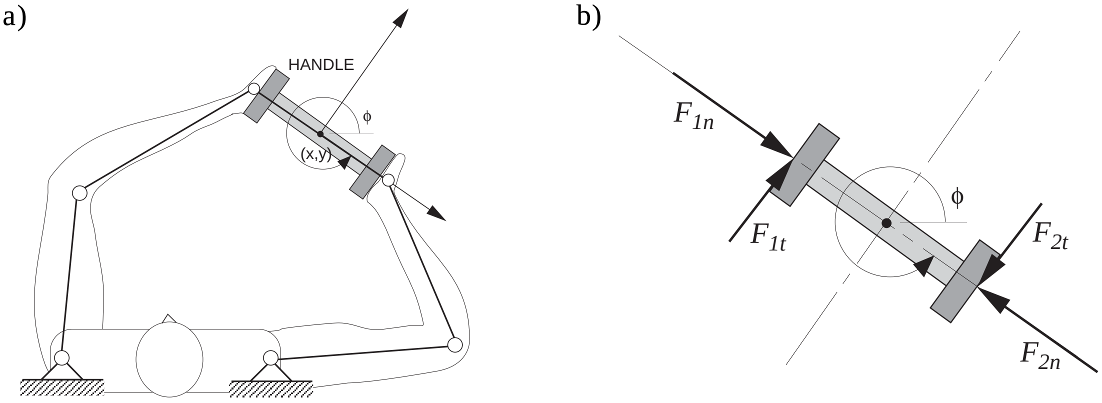

Figure 4.7: a) Schematic of human grasping the object and b) Forces exerted by the human subject.

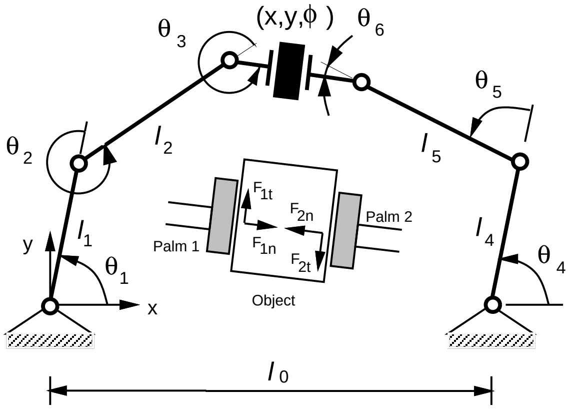

Figure 4.8: Two robots holding an object.

namics of the two manipulators can be described by two sets of three ordinary differential equations (refer to the appendix for the derivation of manipulator dynamics) [140]:

$$\begin{aligned} I_1(\Theta_1)\ddot{\Theta}_1 + C_1(\Theta_1, \dot{\Theta}_1) &= \tau_1 - J_1^T F_1 \\ I_2(\Theta_2)\ddot{\Theta}_2 + C_2(\Theta_2, \dot{\Theta}_2) &= \tau_2 - J_2^T F_2 \end{aligned} \tag{4.1}$$

where Θ_i is the 3×1 vector of the joint coordinates of the i^{th} $(i = 1, 2)$ manipulator, ie. $\Theta_1 = [\theta_1, \theta_2, \theta_3]^T$ and $\Theta_2 = [\theta_4, \theta_5, \theta_6]^T$, $I_i(\Theta_i)$ is the 3×3 inertia matrix, $C_i(\Theta_i, \dot{\Theta}_i)$ is the 3×1 vector of nonlinear terms (Coriolis and centrifugal forces), τ_i is the 3×1 vector of the joint torques, J_i is the 3×3 Jacobian matrix relating the velocity of the center of mass of the object to the joint velocities and F_i is the 3×1 generalized force vector, representing the force exerted by the manipulator on the object and the moment about the center of mass. Based on the physical dimensions of the subject, we can calculate the mass and moments of inertia of the human arm based on the normalized anthropometric measurements of Winter [130]. The dynamics of the grasped object is given by:

$$M\ddot{p} = F_1 + F_2 \tag{4.2}$$

where, M is the 3×3 inertia matrix of the object and $p = [x,\ y,\ \phi]^T$, is the 3×1 vector representing the position and orientation of the object with (x, y), being the coordinates of the position of the center of the object. As shown in Figure 4.8, the components of the contact force acting on the object along the inward pointing normals are denoted by F_{1n} and F_{2n} while those tangential to the contact plane with F_{1t} and F_{2t}.

The two manipulators completely restrain the motion of the object. Therefore, the position of the center of mass of the object can be expressed as either a function of Θ_1 or a function of Θ_2. Hence, the position of the center of mass of the grasped object is constrained by the equality constraint:

$$\vec{p}_1(\Theta_1(t)) = \vec{p}_2(\Theta_2(t)) \tag{4.3}$$

where the vector $\vec{p}_1(\Theta_1(t))$ denotes the position and orientation of the object expressed as a function of Θ_1 and $\vec{p}_2(\Theta_2(t))$ is the same vector expressed as a function of Θ_2 (which are both explicitly dependent on time, t). The constraints on the normal component of the contact force are:

$$F_{1n} \geq 0, \qquad F_{2n} \geq 0. \tag{4.4}$$

The tangential forces are subject to constraints due to Coulomb's law of friction. This ensures that the object does not slip from the subject's hands and is given by:

$$|F_{1t}| \leq \mu F_{1n}, \qquad |F_{2t}| \leq \mu F_{2n} \tag{4.5}$$

where μ is the coefficient of friction.

4.2.1 The optimal control problem

Based on the formulation of Chapter 3, we need to define a suitable cost function so that we can find the extremal solution. The "minimum jerk solution" proposed by Flash and Hogan [60] for pure translatory motion in $\mathbb{R}^2$ and later extended by Garvin *et al.* [45] to account for rotational component of motion in the plane, is a purely kinematic model of the system and does not consider the mass of the grasped object or dynamics of the human arm. Since, it is a purely kinematic model, we cannot obtain any force information, *i.e.*, how the left and right arm forces are distributed in the manipulation task. In order to obtain force information, we use the model first proposed by Uno *et al.* [121] which minimizes the rate of change of actuator torques. Since the inputs to the system are the rate of change of actuator torques, we are guaranteed a piecewise smooth solution for the forces and torques based on the methodology of Chapter 3. This is a reasonable assumption, since normal point to point movements by humans are "smooth" without any abrupt changes in forces.

Thus to find an optimal solution, we use the following cost function:

$$\min \ \frac{1}{2}\int_{t_0}^{t_f} (\dot{\tau}_1^2 + \dot{\tau}_2^2)\, dt \tag{4.6}$$

subject to the constraints in Equations (4.1-4.5).

We define the input vector to be

$$u = \dot{\tau}. \tag{4.7}$$

Based on the methodology in Chapter 3, we write the optimal control problem as follows:

The optimal control problem is to

$$\min \ \frac{1}{2}\int_{t_0}^{t_f} (\dot{\tau}_1^2 + \dot{\tau}_2^2) dt$$

subject to the following system dynamics (Equation 4.1):

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

where the above equation is a vector equation with x as a 12×1 vector, $[\Theta_1, \Theta_2, \dot{\Theta}_1, \dot{\Theta}_2]^T$.
and the equality and inequality constraints are:

$$\vec{p}_1(\Theta_1(t)) = \vec{p}_2(\Theta_2(t))$$

$$M\ddot{p} = F_1 + F_2$$

$$F_{1n} \geq 0, \qquad F_{2n} \geq 0$$

$$|F_{1t}| \leq \mu F_{1n}, \qquad |F_{2t}| \leq \mu F_{2n}$$

We can easily write down it's reformulation in the unconstrained variational calculus framework based on the methodology of Section 3.4. Boundary conditions must be specified to solve the optimal control problem defined above. For each movement, we know the start and end positions. Further, the motion starts and ends with zero velocity and acceleration. Thus we have a total of 9 boundary conditions at each point. Since we have a 12-dimensional state space, we can specify 3 additional boundary conditions at each end. For example, it may be meaningful to specify the internal forces at the beginning and the end of motion [140].

4.3 Results

4.3.1 Planar two arm movements

In this section, we present some of our experimental and theoretical results. These results are based on pure translatory tasks for the subject in the horizontal plane. In other words, it is possible for the subject to go from the initial to final position via a pure translation, although the experimental apparatus allows the subject to perform rotations as well. The trajectories and force histories for four subjects are presented and analyzed.

Repeatability

During the experiments, the time taken to complete the motion varied from 0.8 seconds to 1.2 seconds, and there is considerable variation across subjects. Further, the subjects made a systematic error when reaching the targets. This was expected given that no visual feedback of the arms with the handle was provided and because of the parallax in the perception of the target. However, the duration of the motion and the accuracy in reaching the target are of secondary importance to this study since we are primarily interested in the kinematic and dynamic features of the measured trajectories.

We tested the repeatability of the trajectories by comparing the motions performed by the same subject and by different subjects on different trials. Numerical calculations of repeatability measures for different trajectories (within subject and across subjects) are presented in [45]. The velocity histories were found to be repeatable across trials for the same subject and across subjects. The force trajectories on the other hand were not repeatable. This could be due to the fact that the task is over constrained and hence the subjects have no way of controlling the internal force or the “squeeze” force that they exert during a typical manipulation task. We will present experimental evidence in this regard later in the section. Note that in all the experiments studied here, the angular variations (ϕ) and the torques recorded by the force/torque sensors are close to zero and will not be shown in the plots.

Frontal and sagittal plane motions

The trajectories for a representative subject for the $3 \rightarrow 2$ and the $4 \rightarrow 1$ motions (the numbers refer to the target numbers in Figure 4.3) are shown in Figures 4.9 and 4.10. The spread in the sagittal plane trajectories ($4 \rightarrow 1$) can be seen to be less than 0.01 meters while the spread in the frontal plane trajectories ($3 \rightarrow 2$) is less than 0.05 meters. At first sight, it appears that the trajectories are straight lines, but it is clear from Figures 4.9(b) and 4.10(b), that the average trajectory is curved. This is also the case for motions in the opposite direction, $1 \rightarrow 4$ and $2 \rightarrow 3$, as shown in Figure 4.11. In fact, if we compare the trajectories for $1 \rightarrow 4$ and $4 \rightarrow 1$, and similarly for $2 \rightarrow 3$ and $3 \rightarrow 2$, we find that the curvatures have opposite signs. The average curvature for these trajectories is shown in Table 4.1 for subjects S1-S4. While the signs of the curvature for the trajectories was observed to be the same across subjects, there was a significant variability in the actual values. However, the variability for one subject across trials was found to be very low. For example, for subject S2, the standard deviation in the curvature for the $4 \rightarrow 1$ motion and the $3 \rightarrow 2$ motion was observed to be 0.0035 meters and 0.0098 meters(less than 2% of the average value, which makes the data statistically significant). The bias in curvature was not statistically significant in oblique translatory motions. The average trajectories for the $2 \rightarrow 4$ and $1 \rightarrow 3$ motions are shown in Figure 4.12 for two representative subjects. From the individual plots (not all are shown here), we found the $1 \rightarrow 3$ trajectories to be less curved than all other trajectories.

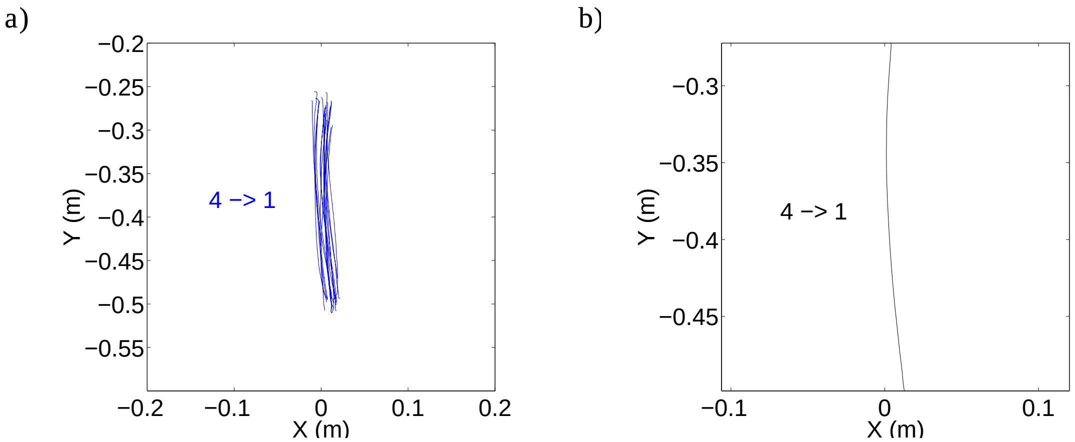

Figure 4.9: a) Observed trajectories and b) Average trajectory for the $4 \rightarrow 1$ motion.

Figure 4.13 shows the velocity profiles for $4 \rightarrow 1$ and $3 \rightarrow 2$ motion averaged over all the trials for a subject. Since the velocity profiles are repeatable, we can conclude some invariant characteristics for other subjects also. As seen from Figure 4.13(a), $\dot{y}$ is bell shaped

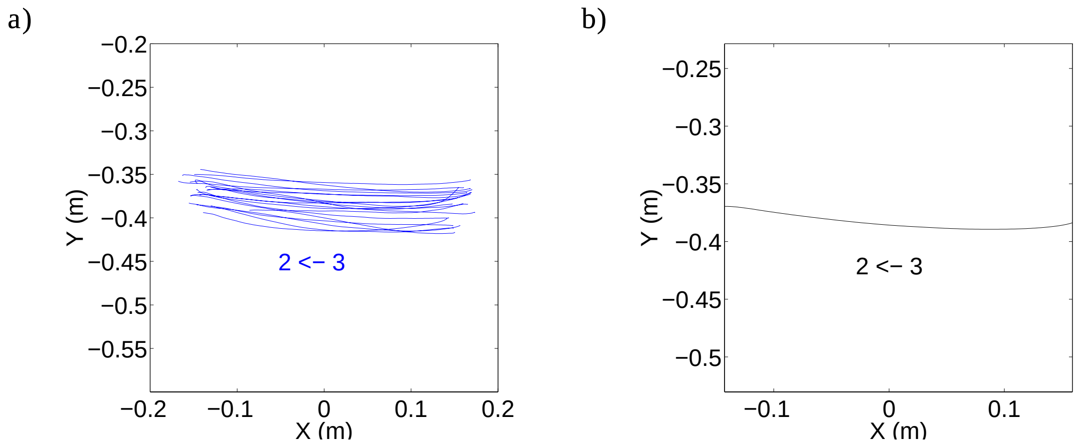

Figure 4.10: a) Observed trajectories and b) Average trajectory for the $3 \to 2$ motion.

Table 4.1: Radius of curvature for motions in the frontal and sagittal planes.

Motion	S1	S2	S3	S4
$2 \to 3$	$-0.53m$	$-0.48m$	$-0.45m$	$-0.46m$
$3 \to 2$	$+0.53m$	$+0.57m$	$+0.61m$	$+0.59m$
$4 \to 1$	$+0.49m$	$+0.50m$	$+0.50m$	$+0.49m$
$1 \to 4$	$-0.52m$	$-0.51m$	$-0.52m$	$-0.51m$

and asymmetric to the left. Thus the acceleratory phase is smaller than the deceleratory phase. This implies that humans, in response to the task, accelerate the object faster towards the onset of motion and after having reached the peak velocity relatively quickly, tend to decelerate to lower velocities after that. It is also interesting to note the sinusoidal shape of $\dot{x}$ which reflects the observed curvature in the $4 \to 1$ motion. These observations are also observed in Figure 4.13(b) for the $3 \to 2$ motion. Similar observations are also made for the $1 \to 4$ and $2 \to 3$ motion though they have not been plotted.

Force history

As the total (resultant) force acting on the object is uniquely determined by the trajectories, we are mainly interested in the internal forces. There are three components to the internal force vector: a moment perpendicular to the plane and two force components. We will denote the internal force in the direction normal to the two palms by F_n, where

$$F_n = F_{1n} - F_{2n}$$

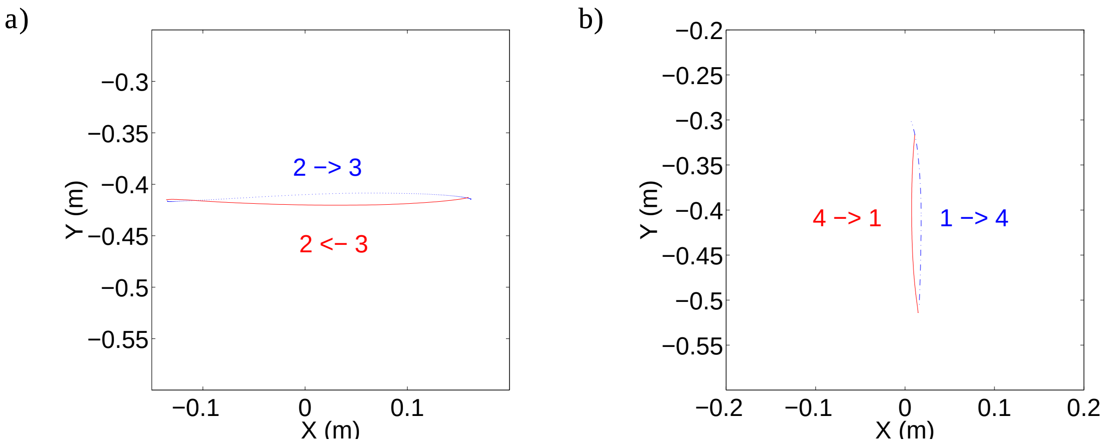

Figure 4.11: Averaged trajectories for: a) the $2 \rightarrow 3$ and $3 \rightarrow 2$ motions; and b) the $1 \rightarrow 4$ and $4 \rightarrow 1$ motions.

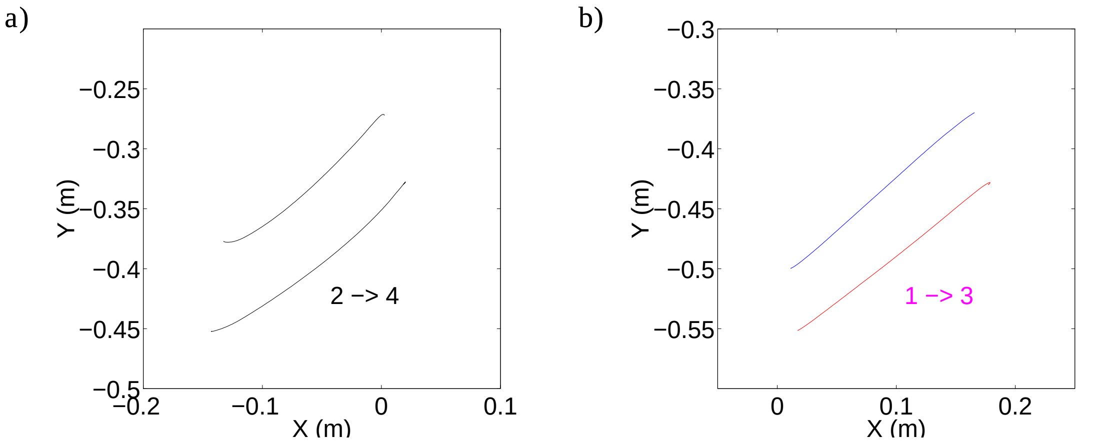

Figure 4.12: Trajectories for: a) the $2 \rightarrow 4$ motion; and b) the $1 \rightarrow 3$ motion.

We will call this component the *interaction force*, using the terminology of [134]. The tangential component and the moment normal to the plane are observed to be negligible in this study and are not discussed further.

The internal forces varied significantly across subjects. While the instructions for the task (moving from one target to another) specify the initial and final positions and velocities, there is no mechanism for specifying the initial and final internal forces. Each subject used a different initial and final internal (grip) force and therefore, there is considerable variability in the magnitudes across subjects. For this reason we did not average the internal force data across subjects.

The internal force history for the motion ($2 \rightarrow 3$) is shown in Figure 4.14(a) for two representative subjects. The solid line denotes subject, S3, and the dashed line denotes subject, S4. The internal force increased as the velocity increased to a peak and then

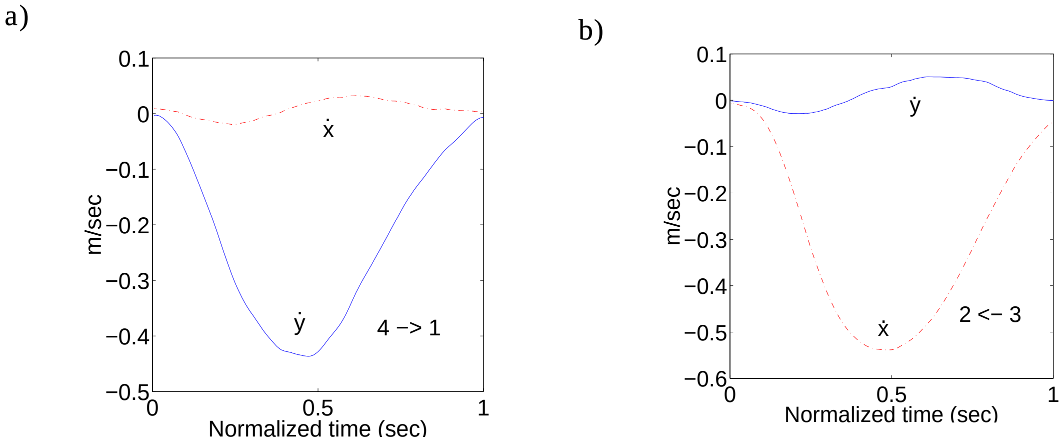

Figure 4.13: Velocity profiles for a) $4 \rightarrow 1$ motion and b) $3 \rightarrow 2$ motion.

decreased toward the end of motion. While the magnitude of the internal force varied across subjects, this general trend was observed in all subjects.

The normal component of the forces exerted on the object are shown in Figure 4.14(b). In the $2 \rightarrow 3$ motion, there was an initial dominance by the left arm followed by a right arm dominance in the second half of motion. In other words, the palm that pushed the object in the direction of acceleration or deceleration dominated. This is consistent with the observations of Reinkensmeyer *et al.* [105] in their study of bimanual, single-degree-of-freedom, wrist movements. However, the forces exerted by the left arm were larger than the right arm. This can be easily explained if we look at the data of [30, 45, 138, 139] that shows that the velocity profiles (not shown here) are always asymmetric. The time taken to go from zero velocity to the maximum velocity is always less than the time taken to decelerate from the peak velocity to rest. In other words, the magnitude of the peak acceleration (to the right), is always lower than the magnitude of the peak deceleration (to the left). Since, in the $2 \rightarrow 3$ motion, the hand that dominates initially (the pushing hand) is the left hand, we expect to find the left arm force to be larger than the right arm force.

The internal force for the sagittal plane motions are shown in Figure 4.15. During the $4 \rightarrow 1$ motion, once again there was an increase in the internal force F_n, as the object was moved from the initial to final position. However, the increase was more significant and occurred through a larger part of the motion than observed for frontal plane motions (Figure 4.14). This initial increase was also seen when the object was moved from $1 \rightarrow 4$ as shown in Figure 4.16. However, in this case the increase was followed by a significant decrease in the internal force. One possible hypothesis to explain the sharp rise and fall in the interaction forces for $4 \rightarrow 1$ and $1 \rightarrow 4$ motions respectively, is based on the assumption

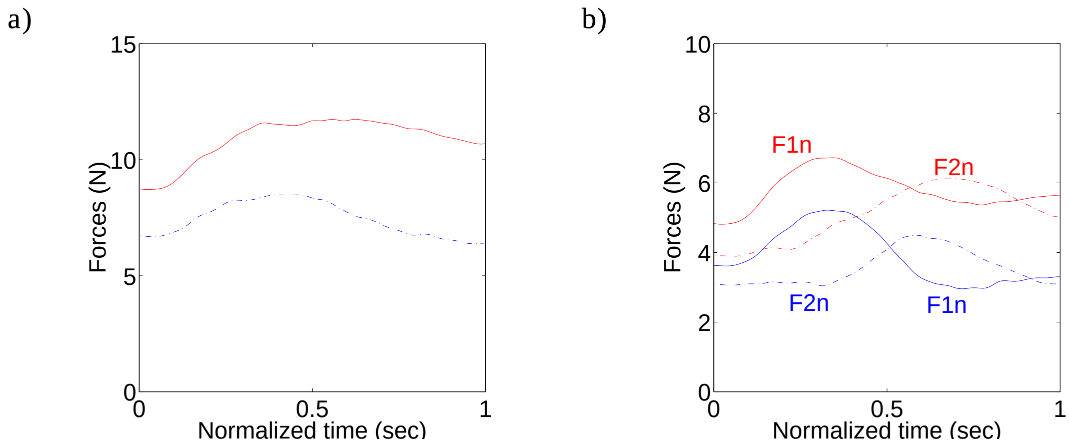

Figure 4.14: a) Internal forces and (b) left and right palm forces for the $2 \rightarrow 3$ motion for two representative subjects.

that if humans tend to apply constant torque about the shoulder joint, then further the object is from the shoulder (as in position 4) smaller will be the interaction force. That would also explain why the interaction force is higher at the end of the $4 \rightarrow 1$ motion.

Thus, there are two general trends observed in Figs. 4.14(a) and 4.16. As mentioned earlier, the internal force appeared to increase as the velocity of the object increased. In addition, the internal force increased as the object was moved closer to the subject and decreased as the object was moved away.

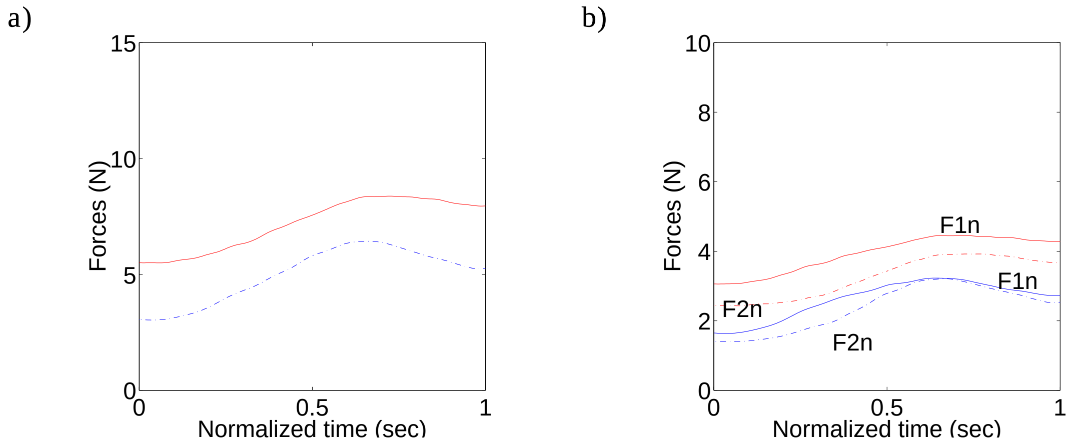

Figure 4.15: a) Internal forces and (b) left and right palm forces for the $4 \rightarrow 1$ motion for two representative subjects.

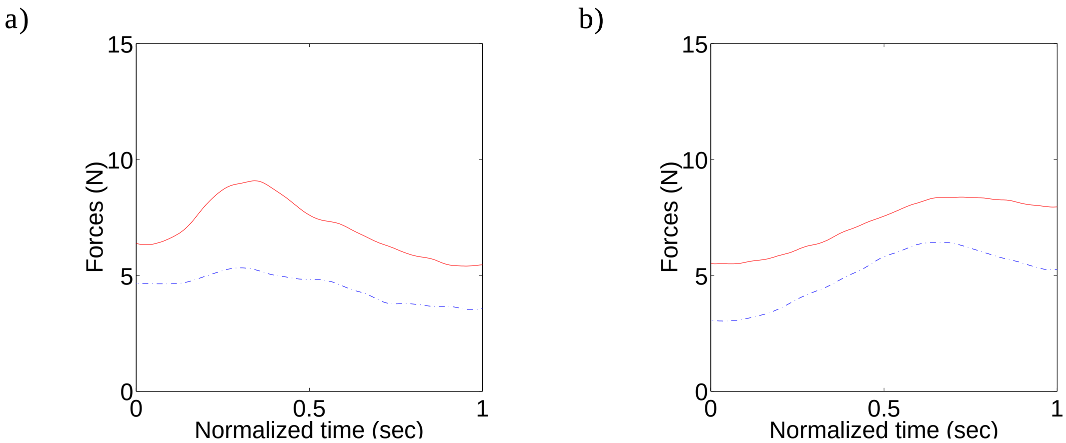

Figure 4.16: Internal forces for (a) the $1 \rightarrow 4$ motion and (b) the $4 \rightarrow 1$ motion for two representative subjects.

General translatory motions

The trajectories in the frontal plane and those in the sagittal plane showed a tendency to curve, and this tendency was consistent across trials with the same subject and across subjects. In general translatory motions, the curvature in the observed trajectories was not consistent across subjects. However, the exception is the trajectory for the motion $1 \rightarrow 3$ which was found to be straight for all subjects. The average trajectory and the average velocity profiles for this motion are shown in Figure 4.17. Because the trajectory was very close to a straight line, the velocities in the x and y direction were the same except for a scaling factor. Neither of these two observations could be made for any other oblique motion. Further the $1 \rightarrow 3$ trajectory and the force distribution appeared to be more repeatable than for other oblique motions. The average internal force profile and the palm force profiles for the $1 \rightarrow 3$ motion are shown for two different subjects in Figure 4.18. Once again, the internal forces and the left and right arm forces show the same trends observed earlier. The internal force increased as the velocity increased and then decreased toward the end of the motion. Also, the dominance of the left arm force in the beginning and the right arm toward the end was clearly seen for both the subjects in Figure 4.18(b).

Computational results and experimental observations

In this section, we compare the experimental observations with the predictions from the computational model using the minimum torque change criterion for the $1 \rightarrow 3$ oblique

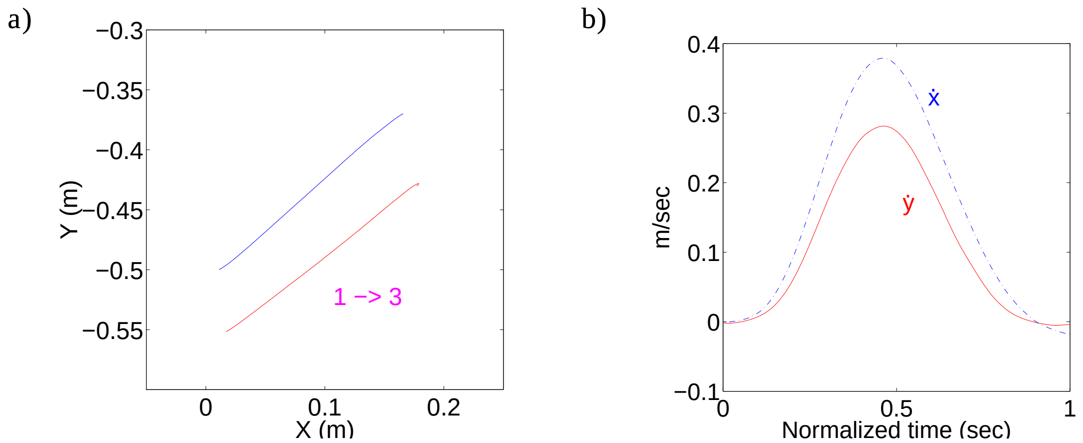

Figure 4.17: a) Average trajectory; and b) average velocity profiles for the 1 → 3 motion.

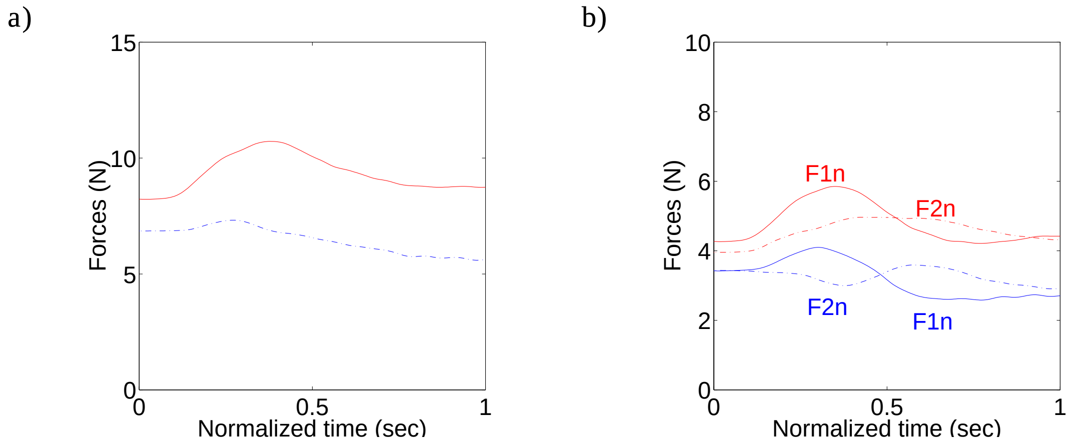

Figure 4.18: a) Internal forces and (b) left and right palm forces for the 1 → 3 motion for two representative subjects.

motion. Instead of using the averaged results, we randomly chose an experimental trial[1] for comparison. The boundary conditions (initial and final position, velocity, acceleration and interaction force) for the computational model are calculated from the experimental data.

The predicted and observed trajectory is shown in Figure 4.19(a) and the velocities are shown in Figure 4.20. The discrepancy between the theoretical predictions and the experimental data is small compared to the variance of the data (not shown in the plots). However, this is not true of the interaction force shown in Figure 4.19(b). Though there is a pretty good agreement towards the beginning and end of the motion, there seems to be a larger discrepancy in the interaction forces towards the center of motion. We

[1] Since we need boundary conditions to generate the solution to (4.1-4.6), we use experimentally observed end conditions rather than averaged boundary conditions.

expect the interaction forces to depend on the acceleration of the object rather than its velocity. This is clearly not the case, as the bell-shaped velocity profile leads to lower accelerations towards the center of motion.[2] Clearly higher interaction force towards the center of motion suggests a velocity-dependent nature of the interaction forces.

The predicted variation of the interaction force is very small. However, the experimental data shows that the interaction force first increases and then tails off. In particular, the computational model does not capture the increase in interaction force with an increase in object velocity [35].

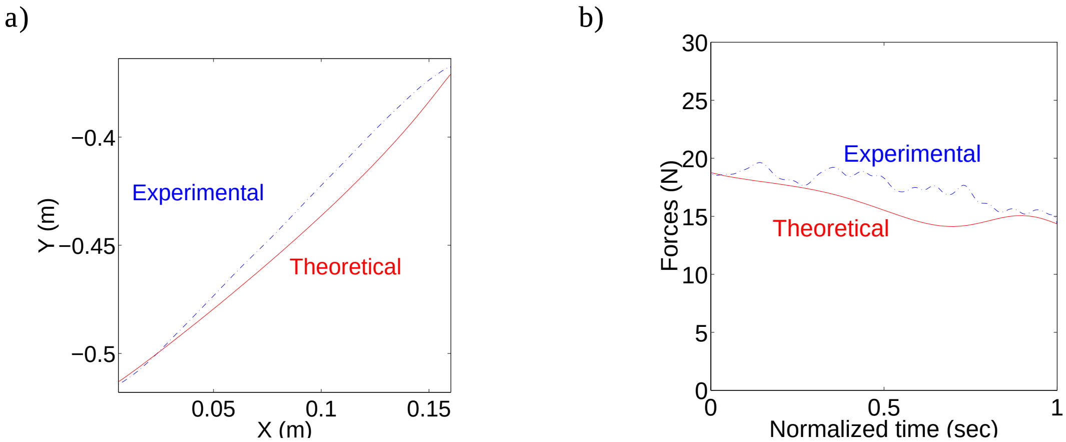

Figure 4.19: Theoretical and experimental a) Trajectory and b) Interaction force for motion $1 \to 3$.

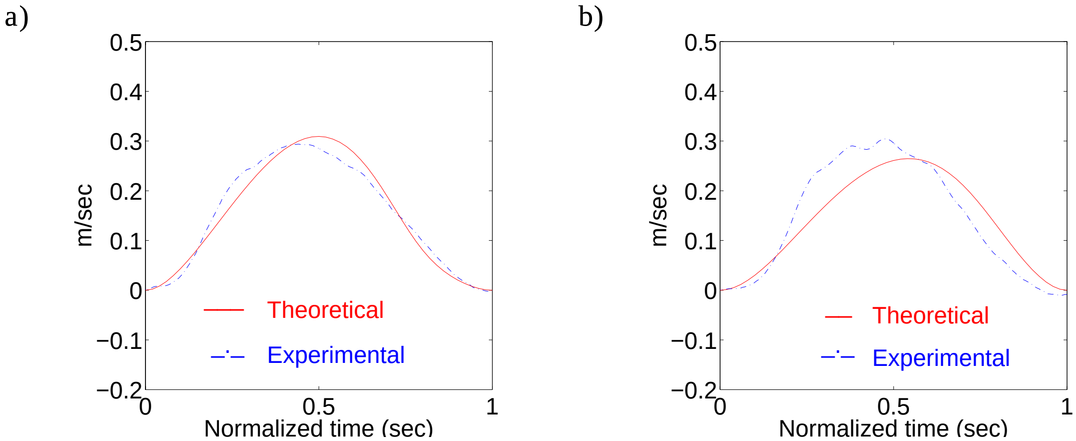

Figure 4.20: Theoretical and experimental velocities: a) $\dot{x}$ and b) $\dot{y}$ for motion $1 \to 3$.

[2] Thus, we expect the interaction forces to drop towards the center of motion.

Effect of increasing the weight of the grasped object

If the object is made heavier, one expects the internal force to increase so that the ratio of the tangential to the normal force at each palm is less than the coefficient of friction. Figure 4.21 shows the effect of increasing the mass of the object from 0.85 to 1.95 Kg. for two representative motions for one subject averaged across all trials. Increasing the mass of the grasped object resulted in an increase in the internal force but there was no significant change in the trajectory.

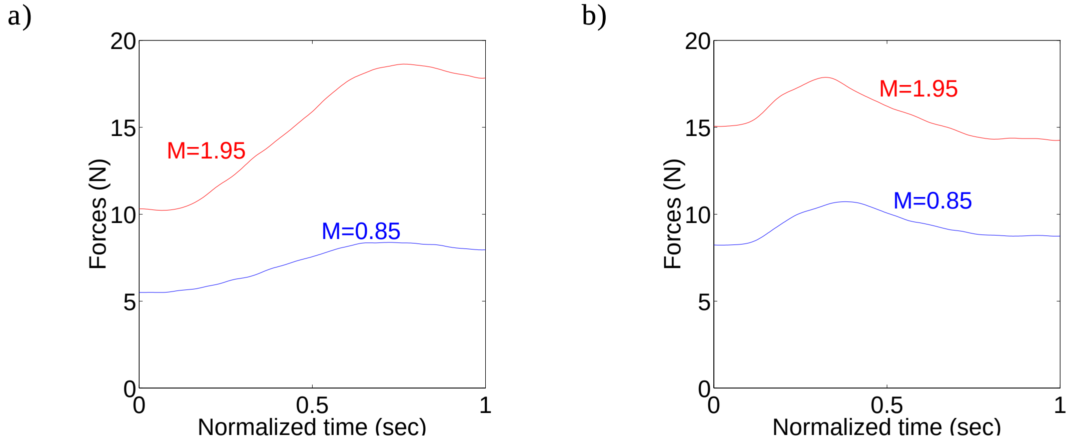

Figure 4.21: Internal force plots for motion a) $4 \rightarrow 1$ and b) $1 \rightarrow 4$ for two representative subjects.

4.3.2 Vertical two arm movements

Repeatability

In vertical two arm studies, we studied the trajectories for movements in the frontal and sagittal plane. All movements in the frontal and sagittal plane had negligible rotational component. The subjects were not given any specific instructions regarding rotating the object during the manipulation task. The results analyzed offline showed that the subjects were not rotating the object when they carried out the manipulation task. This was verified by computing the rotation matrix at each data point. Figure 4.22 (a) and (b) show the cosine of angle of rotation about the axis corresponding to the axis of rotation for the $6 \rightarrow 7$ and $7 \rightarrow 6$ motion respectively. The data is plotted for all the trials for a representative subject. As seen from the figure, the value varies by less than 0.05 around 0. A similar observation was also made for other subjects indicating that the subjects moved the object with negligible rotation.

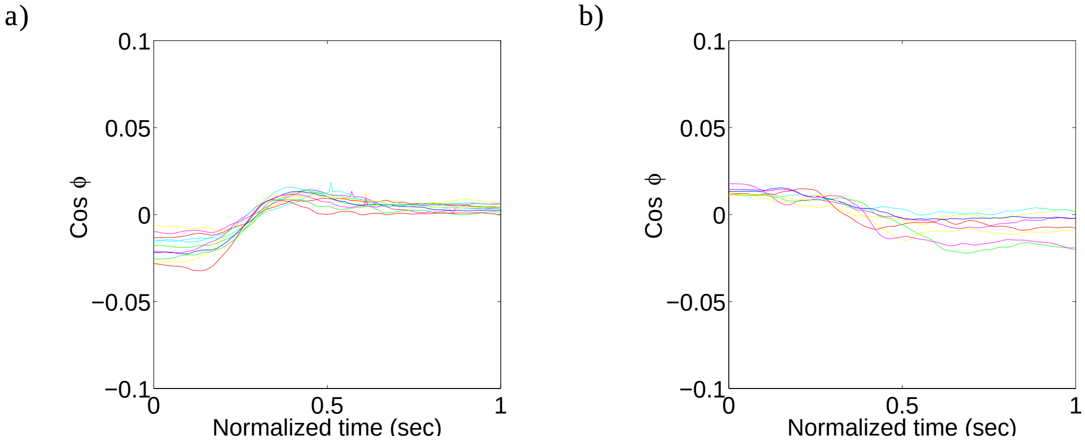

Figure 4.22: Plots showing the cosine of the angle of rotation for a) 6 → 7 and b) 7 → 6 motion.

We observed that all the trajectories and velocity profiles were highly repeatable across trials and across subjects for all the investigated motions. Hence it is meaningful to average the trajectories and velocity profiles. The force profiles on the other hand are not repeatable as in the case of planar arm movements. We hypothesize that a similar explanation as planar arm movements holds in vertical arm movements also, *i.e.*, since the task is over constrained the subjects had no way of controlling the "squeeze" force. However, the shape of the force profiles are identical across all the subjects for all the trials. Since the significant components of each movement is the $X - Z$ plane, we have not plotted the trajectories and forces in the Y direction for most of the plots.

Trajectories and velocity profiles

In Figure 4.23, we show the observed trajectories for a particular subject and then the average trajectories over all trials for that subject. The trajectories, $5 \rightarrow 8$ and $8 \rightarrow 5$ are significantly different than the trajectories observed in planar two arm movements. It is clearly seen from Figure 4.23(a) that the trajectories for all the trials for the $5 \rightarrow 8$ motion by the subject have a leftward curvature and hence the average trajectory is also curved. However, unlike planar arm movements, the trajectories for the $8 \rightarrow 5$ movement are straight. This result is also in agreement with the findings of Atkeson and Hollerbach [6]. This could be partly explained by the fact that during the vertical arm movements there is a greater effort by the subject to lift the grasped object in upward motion and a straight path may not necessarily be the optimal trajectory. However, for downward movements, less effort is required as the motion is assisted by gravity. This is also reflected in the individual left and right arm forces and the interaction forces for two representative subjects in Figure 4.27 and Figure 4.28. For frontal plane motion the trajectories are straight lines when projected in the $X - Z$ plane which is the significant plane of motion. These results are also in agreement with the findings of Atkeson and Hollerbach [6]. This is clearly observed from the trajectories for the subject in Figure 4.24(a) and the average trajectories in Figure 4.24(b). In Figures 4.25(a) and 4.25(b), we show the three-dimensional plots of the trajectories in the frontal and sagittal plane respectively.

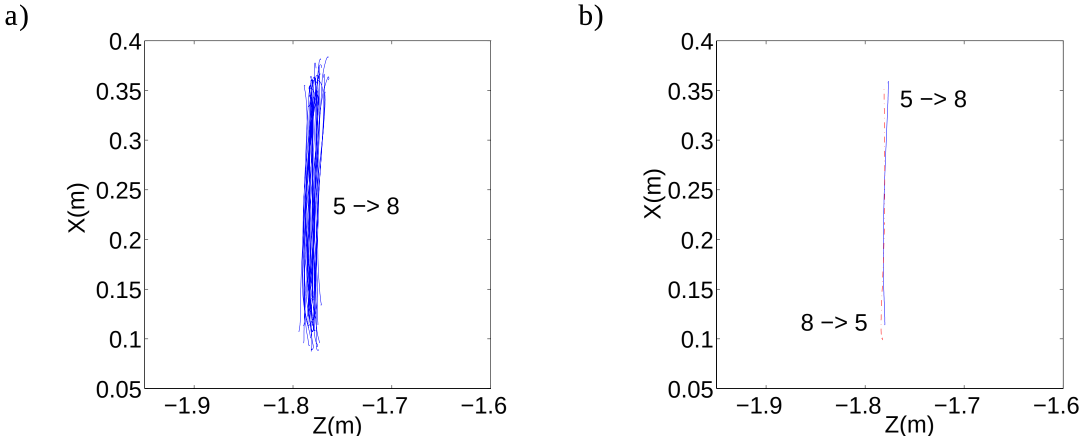

Figure 4.23: a) Observed trajectories for $5 \rightarrow 8$ motion and b) Average trajectories for $5 \rightarrow 8$ and $8 \rightarrow 5$ motion.

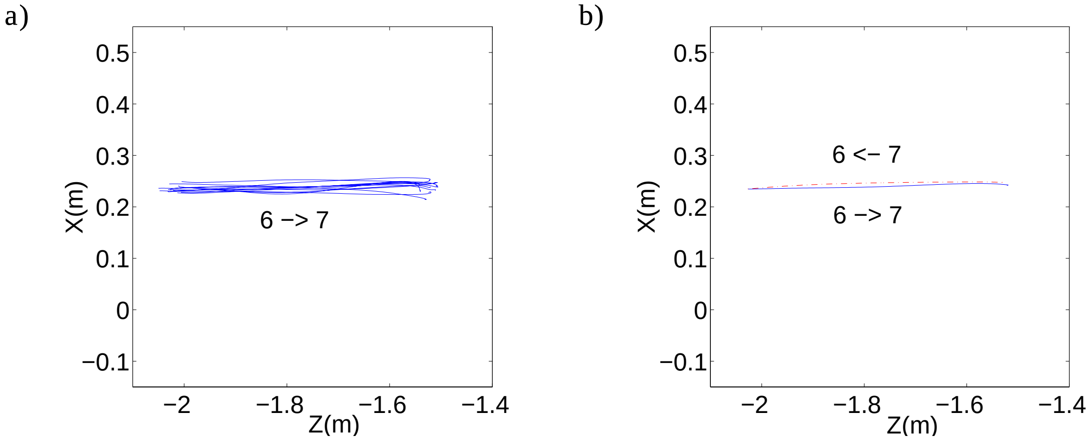

Figure 4.24: a) Observed trajectories for $6 \rightarrow 7$ motion and b) Average trajectories for $6 \rightarrow 7$ and $7 \rightarrow 6$ motion.

The velocity profile for $5 \rightarrow 8$ and $8 \rightarrow 5$ motion are as shown in Figure 4.26(a) and for $6 \rightarrow 7$ and $7 \rightarrow 6$ motion in Figure 4.26(b). It is clearly seen that based on the observed

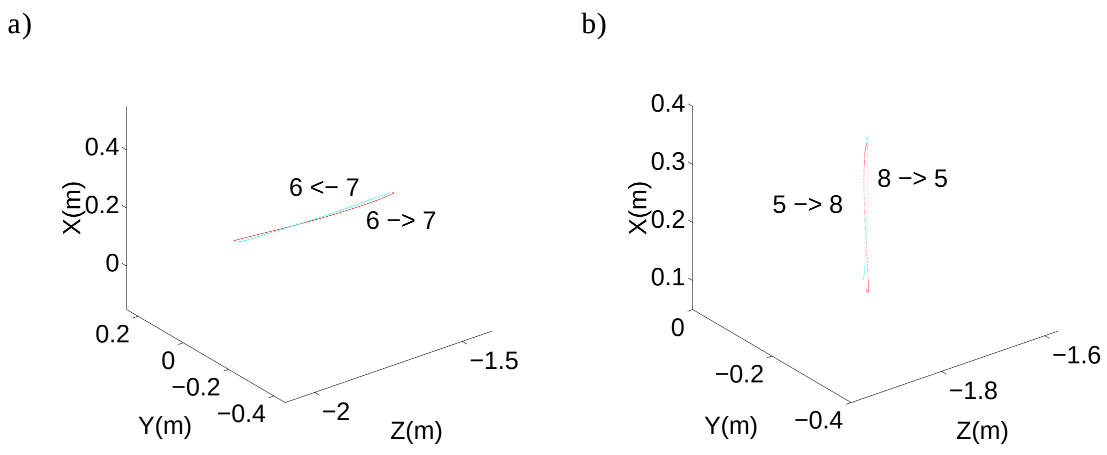

Figure 4.25: Three dimensional trajectories for the a) frontal and b) sagittal plane motion.

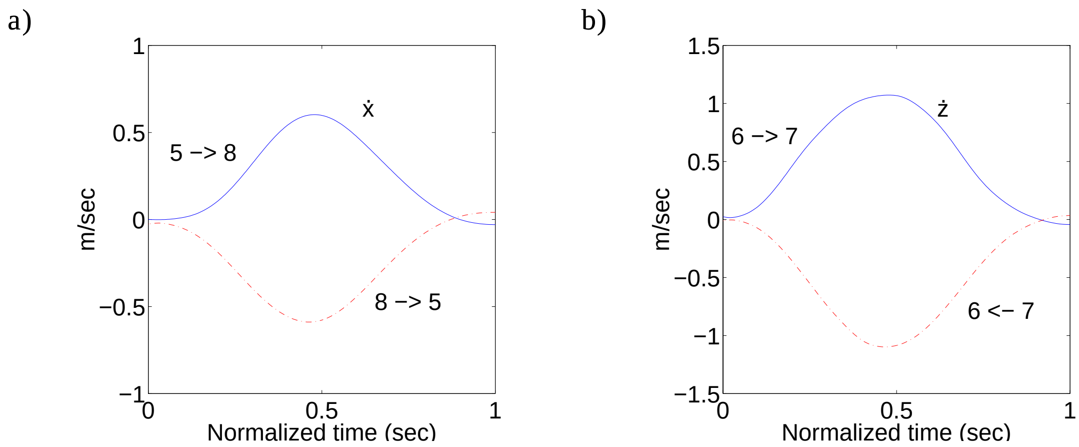

Figure 4.26: Average velocity profiles for a) $5 \rightarrow 8$, $8 \rightarrow 5$ motion and b) $6 \rightarrow 7$ and $7 \rightarrow 6$ motion.

near straight line trajectories, the velocity profiles for these motions are more symmetric that those in planar arm movements. If we align the peaks of the $5 \rightarrow 8$ and $8 \rightarrow 5$ motion by reflecting the velocity profile for the $8 \rightarrow 5$ motion along the Z-axis, we observe that the $8 \rightarrow 5$ motion leads the $5 \rightarrow 8$ motion. Also the time taken to complete the $5 \rightarrow 8$ motion is marginally greater than the $8 \rightarrow 5$ motion which indicates that the distance traveled during the $8 \rightarrow 5$ motion is shorter than the $5 \rightarrow 8$ motion since the area under the velocity-time curve for the $5 \rightarrow 8$ motion is lager than the $8 \rightarrow 5$ motion. This supports the observation that the trajectory for the $8 \rightarrow 5$ motion is nearly straight while that of the $5 \rightarrow 8$ motion is curved to the left.

Force history

The interaction force profiles for the $8 \to 5$ and $5 \to 8$ motion are as shown in Figure 4.27(a) and Figure 4.28(a) respectively. It is clear from the figures that the forces do not exhibit the characteristics of planar movements as the motion is in the vertical plane. The force profiles are sinusoidal and start and end with roughly the same values. In each of these plots, we have plotted the results for two representative subjects. Clearly, the interaction forces are higher for the upward movement than downward movement as in the $5 \to 8$ movement, work is done against gravity. Though the shapes of the interaction force profile is repeatable, the actual values are not. This is because the manipulation task is over constrained and that makes the control of "squeeze force" difficult.

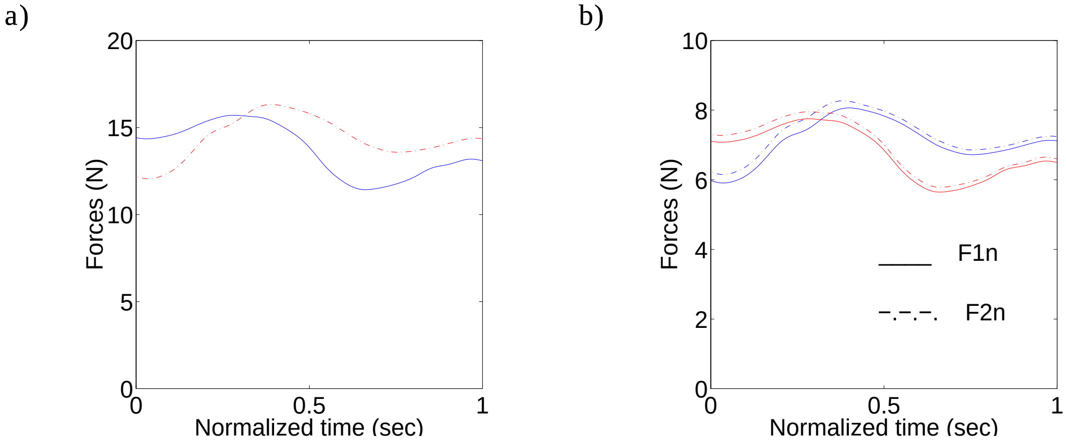

Figure 4.27: a) Internal forces and (b) left and right palm forces for the $5 \to 8$ motion for two representative subjects.

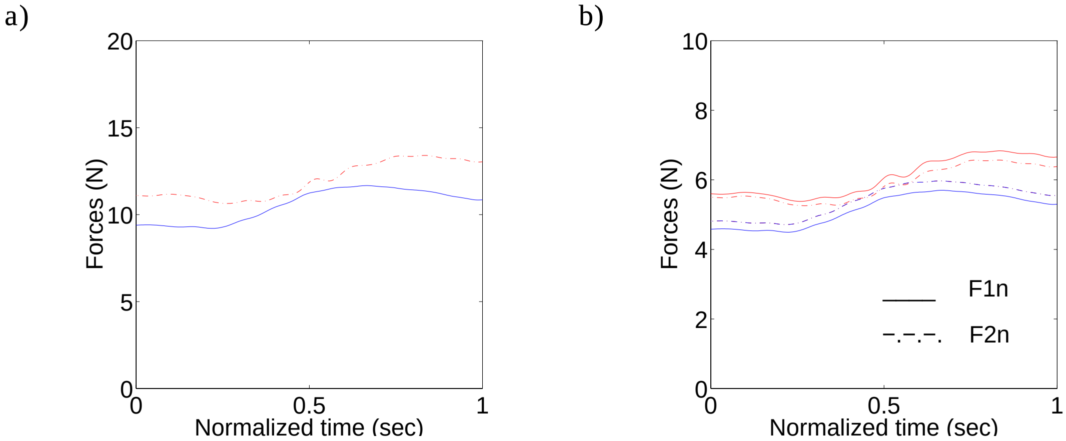

Figure 4.28: a) Internal forces and (b) left and right palm forces for the $8 \to 5$ motion for two representative subjects.

The left and right arm force trajectories for the $5 \to 8$ and $8 \to 5$ motion are shown in Figure 4.27(b) and Figure 4.28(b) respectively. It is seen that the force profiles for the individual arms are similar for both the subjects and there is no dominance of one arm over the other during the execution of the required task. This is expected since the movement is in the vertical plane and parallel to the body with the object grasped approximately mid-way between the two shoulders.[3]

The internal forces and left and right arm forces for the $6 \to 7$ motion are shown in Figure 4.29 for two representative subjects. The internal forces show a significant rise towards the beginning of motion as is observed in other motions. This is primarily due to the initial reaction which could be caused due to the fear of loosing the grasped object. The left and right arm forces show that the left arm is dominant towards the initial part of motion and the right arm takes over in the latter half. The intersection of the two force profiles as seen in Figure 4.29(b) is midway for the first subject and asymmetric for the second subject. However, in both the cases the dominance of left and right arm is clearly evident.

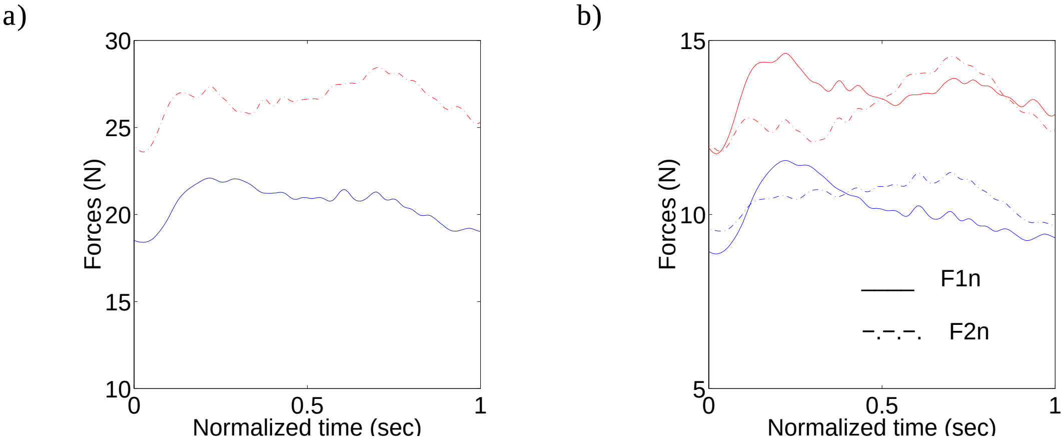

Figure 4.29: a) Internal forces and (b) left and right palm forces for the $8 \to 5$ motion for two representative subjects.

4.4 Conclusions and future work

There are many trajectory generation and motion planning schemes that have been proposed for cooperating robot arms in the literature. However, there is no clear rationale for selecting one method over another.

[3] The dashed force profiles are for the right hand while the solid lines represent the left hand.

The first thing that is worth noting is that human trajectories and velocities are surprisingly repeatable both in planar and vertical arm movements. This suggests that there is a definite strategy used by humans in manipulation tasks. The near-straightness of the trajectories and the smoothness of the velocity profiles suggest optimality by some measure. The increase in internal forces with an increased external load (weight) for planar movements suggests that the optimality criterion must incorporate the force distribution.

However, there are several observations that are difficult to explain with a simple optimality criterion. First, trajectories in the sagittal and frontal plane for planar motion shows a curvature that is consistent across subjects. This is particularly significant for manipulation in the sagittal plane. If the two arms are identical, there is no physical explanation for the asymmetry induced by this curvature. Clearly, the two arms are not identical. However there is no simple cost function that can model this asymmetry. It is not completely clear why the trajectories in the planar studies exhibit curvature while those in vertical arm movements are nearly straight. One way to possibly account for this is to compare the eigenvalues of the inertia matrix of the manipulandum and the two arms in the task space. Comparing the maximum and the minimum eigenvalues of the inertia matrix for the manipulandum and the two arms gives a ratio of approximately 1 indicating that we cannot neglect the inertia of the manipulandum in the experiments. This could partly account for the observed curvature in the frontal and sagittal plane trajectories for planar arm movements. Similar curvature is not present in vertical arm movements as the grasped object is freely manipulated in space. Another significant observation that is difficult to explain using physical principles is the increase in internal force with an increase in velocity. One expects internal forces to be larger for increased accelerations. However, this is not the case.

It is tempting to compare this work to the work on human grasping [42, 64, 65, 105]. It is worth noting that the internal forces in the experiments of [42, 64, 65, 105] are very high compared to the resultant force required to accelerate it. While the ratio of the grip force (equal to half the internal force) to the load force (equal to half the resultant force) in these papers varies from 3 to 7, the ratio in the bimanual tasks reported here is only around 1. For example, the peak internal force (F_n) for the $6 \rightarrow 7$ motion for subjects S3 and S4 for planar movements was observed to be between 8N and 11N. The peak resultant force (F_R) was around 10N of which the weight of the grasped object accounted for 8.33N. It is also worth noting that weight accounts for a much larger fraction of the resultant force in our experiments.

Another difference between the experimental paradigm of this work and that used in previous studies has to do with the coupling between the grasping and the manipulation functions. In [42, 64, 65], the control of the grasp forces can be completely decoupled from the manipulation task because the joints and muscle groups used in manipulation are completely decoupled from those used in grasping. This is not true in our work and in [105], where the same effectors (palms) are responsible for holding the object and manipulating it.

We presented a computational model for planar arm movements derived from the minimum torque-change model for single-arm reaching tasks. The model predictions and the experimental findings were roughly consistent as far the trajectories and the velocities are concerned. However, the interaction forces were consistently different. Nevertheless if the asymmetry in the human neuromuscular system can be reflected in this model, it may serve as a model for predicting trajectories and force distribution in manipulation tasks. It can also be used for generating trajectories for synthetic human models in computer graphics and for motion planning in robotic systems. Finally, it is superior to previous trajectory planning models in that it explicitly incorporates the distribution of forces between the two arms and frictional constraints in friction-assisted grasps.

Our results in vertical arm movements are significantly different that those observed in the planar case. The vertical arm trajectories for upward movements exhibit curvature while the downward movements are straight. The force profiles for these motions also exhibit different internal forces based on the manipulation task. This can be partially explained by the fact that in one case the work is done against gravity ($5 \rightarrow 8$ motion) while in the other case, it is along the direction of the gravitational force ($8 \rightarrow 5$ motion). Frontal plane movements also exhibited near straight line trajectories and the force profiles are similar to those in the planar case.

In this chapter, we have provided a detailed analysis of planar and vertical manipulation tasks involving friction assisted grasps. The long term goal of this study is to see (a) if human trajectory formation and the distribution of forces between the arms can be explained by some optimality criterion; and (b) if such criteria can be used in the control and coordination of robotic arms. This study may also help improve our understanding of how humans use their arms in bimanual tasks. This is potentially useful for the design of haptic interfaces (and human-machine interfaces in general) in which two arms are required.

Chapter 5

Multiple robot motion planning

We addressed the mathematical and numerical techniques for solving general motion planning problems in Chapter 3 and a specific application to modeling cooperative human manipulation tasks in Chapter 4. In this chapter, we address the problem of motion planning for cooperating mobile robots such as the experimental system shown in Figure 5.2. We are interested in solving motion planning problems such as the ones shown in Figures 5.3, 5.4 and 5.5. The problem of nonholonomic motion planning is very well addressed by Li and Canny [80]. They address some of the methodologies and control algorithms which are a departure from the traditional methodologies in motion planning, for example [24, 76]. There are three levels of indeterminacy in the motion planning problem. First, given a start and a goal configuration, the trajectory of the object will need to be determined. Even if this trajectory is *a priori* specified, it may have to be modified to avoid obstacles as shown in Figures 5.3, 5.4 and 5.5. Second, since a mobile manipulator is kinematically redundant [103, 128], the object motion must be decomposed into displacements of the joints of the manipulator and rotations of the wheels of the mobile cart. Finally, the system is statically indeterminate. Therefore, the motion planning task also involves distributing the load between the cooperating mobile manipulators in an optimal fashion.

Depending on the choice of the cost functional, we can resolve one or more levels of indeterminacy. For example, by minimizing a cost functional that is derived from the velocity of the system, we can obtain a plan that specifies the velocities of the object, the manipulators and the platforms [31, 33]. If the platforms and manipulators are equipped with good velocity controllers, such a kinematic plan can be used to plan the trajectory of the system. However, as we show later, such plans may result in discontinuities in the input velocities and may be impractical. If the cost functional depends on the actuator forces and the force distribution, as shown later, the resulting plan resolves indeterminacies at all three levels. Such plans are only meaningful for systems in which torque (force) control is possible. Since the discontinuities in the velocity profiles leads to errors due to slippage and dead-reckoning, it is be meaningful to obtain smoother velocity profiles which are derived by minimizing the actuator forces based on the methodology in Chapter 3. These plans will naturally have discontinuous actuator force/torque profiles.

Here we consider the dynamic modeling of cooperating mobile manipulators with nonholonomic velocity constraints. In addition, we impose suitable unilateral constraints in order to model joint limits and avoid collisions with obstacles. We first adopt a kinematic cost functional in which we use the L_2 norm of the wheel velocities to minimize the distance traveled by the powered wheels. In order to facilitate stable grasping, we constrain the relative position and orientation between the two platforms so that the object is within the reachable workspace of each manipulator. Further, because the dynamics of the mobile manipulators may not allow paths with high curvature, we limit the radius of curvature of the platform trajectories. In our experimental system, the DC servo motors powering the wheels are velocity controlled. Thus, a cost function which guarantees a continuous velocity profile is meaningful to prevent slippage and errors due to dead-reckoning which can arise with discontinuous velocity profiles. Such a cost function (as the norm of actuator forces/torques) will indirectly minimizes the energy required for actuation and the magnitudes of the forces over time.

5.1 Experimental system

In our experimental testbed shown in Figure 5.2, two mobile platforms can be used to mount robot manipulators to cooperatively grasp and transport an object. The TRC platform is a nonholonomic cart with two actuated degrees-of-freedom. The TRC platforms are car-like computer controlled nonholonomic vehicles which have two DC-motor actuated wheels and four casters, one at each corner. The linear and angular velocities of the drive wheels can be controlled through a RS-232 serial line with a bandwidth of 15Hz. The drive wheels are equipped with encoders. Each platform is controlled by an IBM PC 486 compatible computer. The wheel rotations are used with a dead-reckoning scheme to estimate the position and orientation of the platform for real time control. The relative position and orientation between the two platforms is measured through an instrumented four-bar-linkage whose schematic is shown in Figure 5.1. As shown in the figure, each joint of this linkage is equipped with an encoder, but this information is only used for analyzing the performance of the system; it is not used for real-time control. Thus the readings of the encoder can be used to obtain accurate separation between the two platforms. This system is used to obtain the distance between the two platforms and compare it with the desired separation. Finally, it is important to note that currently there is no way of experimentally measuring the absolute position and orientation of the two platforms. The mobile platforms enable appropriate (and even optimal) positioning and configuring before grasping, and possible reconfiguration, if necessary. For example, in the scenario shown in Figure 5.3, a system of two mobile manipulators, when faced with an obstacle, can "step aside" by performing a maneuver that is roughly analogous to parallel parking and continue on their trajectory. In Figure 5.4, two mobile manipulators change their formation from a "march in a single file" (stage 1 in the figure) to a "march abreast" (stage 2 in the figure) when confronted with an obstacle in front of them. In this way they can follow the desired trajectory for the object while reconfiguring, rather than having to stop, retrace their steps and change their formation. The team of manipulatory agents can reconfigure to a single-file formation to squeeze through a narrow corridor, as shown in Figure 5.5.

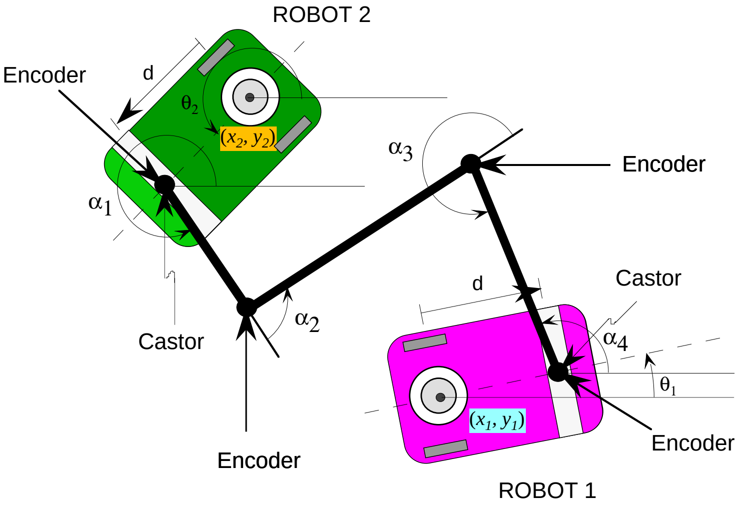

Figure 5.1: Schematic of the four-bar linkage connecting the two platforms.

In order to allow each mobile manipulator to function independently, and to cooperatively perform manipulation tasks, each is controlled by an independent controller and they are loosely coupled depending on the nature of the task [1]. Each agent is equipped with force/torque sensors and can sense the actions of the other agent. During a cooperative manipulation task, the only information that is exchanged at servo rates is the force information [8]. In a typical maneuver, each agent is given a nominal trajectory for the object. Each platform follows an appropriate trajectory while the manipulators maintain a stable grasp of the object by exerting the appropriate forces [132]. In this way the grasp is maintained even if the platform trajectories deviate from the prescribed trajectory.

5.2 Obstacles as constraints

To formulate the problem of planning an obstacle-free path, we represent obstacles as unilateral constraints in the state space. We need to be able to compute gradients of these constraints along different directions in the state space to solve the obstacle avoidance problem in the optimal control framework. In order to do this, we first define a suitable measure of the distance between the robot system in a general configuration and any obstacle. We then develop expressions for: (a) the distance function; and (b) gradients of this function with respect to the configuration space variables.

Efficient algorithms for computing distances between two compact convex sets in R^n are described in [47, 81]. However, when the sets intersect, most algorithms ([47] included) do not compute the extent of the intersection. In order to compute the gradient, it is necessary to have a continuous distance function that returns a “negative distance” when the objects intersect, a zero distance when the objects touch and a positive distance when the objects do not intersect. We model our objects as compact sets in R^2. *Negative* distance or the *penetration* distance between two objects is defined to be the smallest relative rigid body translation in R^2 which will ensure that the two objects have no interior points in common. We simplify the problem of computing the distance and its gradient by using a spherical representation [27] for the links of the robot system and by representing the obstacles with convex polygons. As shown in Figure 5.6, each link is represented by circumscribing circles and the robot is a union of these circles. By decomposing each link into smaller polygons, we can describe the robot by smaller circumscribing circles and obtain a description with a finer resolution. The advantage of this approach is that the distance between a circular robot and a polygonal obstacle can be easily computed. In the present problem, we model the robots to be circular[1] and assume the obstacles to be polygonal of varied shapes and sizes [29].

The distance between a robot link (a circle, S_A) and an obstacle (a polygon, S_B) can be written in terms of (x, y), the coordinates of the center of S_A (point C), and θ, the

[1] Equivalently we can assume that the robots are polygonal and the obstacles are circular. But we have chosen to approximate the robots since there are fewer robot links than obstacles in most problems.

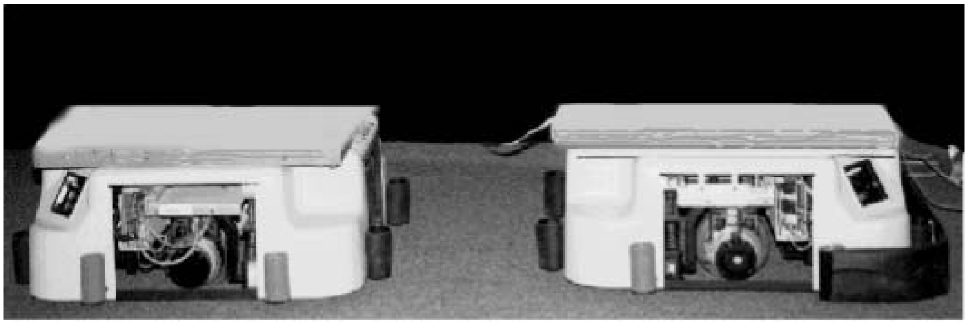

Figure 5.2: A team of two cooperating mobile platforms.

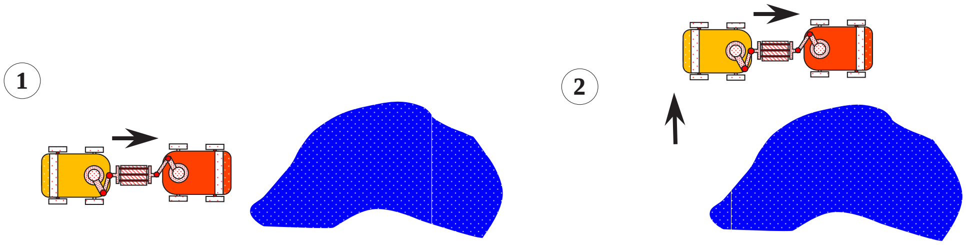

Figure 5.3: Executing a “parallel-parking” maneuver to circumvent an obstacle.

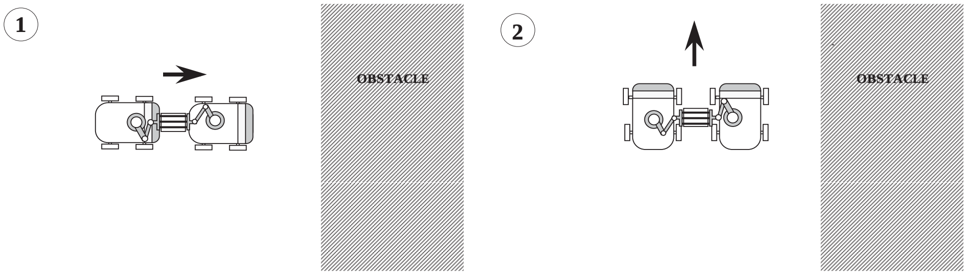

Figure 5.4: Making a sharp turn when confronted with a wall. The agents reconfigure from a single file formation to a “march abreast” formation.

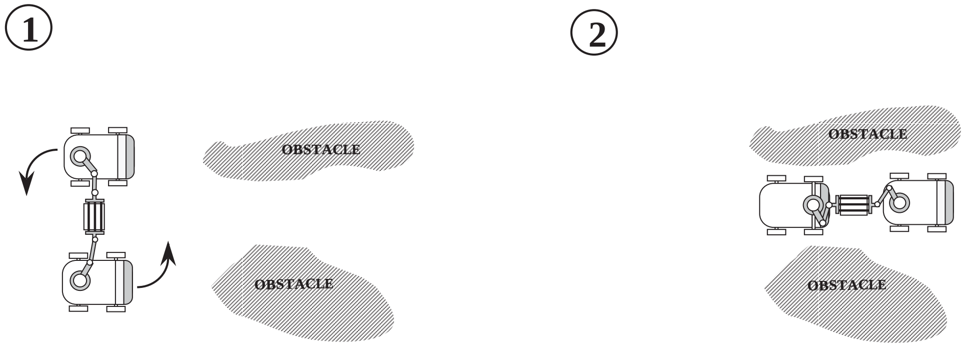

Figure 5.5: Reconfiguring from a "march abreast" formation to a "single file" formation while maneuvering through a corridor.

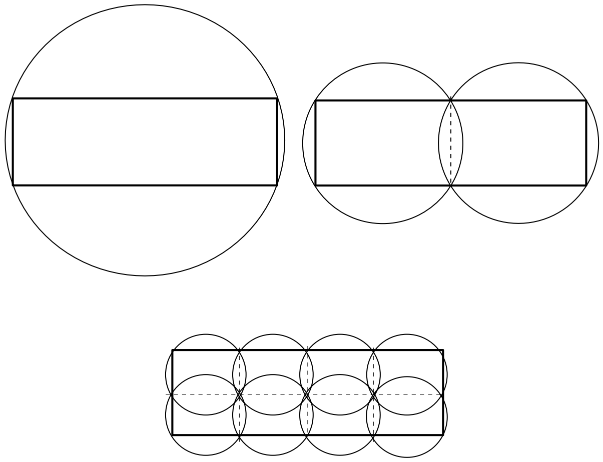

Figure 5.6: Modeling a convex link with a desired resolution.

orientation of the link (see Figure 5.7). Let the point on S_B that is closest to C be denoted by O with coordinates (x_o, y_o). The distance, $d(x, y, \theta)$, and its gradient, depend on whether O is on an edge (Case 1) or at the vertex (Case 2).

In Case 1, $d(x, y, \theta)$ can be written as:

$$d(x, y, \theta) = |-(x - x_o)\sin\alpha + (y - y_o)\cos\alpha| - r_A$$

where r_A is the radius of the robot link. d_x, d_y and d_θ, the partial derivatives of $d(x, y, \theta)$ with respect to x, y and θ respectively, are given by:

$$\begin{aligned} d_x &= -\sin\alpha \; sgn\{-(x - x_o)\sin\alpha + (y - y_o)\cos\alpha\} \\ d_y &= \cos\alpha \; sgn\{-(x - x_o)\sin\alpha + (y - y_o)\cos\alpha\} \qquad (5.1) \\ d_\theta &= 0 \end{aligned}$$

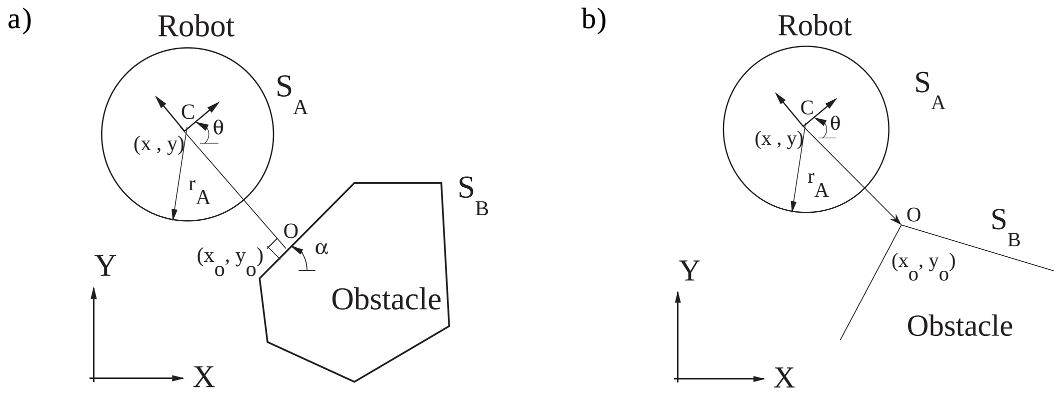

Figure 5.7: Distance between a circular robot and convex polygon: (a) Case 1 (b) Case 2.

In Case 2, the closest point on the obstacle to the robot is the vertex of the polygon. Thus $d(x, y, \theta)$ is given by:

$$d(x, y, \theta) = \sqrt{(x - x_o)^2 + (y - y_o)^2} - r_A$$

and the gradient is given by:

$$\begin{aligned} d_x &= (d + r_A)^{-1}(x - x_o) \\ d_y &= (d + r_A)^{-1}(y - y_o) \qquad (5.2) \\ d_\theta &= 0 \end{aligned}$$

When the center of S_A belongs to S_B, the negative distance is given by:

$$d(x, y, \theta) = -| - (x - x_o) \sin\alpha + (y - y_o) \cos\alpha| - r_A \qquad (5.3)$$

Both computations, (Case 1 or Case 2) and (5.3), return the distance, $-r_A$, when C lies on the edge of the polygon.

5.3 Dynamic model

5.3.1 System description

In our experimental testbed shown in Figure 5.2, two mobile platforms can be used to mount robotic manipulators which can cooperatively grasp and transport an object. The platform is a nonholonomic cart with two actuated degrees-of-freedom. The task is to obtain a trajectory and the history of actuator forces that will move the system from an initial configuration to a desired final configuration, while enabling the two manipulators to maintain contact with the object. There are several assumptions that we make based on the physical characteristics of our system.

- The end effectors are flat palms that can only push and not pull as shown in Figure 5.8. Thus the contact forces are nonnegative and subject to frictional constraints. Further, we generate motion plans that require a zero moment about the center of the palm for each manipulator.

- Since the platforms operate in an essentially two-dimensional world, we assume that all rigid bodies "live" in the special Euclidean group in two dimensions. This leads us to the planar models shown in Figure 5.8. We do not explicitly model the weight of the grasped object. Instead, we require that the normal force on each manipulator be larger than some critical value so that the resulting frictional force in the vertical direction equilibrates the gravitational force.

- The control bandwidth for the platforms is much lower than that for the manipulators. It is meaningful to plan the complete motion for the platform, while the manipulators are controlled to accommodate errors in platform trajectories. Since the manipulator actuator forces will be determined primarily by errors that occur during execution and by feedback control laws, we only determine the set points for desired contact forces and assume that a suitable impedance controller will simultaneously regulate contact forces and the object trajectory.

- Because we are interested in determining the manipulator contact forces (and not the actuator forces), we assume a simplified model for the manipulator as shown in Figure 5.8(a). Each manipulator has two prismatic (actuator forces, N_i and T_i) and one rotary actuator (actuator torque, τ_i). On each manipulator, the actuator forces corresponding to the two linear actuators are the normal and tangential contact forces. All the manipulator mass is lumped at the object and at the base.

- Finally, in order to simplify the control problem, we generate motion plans in which the position of the center of the grasped object is midway between the centers of the bases of the two manipulators. With such plans, even if the manipulators are spatial, the manipulation task is confined to a vertical plane and this makes the impedance control task simpler. This requirement leads us to the model shown in Figure 5.8(b), where $x_c = \frac{x_1+x_2}{2}$ and $y_c = \frac{y_1+y_2}{2}$.

- We assume that the actuators for the platform consist of coaxial motors on drive wheels along an axis that passes through the geometric center of the platform as shown in Figure 5.8(b). Further, the manipulators are mounted centrally on each platform.

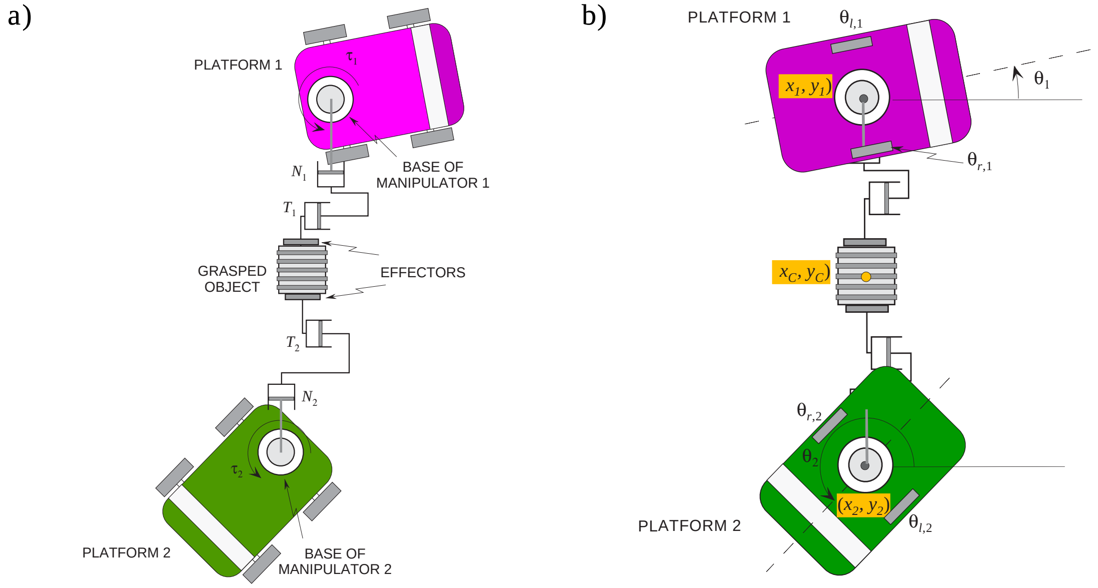

Figure 5.8: A planar model for cooperating mobile manipulators showing (a) the simplified model for the manipulator, and (b) the generalized coordinates.

5.3.2 Dynamics

The dynamic equations for the closed chain formed by the two agents, the object and the ground, can be derived using Lagrange's equation. Let q denote the 10×1 vector of generalized coordinates (see Figure 5.8(b)):

$$q = [x_1, y_1, \theta_1, \theta_{r,1}, \theta_{l,1}, x_2, y_2, \theta_2, \theta_{r,2}, \theta_{l,2}]^T \tag{5.4}$$

The Lagrangian, $L(q, \dot{q})$, is given by:

$$\frac{1}{2}\sum_{i=1}^{2}\left[m_{p,i}(\dot{x_i}^2 + \dot{y_i}^2) + I_{p,i}(\dot{\theta}_i^2) + I_{w,i}(\dot{\theta}_{r,i}^2) + I_{w,i}(\dot{\theta}_{l,i}^2)\right] + \frac{1}{2}M\left[\left\{\frac{\dot{x}_1 + \dot{x}_2}{2}\right\}^2 + \left\{\frac{\dot{y}_1 + \dot{y}_2}{2}\right\}^2\right]$$

where $\theta_{r,i}$ and $\theta_{l,i}$ are the wheel rotations, $I_{p,i}$ is the moment of inertia of the platform about a vertical axis through the centroid, $I_{w,i}$ is the wheel inertia, $m_{p,i}$ is the mass of platform and M the mass of the object being manipulated. The subscript, i denotes the i^{th} platform. Note that $I_{p,i}$ and $m_{p,i}$ (and if necessary M) include the lumped inertia of the manipulator.

The equations of motion are given by Lagrange's equation:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\right) - \frac{\partial L}{\partial q} = Q + A^T\sigma \tag{5.5}$$

where, Q is a 10×1 vector of generalized forces acting on the platform, σ is a 6×1 vector of Lagrange multipliers and A is the 6×10 matrix of coefficients representing the kinematic constraints which are given by:

$$\begin{array}{rcl} \dot{x}_1 \sin\theta_1 - \dot{y}_1 \cos\theta_1 & = & 0 \\ \dot{x}_1 \cos\theta_1 + \dot{y}_1 \sin\theta_1 - \dfrac{R}{2}(\dot{\theta}_{r,1} + \dot{\theta}_{l,1}) & = & 0 \\ \dot{\theta}_1 - \dfrac{R}{2B}(\dot{\theta}_{r,1} - \dot{\theta}_{l,1}) & = & 0 \\ \dot{x}_2 \sin\theta_2 - \dot{y}_2 \cos\theta_2 & = & 0 \\ \dot{x}_2 \cos\theta_2 + \dot{y}_2 \sin\theta_2 - \dfrac{R}{2}(\dot{\theta}_{r,2} + \dot{\theta}_{l,2}) & = & 0 \\ \dot{\theta}_2 - \dfrac{R}{2B}(\dot{\theta}_{r,2} - \dot{\theta}_{l,2}) & = & 0 \end{array} \tag{5.6}$$

where, R, is the wheel radius. These equations can be compactly written as:

$$A(q)\dot{q} = 0 \tag{5.7}$$

where, A can be further simplified to consist of the following form:

$$A = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \tag{5.8}$$

Based on Equation (5.7), we can write

$$A_i = \begin{bmatrix} \sin\theta_i & -\cos\theta_i & 0 & 0 & 0 \\ \cos\theta_i & \sin\theta_i & 0 & -\frac{R}{2} & -\frac{R}{2} \\ 0 & 0 & 1 & -\frac{R}{2B} & \frac{R}{2B} \end{bmatrix} \qquad i = 1, 2 \tag{5.9}$$

After some manipulation we obtain the dynamic equations for the system in the form [31]:

$$I\ddot{q} = Q + A^T\sigma \tag{5.10}$$

where, the inertia matrix, I, is given by:

$$I = \begin{bmatrix} \bar{M}_1 & 0 & 0 & 0 & 0 & M/4 & 0 & 0 & 0 & 0 \\ 0 & \bar{M}_1 & 0 & 0 & 0 & 0 & M/4 & 0 & 0 & 0 \\ 0 & 0 & I_{p,1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{w,1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & I_{w,1} & 0 & 0 & 0 & 0 & 0 \\ M/4 & 0 & 0 & 0 & 0 & \bar{M}_2 & 0 & 0 & 0 & 0 \\ 0 & M/4 & 0 & 0 & 0 & 0 & \bar{M}_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & I_{p,2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & I_{w,2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & I_{w,2} \end{bmatrix}$$

where $\bar{M}_1 = \frac{M}{4} + m_{p,1}$, $\bar{M}_2 = \frac{M}{4} + m_{p,2}$ and

$$Q \quad = \quad [N_{1,x} + T_{1,x}, N_{1,y} + T_{1,y}, \tau_1, \tau_{r,1}, \tau_{l,1}, N_{2,x} + T_{2,x}, N_{2,y} + T_{2,y}, \tau_2, \tau_{r,2}, \tau_{l,2}]^T$$

$\tau_{r,i}$ and $\tau_{l,i}$ are the actuator torques for platform, i. The normal force, N_i, and the tangential force, T_i, are the two manipulator actuator forces, with x and y components denoted by appropriate subscripts. The joint torque τ_i is the torque at the base joint of the manipulator.

We can eliminate the six Lagrange multipliers from Equations (5.7) and (5.10) to get four equations which represent dynamic equality constraints for our system. They can be written as:

$$\begin{aligned}
&\left[\bar{M}_1\ddot{x}_1 + \frac{M}{4}\ddot{x}_2\right]\cos\theta_1 + \left[\bar{M}_1\ddot{y}_1 + \frac{M}{4}\ddot{y}_2\right]\sin\theta_1 = Q_{1,x}\cos\theta_1 + Q_{1,y}\sin\theta_1 + \sigma_2 \\
&\left[\frac{M}{4}\ddot{x}_1 + \bar{M}_2\ddot{x}_2\right]\cos\theta_2 + \left[\frac{M}{4}\ddot{y}_1 + \bar{M}_2\ddot{y}_2\right]\sin\theta_2 = Q_{2,x}\cos\theta_2 + Q_{2,y}\sin\theta_2 + \sigma_5 \\
&I_1\ddot{\theta}_1 = \tau_1 + \frac{B[(\tau_{r,1} - \tau_{l,1}) - I_{w,1}(\ddot{\theta}_{r,1} - \ddot{\theta}_{l,1})]}{R} \\
&I_2\ddot{\theta}_2 = \tau_2 + \frac{B[(\tau_{r,2} - \tau_{l,2}) - I_{w,2}(\ddot{\theta}_{r,2} - \ddot{\theta}_{l,2})]}{R} \qquad (5.11)
\end{aligned}$$

where $Q_{i,x} = N_{i,x} + T_{i,x}$, $Q_{i,y} = N_{i,y} + T_{i,y}$ $(i = 1, 2)$ and

$$\begin{aligned}
\sigma_2 &= \frac{[(\tau_{r,1} + \tau_{l,1}) - I_{w,1}(\ddot{\theta}_{r,1} + \ddot{\theta}_{l,1})]}{R} \\
\sigma_5 &= \frac{[(\tau_{r,2} + \tau_{l,2}) - I_{w,2}(\ddot{\theta}_{r,2} + \ddot{\theta}_{l,2})]}{R}
\end{aligned}$$

There are three types of inequality constraints on the system:

1. The two manipulators have limited workspaces. Thus the platforms cannot drift too far apart. In order to model this, we impose limits on the prismatic joints along the line joining the bases of the two manipulators. If the length of the $R{-}P{-}P{-}P{-}P{-}R$ chain between the two platforms (see Figure 5.8(b)) is l, there is a lower limit and an upper limit on l:

$$l_l \;\le\; l = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \;\le\; l_u \qquad (5.12)$$

   The maneuverability of a platform relative to the other is not restricted unless l is at a limit.

2. We artificially limit the turning radius of the platform. Although in our experimental platforms, they have no such limit, this constraint makes our plans conservative. We

choose this so that the motion plans can be used with car-like systems that do have constraints on the turning radius.

$$\rho_i^2 \dot{\theta}_i^2 - \dot{x}_i^2 - \dot{y}_i^2 \leq 0 \qquad i = 1, 2. \tag{5.13}$$

Here, ρ_i is the minimum radius of curvature for the path of the geometric center of platform, i.

3. In order for the object to be held in a stable grasp, the normal force, N_i must be non negative and the tangential force, T_i, must be bounded. We adopt the Coulomb friction model and the constraint equations are given by:

$$N_1 \geq \delta > 0, \quad N_2 \geq \delta > 0, \quad |T_1| \leq \mu N_1, \quad |T_2| \leq \mu N_2 \tag{5.14}$$

Here δ is an appropriately chosen constant that will guarantee the ability to equilibrate gravitational forces on the object and μ is the coefficient of friction.

5.4 Kinematic motion plans

In this section, we will demonstrate the planning of shortest distance paths in the presence of constraints. This will involve the determination of geodesics on the configuration space (manifold). If the configuration space is R^n (but with no obstacles), the geodesics are simply straight lines in R^n. However, this is not true in general. Even in R^n, the presence of obstacles can significantly alter the trajectory.

Note that in this section, we will only be interested in planning optimal trajectories with piecewise continuous velocities. This assumes that the system has a suitable velocity controller that can generate such trajectories. If smoother trajectories are desired, the dynamics of the system need to be incorporated and this is the subject of the next section.

5.4.1 The variational problem

We formulate the shortest distance problem by synthesizing a cost function that is the integral of the norm of the actuator velocities. This cost function is given by:

$$J = \int_{t_o}^{t_f} \left(\dot{\theta}_{r,1}^2 + \dot{\theta}_{l,1}^2 + \dot{\theta}_{r,2}^2 + \dot{\theta}_{l,2}^2 \right) dt \tag{5.15}$$

It is easily seen that the integrand is sum of the squares of the Cartesian velocities of the two platforms with appropriate geometric scaling factors. $u = [\dot{\theta}_{r,1}, \dot{\theta}_{l,1}, \dot{\theta}_{r,2}, \dot{\theta}_{l,2}]^T$ is the vector of control inputs.

There are three types of constraints that must be incorporated. First, the distance between the two platforms is constrained as shown in Equation (5.12). Second, the turning radius for each platform may be limited as discussed in Section 5.3 (see Equation (5.13)). Finally, we have additional constraints imposed by obstacles in the environment. These constraints are of the form:

$$\epsilon - d_{ij} \quad \leq 0 \qquad i = 1, 2, \ldots, n_l; \;\; j = 1, 2, \ldots n_o \tag{5.16}$$

where, ϵ denotes the required clearance between the robot and the obstacle, d_{ij} denotes the distance of the manipulator link, i from the obstacle, j, n_l is the number of circles used to model the manipulator links and n_o is the number of convex polygonal obstacles. Thus there are $n_l \times n_o$ constraints. However, at a given configuration, it is possible to deactivate certain constraints due to obstacles that are far away using appropriate thresholding.

Example 1: Consider the example shown in Figure 5.9 in which two mobile manipulators maneuver through a constriction created by two obstacles. We will show how to find kinematic motion plans for this task. The treatment in this example is kept as simple as possible. We will not follow the procedure of differentiating state constraints described in Section 3.4.1. Since the state constraints will only be of first order constraints ($p = 1$), satisfactory performance can be obtained by directly adjoining the constraints in the Hamiltonian. Also, we will not try to reduce the dimensionality of the problem by substituting for the control inputs in terms of the state derivatives as discussed in Section 3.4.1.

There are two obstacles in the problem, one of which is concave. This obstacle is decomposed into two convex obstacles (see Figure 5.9). Each mobile platform is modeled by a circle as shown in Figure 5.6. The grasped object is above the plane of the platform motion and is not modeled for obstacle avoidance. The parameters describing the manipulators are shown in Table 5.1 (Note that the inertial parameters are not relevant for this example as we are only interested in a kinematic motion plan).

The kinematic model of cooperating mobile platforms is given by the following set of equations:

$$\begin{aligned}
\dot{x}_1 &= \tfrac{R}{2}(u_1 + u_2)\cos\theta_1 & \dot{x}_2 &= \tfrac{R}{2}(u_3 + u_4)\cos\theta_2 \\
\dot{y}_1 &= \tfrac{R}{2}(u_1 + u_2)\sin\theta_1 & \dot{y}_2 &= \tfrac{R}{2}(u_3 + u_4)\sin\theta_2 \\
\dot{\theta}_1 &= \tfrac{R}{2B}(u_1 - u_2)\cos\theta_1 & \dot{\theta}_2 &= \tfrac{R}{2B}(u_3 - u_4)\cos\theta_2 \\
\dot{\theta}_{r,1} &= u_1 & \dot{\theta}_{r,2} &= u_3 \\
\dot{\theta}_{l,1} &= u_2 & \dot{\theta}_{l,2} &= u_4
\end{aligned} \tag{5.17}$$

We define the state vector, x, consists of the generalized coordinates:

$$x = [x_1, y_1, \theta_1, \theta_{r,1}, \theta_{l,1}, x_2, y_2, \theta_2, \theta_{r,2}, \theta_{l,2}]^T$$

The inequalities can be converted to equalities and written in the following form:

$$\begin{array}{ll}
\hat{h}_1 = \rho_1^2 \left[\frac{R}{2B}(\dot{\theta}_{r,1} - \dot{\theta}_{l,1})\right]^2 - \left[\frac{R}{2}(\dot{\theta}_{r,1} + \dot{\theta}_{l,1})\right]^2 + \xi_1^2 & \hat{h}_5 = \epsilon - d_{11} + \xi_5^2 \\
\hat{h}_2 = \rho_2^2 \left[\frac{R}{2B}(\dot{\theta}_{r,2} - \dot{\theta}_{l,2})\right]^2 - \left[\frac{R}{2}(\dot{\theta}_{r,2} + \dot{\theta}_{l,2})\right]^2 + \xi_2^2 & \hat{h}_6 = \epsilon - d_{21} + \xi_6^2 \\
\hat{h}_3 = l_l - l + \xi_3^2 & \hat{h}_7 = \epsilon - d_{12} + \xi_7^2 \\
\hat{h}_4 = l - l_u + \xi_4^2 & \hat{h}_8 = \epsilon - d_{22} + \xi_8^2
\end{array} \tag{5.18}$$

To reduce the problem to the standard form, we will follow the procedure of Section 3.4 (with $l_\phi = 0$) and define the extended state vector as:

$$X = [X_x^T, X_u^T, X_\xi^T, X_\lambda^T, X_\psi^T]^T \tag{5.19}$$

where

$$\begin{array}{ll}
X_x = [x_1, y_1, \theta_1, \theta_{r,1}, \theta_{l,1}, x_2, y_2, \theta_2, \theta_{r,2}, \theta_{l,2}]^T, & \dot{X}_u = [u_1 \;\; u_2 \;\; u_3 \;\; u_4]^T, \\
\dot{X}_\lambda = [\lambda_1 \dots \lambda_{10}]^T, & \dot{X}_\xi = [\xi_1 \dots \xi_8]^T \\
\dot{X}_\psi = [\psi_1 \dots \psi_8]^T &
\end{array}$$

$\lambda_1, \cdots, \lambda_{10}$ are multipliers for the state equations given in Equation (5.17), $\psi_1, \cdots, \psi_8$ are multipliers for the modified inequality constraints and $\xi_1, \cdots, \xi_8$ are the slack variables in Equation (5.18). The Hamiltonian for the variational problem can now be expressed as a function of the $40-$dimensional extended state vector X and $\dot{X}$. The boundary conditions are:

$$X_x|_{t_0} = x(t_o), \quad X_u|_{t_0} = 0, \quad X_\lambda|_{t_0} = 0, \quad X_\xi|_{t_0} = 0, \quad X_\psi|_{t_0} = 0,$$

$$X_x|_{t_f} = x(t_f), \quad \frac{\partial H}{\partial \dot{X}_u}|_{t_f} = 0, \quad \frac{\partial H}{\partial \dot{X}_\lambda}|_{t_f} = 0, \quad \frac{\partial H}{\partial \dot{X}_\xi}|_{t_f} = 0, \quad \frac{\partial H}{\partial \dot{X}_\psi}|_{t_f} = 0$$

where

$$\frac{\partial H}{\partial \dot{X}_u}|_{t_f} = \begin{bmatrix} -\frac{R}{2}\lambda_1\cos\theta_1 - \frac{R}{2}\lambda_2\sin\theta_1 - \frac{R}{2B}\lambda_3\cos\theta_1 - \lambda_4 + 2u_1 \\ -\frac{R}{2}\lambda_1\cos\theta_1 - \frac{R}{2}\lambda_2\sin\theta_1 + \frac{R}{2B}\lambda_3\cos\theta_1 - \lambda_5 + 2u_2 \\ -\frac{R}{2}\lambda_6\cos\theta_2 - \frac{R}{2}\lambda_7\sin\theta_2 - \frac{R}{2B}\lambda_8\cos\theta_2 - \lambda_9 + 2u_3 \\ -\frac{R}{2}\lambda_6\cos\theta_2 - \frac{R}{2}\lambda_7\sin\theta_2 + \frac{R}{2B}\lambda_8\cos\theta_2 - \lambda_{10} + 2u_4 \end{bmatrix}, \quad \frac{\partial H}{\partial \dot{X}_\xi}|_{t_f} = \begin{bmatrix} \psi_1\xi_1 \\ \psi_2\xi_2 \\ \psi_3\xi_3 \\ \psi_4\xi_4 \\ \psi_5\xi_5 \\ \psi_6\xi_6 \\ \psi_7\xi_7 \\ \psi_8\xi_8 \end{bmatrix}$$

$$\frac{\partial H}{\partial \dot{X}_\lambda}|_{t_f} = \begin{bmatrix} \dot{x}_1 - \frac{R}{2}(u_1+u_2)\cos\theta_1 \\ \dot{y}_1 - \frac{R}{2}(u_1+u_2)\sin\theta_1 \\ \dot{\theta}_1 - \frac{R}{2B}(u_1-u_2)\cos\theta_1 \\ \dot{\theta}_{r,1} - u_1 \\ \dot{\theta}_{l,1} - u_2 \\ \dot{x}_2 - \frac{R}{2}(u_3+u_4)\cos\theta_2 \\ \dot{y}_2 - \frac{R}{2}(u_3+u_4)\sin\theta_2 \\ \dot{\theta}_2 - \frac{R}{2B}(u_3-u_4)\cos\theta_1 \\ \dot{\theta}_{r,2} - u_3 \\ \dot{\theta}_{l,2} - u_4 \end{bmatrix}, \quad \frac{\partial H}{\partial \dot{X}_\psi}|_{t_f} = \begin{bmatrix} \rho^2_{min1}\dot{\theta}^2_1 - \dot{x}^2_1 - \dot{y}^2_1 + \xi^2_1 \\ \rho^2_{min2}\dot{\theta}^2_2 - \dot{x}^2_2 - \dot{y}^2_2 + \xi^2_2 \\ l_l - l + \xi^2_3 \\ l - l_u + \xi^2_4 \\ \epsilon - d_{11} + \xi^2_5 \\ \epsilon - d_{21} + \xi^2_6 \\ \epsilon - d_{12} + \xi^2_7 \\ \epsilon - d_{22} + \xi^2_8 \end{bmatrix}$$

Table 5.1: The parameters used in Examples $1-4$.

N	ρ_1	ρ_2	R	B	l_l	l_u	ϵ
100	0.1 m	0.1 m	0.1 m	0.3 m	0.9 m	1.1 m	0.01 m
k_0	k_1	μ	$m_{p,1}$	$m_{p,2}$	$I_{p,1}$	$I_{p,2}$	
5	5	0.25	5.0 Kg	5.0 Kg	1.0 Kg-m^2	1.0 Kg-m^2	

Figure 5.9 shows snapshots of the computed motion plan at equal intervals in time (0.25 seconds). The path of the geometric center of each platform is shown in Figure 5.10. The linear velocity of the geometric center, v_i, and the angular velocity, ω_i, of platform i

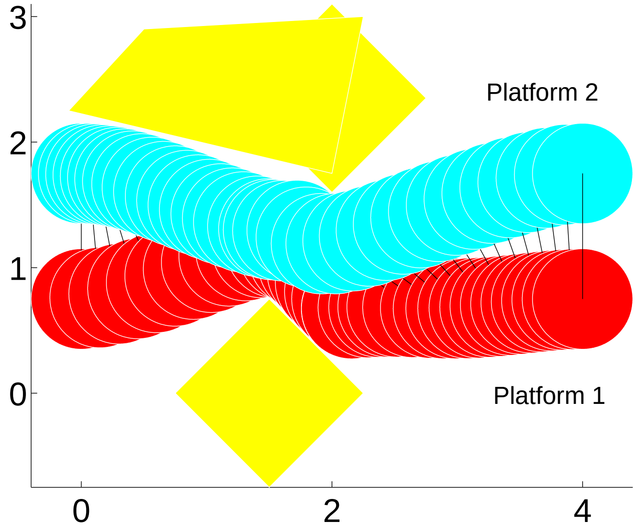

Figure 5.9: Example 1: Two robots going through a narrow constriction.

is shown in Figure 5.11. The shortest path requires the system of mobile manipulators to reconfigure from a “march-abreast” formation to a “follow-the-leader” formation and then back to the first formation in order to squeeze through the constriction in the middle. It is seen that the platforms tend to “hug” the obstacle boundary as they avoid it. Since the system switches from an unconstrained state to a constrained state, we can expect discontinuities in the velocity profile for a kinematic modeling of the system. These are clearly evident in the plots shown in Figure 5.11. Due to the limitations of the TRC-Labmates internal controllers and their narrow bandwidth, it is impractical to use such discontinuous velocity profiles for accurate trajectory following. Errors due to dead-reckoning are unavoidable and hence in order to get satisfactory response from the experimental system, it is meaningful to have “smooth” velocity profiles. Thus we seek minimization of energy or the actuator forces to obtain a continuous velocity profile in the next section.

5.5 Motion planning with dynamics

In this section, we will illustrate with examples the procedure for generating plans that conform to dynamic constraints such as frictional constraints, normal forces and so on. By choosing a cost function that will depend on torques, we will guarantee solution trajectories in which the velocities are piecewise smooth. Correspondingly in the process, we have a satisfactory solution for our experimental system. The actuator forces, and therefore

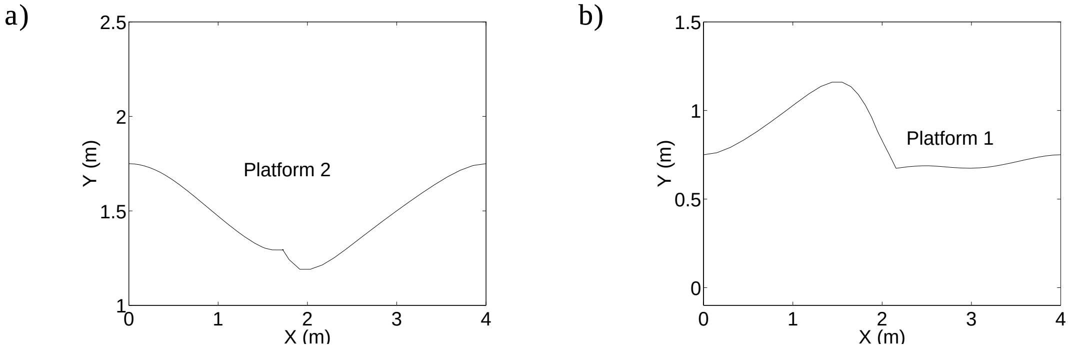

Figure 5.10: Example 1: Paths of (a) Platform 2; and (b) Platform 1.

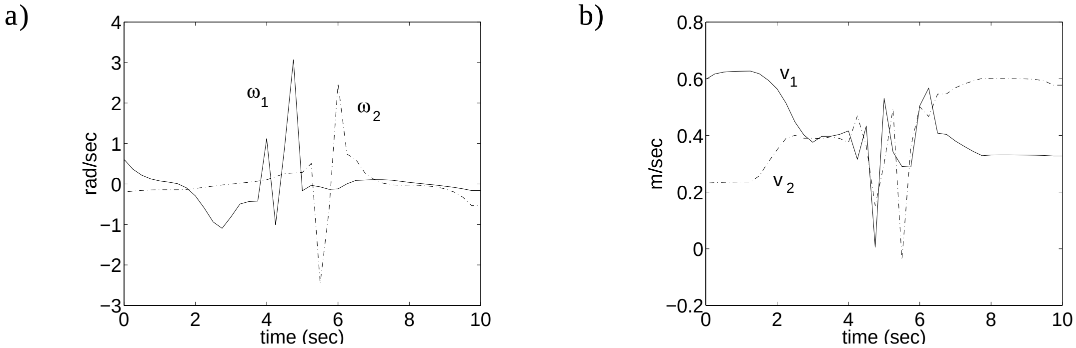

Figure 5.11: Example 1: Velocities of the platforms.

accelerations, may only be piecewise continuous.

5.5.1 The variational problem

In this section, we will consider the minimization of the integral of the norm of the actuator forces and torques for the platforms and the manipulators according to the model shown in Figure 5.8. The cost function is given by:

$$\int_{t_0}^{t_f} \|u\|_W^2 dt \tag{5.20}$$

where the input vector is:

$$u = [\tau_1, \tau_{r,1}, \tau_{l,1}, N_1, T_1, \tau_2, \tau_{r,2}, \tau_{l,2}, N_2, T_2]^T$$

and $\|u\|_W$ is the weighted norm $u^T W u$, with W a user-defined positive-definite weighting matrix.

The state vector, x, consists of the position and velocities of the platforms:

$$x = [P \;\; V]^T$$

where

$$P = [x_1, y_1, \theta_1, \theta_{r,1}, \theta_{l,1}, x_2, y_2, \theta_2, \theta_{r,2}, \theta_{l,2}]^T$$

and

$$V = [\dot{x}_1, \dot{y}_1, \dot{\theta}_1, \dot{\theta}_{r,1}, \dot{\theta}_{l,1}, \dot{x}_2, \dot{y}_2, \dot{\theta}_2, \dot{\theta}_{r,2}, \dot{\theta}_{l,2}]^T$$

In this problem, in addition to the inequality constraints discussed in Section 5.4.1, there is an additional set of inequalities associated with the grasping forces. The normal and tangential forces must satisfy Coulomb's law according to Equation (5.14). All inequalities can be written according to the procedure in Section 3.4 as equalities:

$$\begin{aligned} &\hat{h}_1 = l_l - l + \xi_1^2 && \hat{h}_2 = l - l_u + \xi_2^2 && \\ &\hat{h}_3 = \rho_1^2\dot{\theta}_1^2 - \dot{x}_1^2 - \dot{y}_1^2 + \xi_3^2 && \hat{h}_4 = \rho_2^2\dot{\theta}_2^2 - \dot{x}_2^2 - \dot{y}_2^2 + \xi_4^2 && \\ &\hat{h}_5 = \delta - N_1 + \xi_5^2 && \hat{h}_6 = T_1 - \mu N_1 + \xi_6^2 && \hat{h}_7 = -T_1 - \mu N_1 + \xi_7^2 \\ &\hat{h}_8 = \delta - N_2 + \xi_8^2 && \hat{h}_9 = T_2 - \mu N_2 + \xi_9^2 && \hat{h}_{10} = -T_2 - \mu N_2 + \xi_{10}^2 \end{aligned} \tag{5.21}$$

In addition, we have the six kinematic equality constraints on the state:

$$\begin{aligned} &\dot{x}_1 \sin\theta_1 - \dot{y}_1 \cos\theta_1 = 0, && \dot{x}_1 \cos\theta_1 + \dot{y}_1 \sin\theta_1 - \tfrac{R}{2}(\dot{\theta}_{r,1} + \dot{\theta}_{l,1}) = 0, \\ &\dot{\theta}_1 - \tfrac{R}{2B}(\dot{\theta}_{r,1} - \dot{\theta}_{l,1}) = 0, && \dot{x}_2 \sin\theta_2 - \dot{y}_2 \cos\theta_2 = 0, \\ &\dot{x}_2 \cos\theta_2 + \dot{y}_2 \sin\theta_2 - \tfrac{R}{2}(\dot{\theta}_{r,2} + \dot{\theta}_{l,2}) = 0, && \dot{\theta}_2 - \tfrac{R}{2B}(\dot{\theta}_{r,2} - \dot{\theta}_{l,2}) = 0 \end{aligned} \tag{5.22}$$

Of the equality constraints in Equation (5.21) and Equation (5.22), ten are state constraints that need to be differentiated to make explicit the dependence on the input. Thus, after differentiating the appropriate constraints in Equation (5.21) and Equation (5.22), we obtain:

$$\begin{aligned} -\ddot{l} + 2\dot{\xi}_1^2 + 2\xi_1\ddot{\xi}_1 &= 0 & \ddot{l} + 2\dot{\xi}_2^2 + 2\xi_2\ddot{\xi}_2 &= 0 \\ \rho_1^2\dot{\theta}_1\ddot{\theta}_1 - \dot{x}_1\ddot{x}_1 - \dot{y}_1\ddot{y}_1 + \xi_3\dot{\xi}_3 &= 0 & \rho_2^2\dot{\theta}_2\ddot{\theta}_2 - \dot{x}_2\ddot{x}_2 - \dot{y}_2\ddot{y}_2 + \xi_4\dot{\xi}_4 &= 0 \\ \delta - N_1 + \xi_5^2 &= 0 & T_1 - \mu N_1 + \xi_6^2 &= 0 \\ -T_1 - \mu N_1 + \xi_7^2 &= 0 & \delta - N_2 + \xi_8^2 &= 0 \\ T_2 - \mu N_2 + \xi_9^2 &= 0 & -T_2 - \mu N_2 + \xi_{10}^2 &= 0 \end{aligned} \tag{5.23}$$

and

$$\begin{aligned}
&\ddot{x}_1 \sin\theta_1 + \dot{x}_1\dot{\theta}_1 \cos\theta_1 - \ddot{y}_1 \cos\theta_1 + \dot{y}_1\dot{\theta}_1 \sin\theta_1 = 0 \\
&\ddot{x}_1 \cos\theta_1 - \dot{x}_1\dot{\theta}_1 \sin\theta_1 + \ddot{y}_1 \sin\theta_1 + \dot{y}_1\dot{\theta}_1 \cos\theta_1 - \tfrac{R}{2}(\ddot{\theta}_{r,1} + \ddot{\theta}_{l,1}) = 0 \\
&\ddot{\theta}_1 - \tfrac{R}{2B}(\ddot{\theta}_{r,1} - \ddot{\theta}_{l,1}) = 0 \\
&\ddot{x}_2 \sin\theta_2 + \dot{x}_2\dot{\theta}_2 \cos\theta_2 - \ddot{y}_2 \cos\theta_2 + \dot{y}_2\dot{\theta}_2 \sin\theta_2 = 0 \\
&\ddot{x}_2 \cos\theta_2 - \dot{x}_2\dot{\theta}_2 \sin\theta_2 + \ddot{y}_2 \sin\theta_2 + \dot{y}_2\dot{\theta}_2 \cos\theta_2 - \tfrac{R}{2}(\ddot{\theta}_{r,2} + \ddot{\theta}_{l,2}) = 0 \\
&\ddot{\theta}_2 - \tfrac{R}{2B}(\ddot{\theta}_{r,2} - \ddot{\theta}_{l,2}) = 0
\end{aligned} \tag{5.24}$$

respectively. Based on Equation (5.11) and Equation (5.24) we can combine the two sets of equations to get a system of equations of the form:

$$[f_1(P)]\ \dot{V} = [f_2(P,V)] + [f_3(P)]u \tag{5.25}$$

Note that we can invert the 10×10 matrix $[f_1(P)]$ to obtain the expression for $\dot{V}$. Thus, the state equations are given by:

$$\begin{bmatrix} \dot{P} \\ \dot{V} \end{bmatrix} = \begin{bmatrix} V \\ [f_1(P)]^{-1}\left\{[f_2(P,V)] + [f_3(P)]u\right\} \end{bmatrix} \tag{5.26}$$

In this way, we embed the 6 equality constraints into the state Equation (3.1).

The extended state vector, X, is defined by:

$$X = [X_x^T, X_\eta^T, X_u^T, X_\xi^T, X_\lambda^T, X_\psi^T, X_\zeta^T]^T$$

where

$$\begin{aligned}
X_x &= [x_1, y_1, \theta_1, \theta_{r,1}, \theta_{l,1}, x_2, y_2, \theta_2, \theta_{r,2}, \theta_{l,2}, \dot{x}_1, \dot{y}_1, \dot{\theta}_1, \dot{\theta}_{r,1}, \dot{\theta}_{l,1}, \dot{x}_2, \dot{y}_2, \dot{\theta}_2, \dot{\theta}_{r,2}, \dot{\theta}_{l,2}]^T, \\
X_\eta &= [\eta_{1,1}\ \eta_{2,1}]^T, \qquad \dot{X}_u = [\tau_1, \tau_{r,1}, \tau_{l,1}, N_1, T_1, \tau_2, \tau_{r,2}, \tau_{l,2}, N_2, T_2]^T, \\
\dot{X}_\lambda &= [\lambda_1, \ldots, \lambda_{20}]^T, \qquad \dot{X}_\xi = [\ddot{\xi}_1, \ddot{\xi}_2, \dot{\xi}_3, \dot{\xi}_4, \xi_5, \xi_6, \xi_7, \xi_8, \xi_9, \xi_{10}]^T, \\
\dot{X}_\psi &= [\psi_1, \ldots, \psi_{10}]^T, \qquad \dot{X}_\zeta = [\zeta_1, \zeta_2]^T
\end{aligned} \tag{5.27}$$

$\lambda_1, \cdots, \lambda_{20}$ are multipliers for the state equations for x, and ψ are the multipliers for the 10 modified inequality constraints in Equation (5.23). ζ_1, ζ_2 are the multipliers for the equation:

$$\begin{bmatrix} \dot{\eta}_{1,1} \\ \dot{\eta}_{2,1} \end{bmatrix} = \begin{bmatrix} \eta_{1,2} \\ \eta_{2,2} \end{bmatrix} \tag{5.28}$$

where

$$\eta_{1,1} = \xi_1, \quad \eta_{2,1} = \xi_2$$

Finally, instead of directly adjoining the differentiated state constraints, we adjoin a linear combination of the state constraint and its derivatives following the procedure of Section 3.4.2. For example, the constraint on the lower bound of the distance between the platforms is adjoined as:

$$\psi_1\{(-\ddot{l} + 2\eta_{1,2}^2 + 2\eta_{1,1}\ddot{\xi}_1) + k_1(-\dot{l} + 2\eta_{1,1}\eta_{1,2}) + k_0(l_l - l + \eta_{1,1}^2)\}$$

with the boundary conditions:

$$l_l - l(0) + \eta_{1,1}^2(0) = 0, \quad -\dot{l}(0) + 2\eta_{1,1}(0)\eta_{1,2}(0) = 0.$$

Based on the nomenclature in Section 3.4.1, we have

$$\begin{aligned} &n = 20, \quad m = 10, \quad l_\zeta = 2, \quad \varrho = 4, \quad l_\xi = 10, \quad l_\phi = 0, \\ &q_1 = 2, \quad q_2 = 2, \quad q_3 = 1, \quad q_4 = 1, \quad q_5 = \cdots = q_{10} = 0 \end{aligned}$$

According to Equation (3.33), we have $r = 78$. However, we may express u in terms of the derivatives of x using Equation (5.11). Thus we can reduce the number of variables as described in Section 3.4.1 and we end up with a problem in which X is a 58×1 vector. The boundary conditions for the variational problem are:

$$X_x|_{t_0} = x(t_o), \quad X_\eta|_{t_0} = \eta(t_0), \quad X_u|_{t_0} = 0, \quad X_\lambda|_{t_0} = 0, \quad X_\xi|_{t_0} = 0, \quad X_\psi|_{t_0} = 0,$$

$$X_\zeta(t_0) = 0, \quad X_x|_{t_f} = x(t_f), \quad X_\eta|_{t_f} = \eta(t_f), \quad \frac{\partial H}{\partial \dot{X}_u}|_{t_f} = 0, \quad \frac{\partial H}{\partial \dot{X}_\lambda}|_{t_f} = 0, \quad \frac{\partial H}{\partial \dot{X}_\xi}|_{t_f} = 0,$$

$$\frac{\partial H}{\partial \dot{X}_\psi}|_{t_f} = 0, \quad \frac{\partial H}{\partial \dot{X}_\zeta}|_{t_f} = 0.$$

Based on the methodology of Section 5.4.1, we can similarly write expressions for $\frac{\partial H}{\partial \dot{X}_u}|_{t_f}$, $\frac{\partial H}{\partial \dot{X}_\lambda}|_{t_f}$, $\frac{\partial H}{\partial \dot{X}_\xi}|_{t_f}$, $\frac{\partial H}{\partial \dot{X}_\psi}|_{t_f}$ and $\frac{\partial H}{\partial \dot{X}_\zeta}|_{t_f}$ at the final time, t_f.

5.5.2 Examples and Experimental results

Example 2: We demonstrate the results obtained for the maneuver shown in Figure 5.12. The dynamic parameters are as shown in Table 5.1 and M, the mass of the object being manipulated is 0.005 Kg. This corresponds to an object weight of roughly 0.05 Newtons. We chose such a low weight to confirm our kinematic motion plans as lower mass of the grasped object necessarily leads to a purely kinematic behavior of the system. With a coefficient of friction of 0.25, the minimum normal force, δ, is taken to be 0.1 Newtons. This is clearly a situation where the dynamic interaction between the platforms is minimal. Example 4 will illustrate the effect of changing M and δ.

In this maneuver, the robot agents are in a "serial" follow-the-leader formation and have to make a 90° turn to avoid a possible obstacle in front of the first platform. Figure 5.13 illustrates the path of the center of each mobile platform. The first (leading) platform goes forward while turning to the left, reverses, and then goes forward again. Thus the path exhibits two cusps (Figure 5.13(b)). The second platform follows a smooth path with a reversal near the end of the motion — there is a small cusp that can be seen in Figure 5.13(a). Note that the turning radius constraint does not allow a simple "turn-in-place" maneuver for the first platform. Figure (5.14) shows the x and y components of velocities for each platform. Note that the velocities are smooth. In fact, they are piecewise smooth because they are state variables.

The normal and tangential contact forces for each end-effector are shown in Figure 5.15. Note that the normal force is always above the desired minimum, δ. Since the cost function depends on the contact forces, both N_1 and N_2 are kept small. As a result at least one of the normal forces is always at the minimum. The tangential forces must satisfy Coulomb's law of friction and this is seen in Figure 5.16(b). The manipulator torque for each manipulator base is shown in Figure 5.16(a) and the wheel torques are shown in Figure 5.17. Note that the wheel torques and the contact forces are only piecewise continuous. The sudden changes in the wheel torques for the first platform occur at the cusps of the path where the Cartesian velocities reverse.

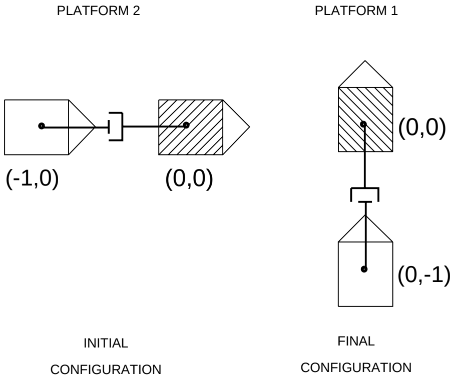

Figure 5.12: Example 2: An example of reconfiguring maneuver.

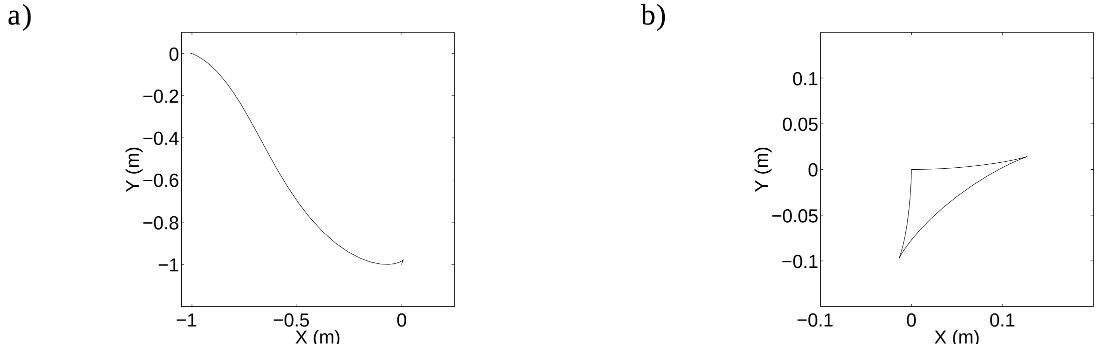

Figure 5.13: Example 2: Paths of (a) Platform 2; and (b) Platform 1.

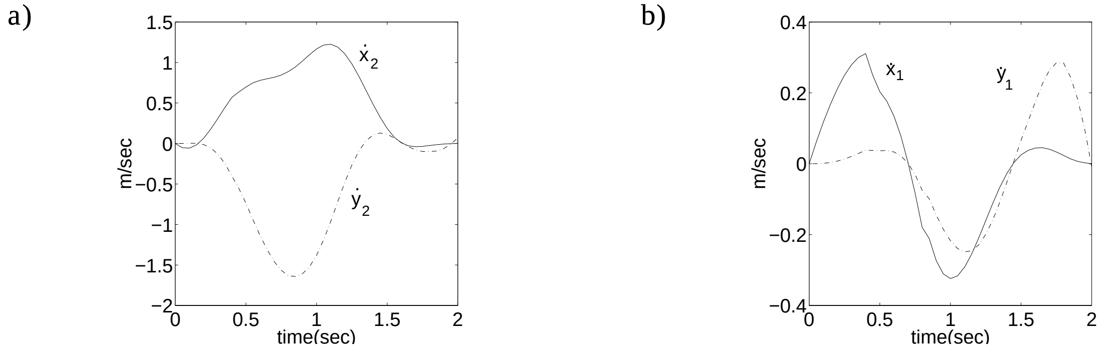

Figure 5.14: Example 2: Velocities of the platforms.

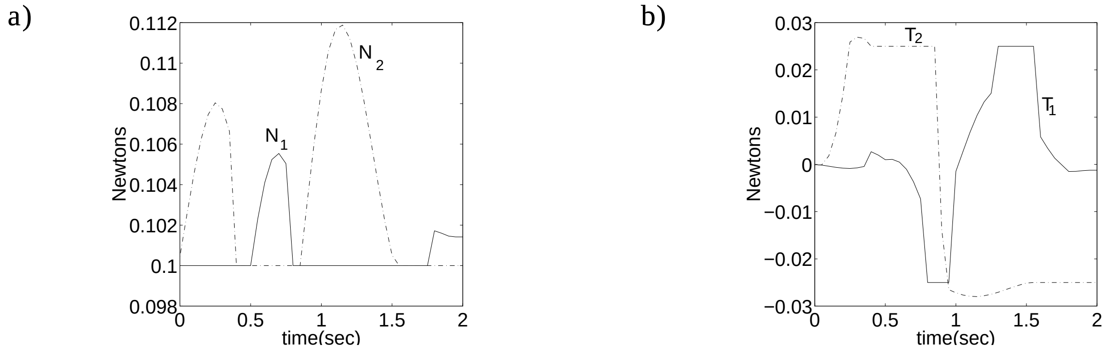

Figure 5.15: Example 2: (a) Normal force, N_i and (b) Tangential force, T_i.

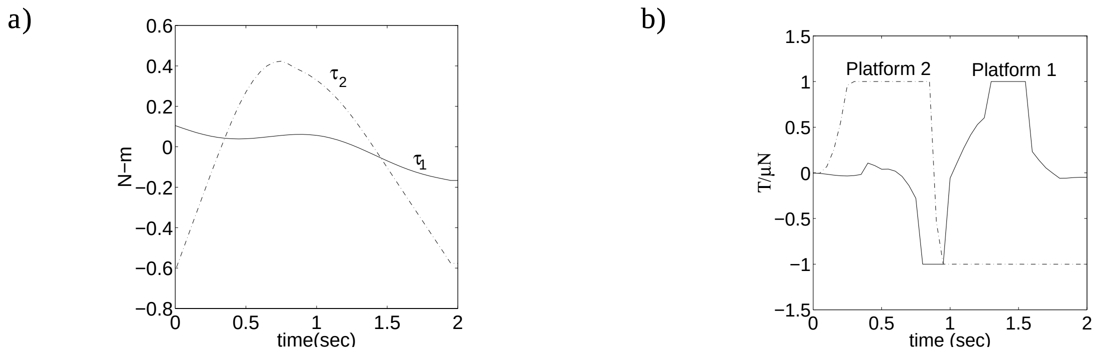

Figure 5.16: Example 2: (a) Base torques, τ_1, τ_2 and (b) $\frac{T_i}{\mu N_i}$.

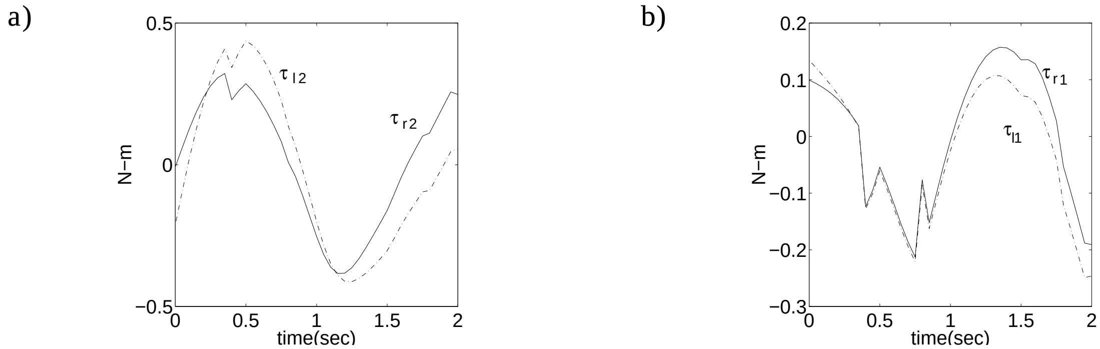

Figure 5.17: Example 2: (a) Wheel torques $\tau_{r,2}, \tau_{l,2}$ and (b) $\tau_{r,1}, \tau_{l,1}$.

Example 3: In Example 3, we treat the problem of executing a turn through 90° and a change in formation from a "serial, follow-the-leader" configuration to "parallel, march-abreast" configuration as shown in Figure 5.18. The parameters are the same as in Example 2. We also investigate the trajectories for different final conditions by varying the final orientation of the two platforms while keeping the final positions fixed. Thus, the initial configuration is the same for all the maneuvers and is given by:

$$[x_1(0), y_1(0), \theta_1(0), x_2(0), y_2(0), \theta_2(0)]^T = [0, 0, 0, -1, 0, 0]^T .$$

The desired final configuration is:

$$[x_1(0), y_1(0), \theta_1(0), x_2(0), y_2(0), \theta_2(0)]^T = [0, 0, \theta_f, 0, -1, \theta_f]^T ,$$

where the final orientation θ_f belongs to the set $\{\pi/3, \pi/2, 2\pi/3, \pi\}$.

The paths for the platforms are shown in Figure 5.19. Each platform attempts to turn at a point by moving first forward, reversing while turning and then moving forward while turning. Thus all paths show two cusps.

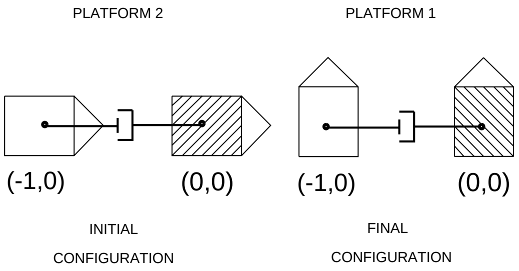

Figure 5.18: Example 3: A $90°$ turn.

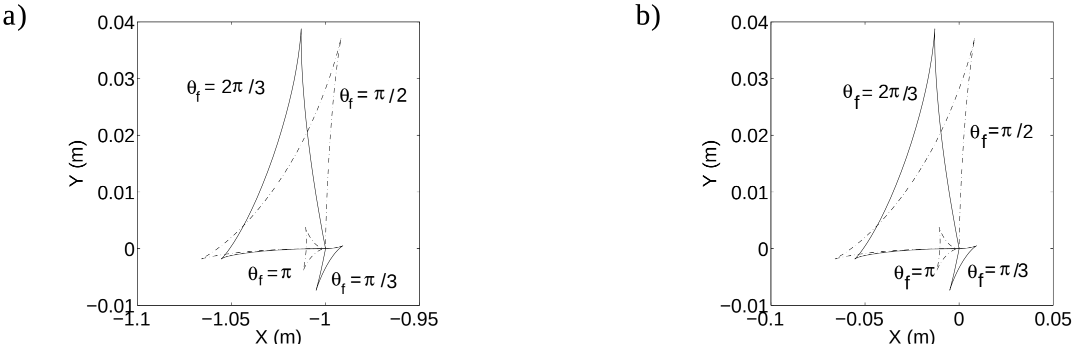

Figure 5.19: Example 3: Paths of (a) Platform 2, and (b) Platform 1 for different θ_f.

Example 4: In Example 4, we consider the same task as in Example 2 (Figure 5.12) but we vary the mass of the manipulated object. This indirectly increases the interaction forces between the mobile platforms. In Figures 5.20, 5.21, 5.22, 5.23, and 5.24, we show the planned motion for three different values of M (0.005, 1.0, and 5.0 Kg.). The corresponding thresholds (δ) on the normal forces are 0.1, 20.0, and 100.0 Newtons respectively.

As the mass and the threshold normal force increases, the dynamic interaction between the platforms becomes more significant. As shown in Figure 5.20, the trajectories show a significant change as M is changed. As the mass is increased from 0.005 to 1.0 Kg., the path of Platform 1 continues to show two cusps (two reversals) but the shape of the path between the cusps changes. With a higher mass (5.0 Kg.), the path begins to show additional cusps (Figure 5.20(c)). The trajectory of the follower (Platform 2) in this case shows only minor changes. For smaller mass values the interaction forces fluctuate although their magnitudes are small. For larger mass values the forces are steadier and larger. Figures 5.21 and 5.22 also illustrate this. For small mass values, the tangential forces approach their maximum or minimum values. Because a larger object mass requires a larger "squeeze force", the internal forces dominate over inertial forces. Finally, as seen in Figures 5.23 and 5.24, the ratio, $\frac{T_i}{\mu N_i}$, shows a steady value for larger mass values of the grasped object.

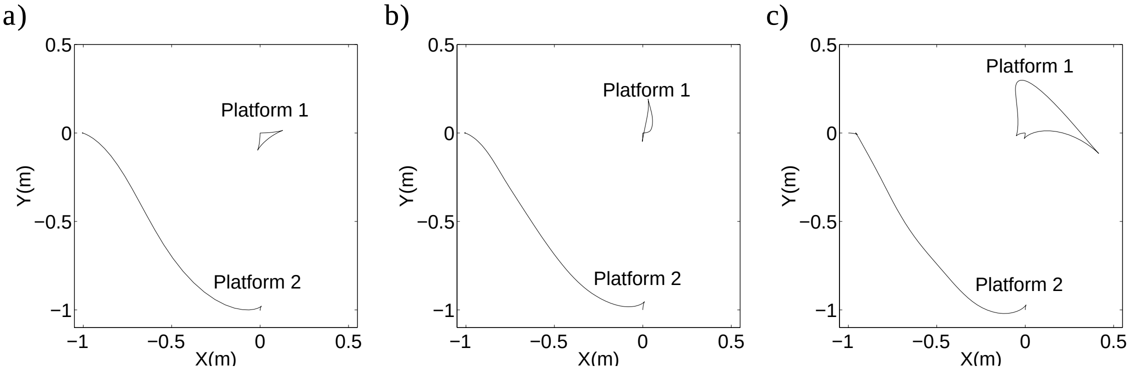

Figure 5.20: Example 4: Paths of the platforms for a) $M = 0.005$ Kg, b) $M = 1.0$ Kg and c) $M = 5.0$ Kg.

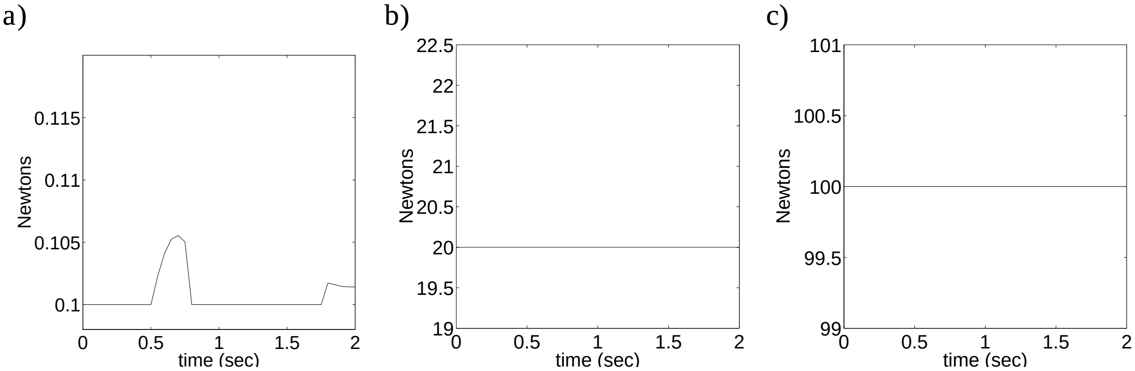

Figure 5.21: Example 4: Normal force, N_1, for a) $M = 0.005$ Kg, b) $M = 1.0$ Kg and c) $M = 5.0$ Kg.

5.6 Experimental results

In the experiments, the platforms are commanded to follow the planned trajectories while the manipulators are commanded to maintain a stable grasp. The platform control is based on position feedback that is obtained from dead-reckoning. Therefore the commanded position (and velocity) trajectories are translated into position (and velocity) trajectories for the drive wheels, and the wheels are appropriately controlled based on position feedback using proportional control. We show the experimental results in Figures 5.25 and 5.26. In all these plots the variation of the relative position and orientation (measured by the instrumented four bar linkage) is shown and compared with the relative position and orientation computed from the optimal motion plan. The results for the 90°

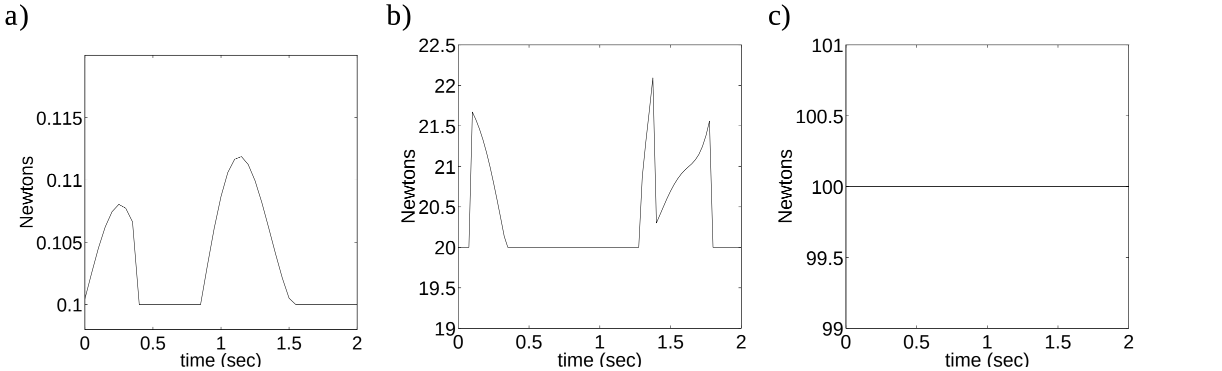

Figure 5.22: Example 4: Normal force, N_2, for a) $M = 0.005$ Kg, b) $M = 1.0$ Kg and c) $M = 5.0$ Kg.

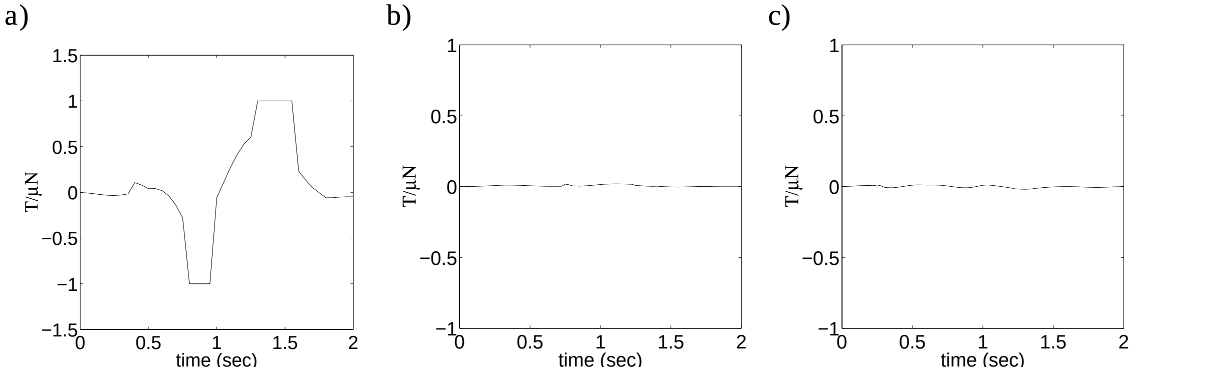

Figure 5.23: Example 4: $\frac{T_1}{\mu N_1}$ for a) $M = 0.005$ Kg, b) $M = 1.0$ Kg and c) $M = 5.0$ Kg.

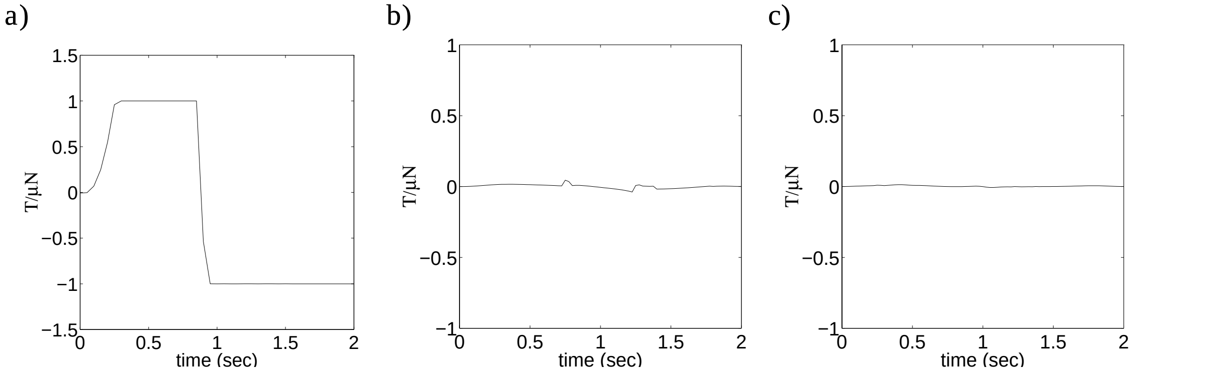

Figure 5.24: Example 4: $\frac{T_2}{\mu N_2}$ for a) $M = 0.005$ Kg, b) $M = 1.0$ Kg and c) $M = 5.0$ Kg.

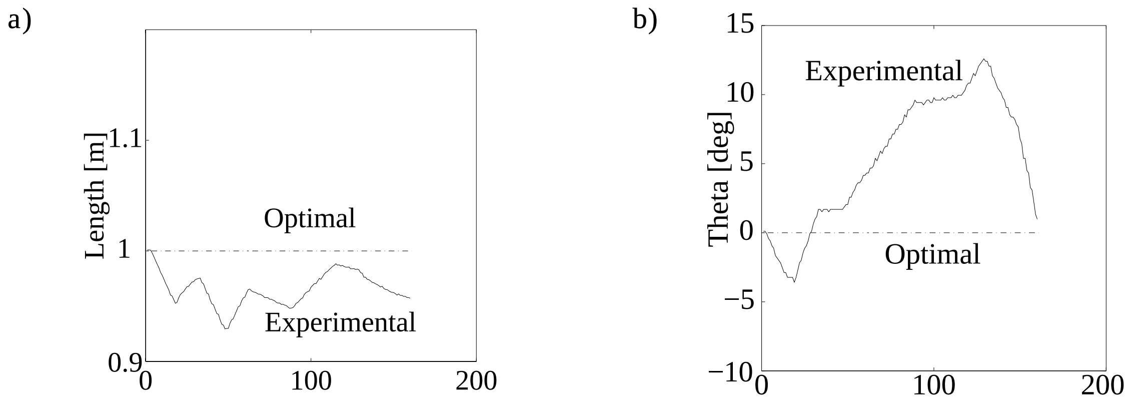

Figure 5.25: Variation of distance (a) and relative angle between platforms (b), $\theta_2 - \theta_1$, for a $90°$ turn.

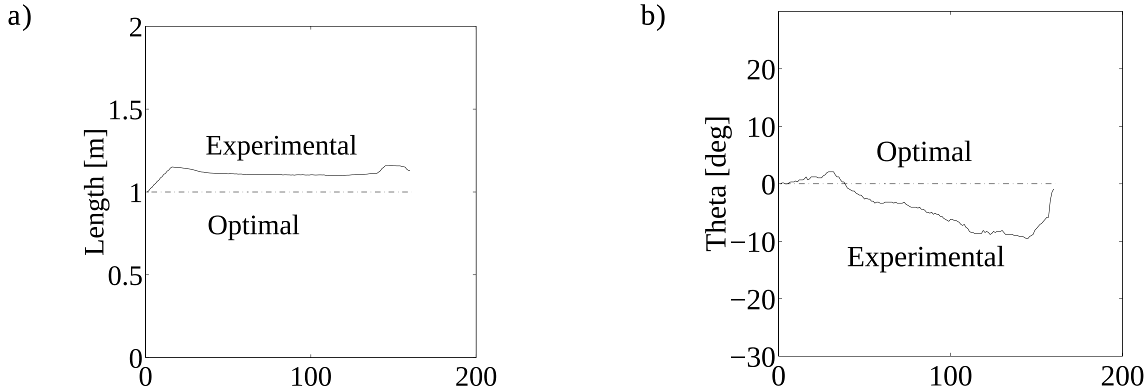

Figure 5.26: Variation of distance (a) and relative angle between platforms (b), $\theta_2 - \theta_1$, for a parallel parking maneuver.

turn maneuver are shown in Figure 5.25. We compare the variation in l (the distance between the bases of the two manipulators) and $\theta = \theta_2 - \theta_1$ (the relative angle between the two platforms) for the optimal motion plan and the experiment. Ideally, we expect to see no discrepancy between the two sets of plots. However, because the platform control is accomplished with dead-reckoning, we can see that the experimental trajectory drifts from the commanded trajectory with time. The angular discrepancy between the commanded (optimal) and the experimental trajectory is as much as 7 degrees.

In Figure 5.26, we show the same results for the parallel parking maneuver. The errors due to dead-reckoning are seen to be more pronounced especially if we look at the variation of l with time. Here the optimal motion plan requires l to be at its upper limit (1.1 meters). The experimental results show that l overshoots the optimal trajectory and therefore violates this constraint. Obviously, this can be circumvented if the optimal path is generated with a factor of safety on the constraints (e.g., with the limits on l between

0.95 and 1.05 meters). More importantly, it is clear that such an open loop approach to control may not be satisfactory for the types of maneuvers we are interested in.

The errors in dead-reckoning mainly occur when there are reversals in the paths of any of the drive motors or the passive wheels (castors in our case). The reversal of a drive motor invariably results in slip while the reversal of the path of castor causes it to change its configuration (the spherical ball of the castor will always trail the pin along which the vertical load is transmitted) which in turn results in frictional forces that indirectly cause the drive wheels to slip. In an experiment where we commanded a platform to go forward and backward on a 1 meter straight line track, the angular errors were observed to be 3 degrees per meter. This is consistent with the discrepancy between the experimental and the optimal trajectories in the plots.

5.7 Optimal gait generation for locomotion systems

In this section, we discuss another possible application of the general methodology for motion planning discussed in Chapter 3 for complex dynamical systems. We will discuss numerical methods for calculating the optimal input patterns for a class of locomotion systems characterized by a combination of nonholonomic constraints and group symmetries [98]. We also address the use of Lagrangian reduction techniques to improve the speed and performance of these methods. One of the central examples used to illustrate this general framework for locomotion is the snakeboard. Results from our investigations into optimal gaits for the snakeboard example are presented, and some of the interesting consequences of placing an optimality criterion on the input patterns are discussed.

The snakeboard exhibits several basic gait patterns, each with a specific desired target motion. In this work, we explore the ways in which the input patterns scale as the demands are increased on the system. In particular, we show that the character of the input patterns- that is, the type of gait chosen- may change according to the desired final state. We liken this to similar transitions found in biological organisms, e.g., as exhibited in a horse when shifting between a trot, gallop, or pace. A great deal of research and speculation has been invested in showing that an animal transitions between gaits as a means of optimizing (minimizing) the energy that is expended. The results presented in the following sections establishes a general framework for studying these and other issues related to optimality of gaits for undulatory locomotion systems.

5.7.1 Dynamic formulations and reduction

Locomotion problems can naturally be formulated as dynamic mechanical systems, and so are governed by Lagrange's equations. In this section, we develop the formulation of the constrained dynamics and address the use of reduction techniques (developed elsewhere [15, 95, 97]) to simplify the analysis of locomotion systems.

5.7.2 Review of Lagrangian dynamics

For the basic locomotion problem, let Q be the configuration space, describing both the position (and orientation) of the body and the internal shape of the locomotion system. We assume the existence of a Lagrangian function, $L(q, \dot{q})$, governing the dynamics of the system. The interaction of the locomotion system with its environment is modeled by k linear functionals, $\omega^1, \ldots, \omega^k$ [95]. We require that trajectories of the system satisfy:

$$\omega_i^j(q)\dot{q}^i = 0, \qquad j = 1, \ldots, k. \tag{5.29}$$

In some cases, the constraints alone will determine the motion of the system, given the shape changes as inputs. This is called the *kinematic* case [68]. More generally, however, the formulation of the dynamics of the system requires the use of Lagrange multipliers to modify Lagrange's equations. In this case, the full dynamics of the system are given by:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}^i}\right) - \frac{\partial L}{\partial q^i} = \lambda_j \omega_i^j + \tau_i. \tag{5.30}$$

Here, τ represents applied forces and torques, such as those applied within the body to change the internal shape, and the λ's are Lagrange multipliers, which loosely correspond to the magnitudes of the forces required to enforce the constraints. Notice that the Lagrange multipliers represent k additional state variables that must be solved for using Equation (5.29) and (5.30) (a total of $2n + k$ states). There are several ways to eliminate the Lagrange multipliers, although these methods often lead to a more complex set of governing equations complex, and can lead to singularities in the representation. In the next section, we describe one method for eliminating the multipliers that is particularly well-suited for problems of locomotion.

5.7.3 The reduction process

When studying locomotion systems, it is important to note that the configuration manifold, Q, can always be divided into two parts: the inertial *position* of the body and the internal *shape* of the system. Th position and orientation of the body will generally be a Lie subgroup of $SE(3)$; for example, $SE(2)$ for a snake or paramecium, or $SO(3)$ for a satellite or the falling cat. The remainder of the state of the system is just the internal shape of the body. We denote this internal shape by M, and call it the *shape space*. In the standard mathematical nomenclature, Q is called a *trivial principal fiber bundle* with *fibers* G and *base space* M. Q is "trivial" because the product structure is global, and it is a "principal" bundle because the fiber is a Lie group. The Lie group structure (which is common to *all* locomotion systems) leads to a systematic method, called *Lagrangian reduction*, for reducing the equations of motion to a more compact form.

Associated with a Lie group, G, is its Lie algebra, $\mathfrak{g}$. The Lie algebra can be identified with the tangent space at the identity, T_eG (see [92]). The reader familiar with screw theory will recognize the Lie algebra elements as defining twists. In particular, we will be interested in *body* velocities, where an element $\xi \in \mathfrak{g}$ is defined as $\xi = g^{-1}\dot{g}$.

Conservation laws naturally arise when a Lagrangian remains fixed under the action of a Lie group, G. More formally, let the action of $g \in G$ on $q = (h, r) \in G \times M = Q$ be given as $\Phi_g(h, r) = (gh, r)$, where gh represents the action of the group G on itself (often represented using homogeneous matrix multiplication). The Lagrangian function is said to be *G-invariant* if $L(\Phi_g(q), D_q\Phi_g v_q) = L(q, v_q)$ for all $g \in G$ and $(q, v_q) \in TQ$.

For all unconstrained systems, Noether's theorem [15] states that the invariance of the Lagrangian implies a momentum conservation law, e.g., conservation of linear and angular momentum. Examples of "locomotion" systems that obey these laws are the falling cat and the satellite with rotors. Letting p denote the body momentum, it is straightforward to show [95] that the conservation law takes the form:

$$\xi = g^{-1}\dot{g} = -\mathbb{A}(r)\dot{r} + \mathbb{I}(r)^{-1}p, \tag{5.31}$$

where $\mathbb{A}(r)$ is called the "local form" of the *mechanical connection*, and $\mathbb{I}$ is the *locked inertia tensor*. $\mathbb{I}(r)$ describes the total inertia of the system when all joints are frozen at configuration r. The *connection*, $\mathbb{A}$, plays a central role in understanding locomotion, and is developed in much greater detail in [15]. Its importance stems from the fact that it determines the robot's motion (note that $g^{-1}\dot{g}$ is the velocity of the robot's reference frame, as seen in body coordinates) as the combination of momentum, p, and internal shape changes, $(r, \dot{r})$. Hence, *the connection will determine how internal shape changes create a net robot motion*. Notice also that the information encoded in the term $\mathbb{A}(r)$ is a function of the shape, r, only.

The presence of external nonholonomic constraints, however, breaks these conservation laws. In fact, it is exactly the failure of the momentum conservation laws for these types of systems that is the key to dynamic undulatory locomotion (and many other forms of locomotion). It describes why the snakeboard example discussed below can build up momentum, even though the external forces (the forces of constraint) do no work on the system.

A full treatment of the reduction process is present in [15]. The equations that result from this process, however, are analogous with those seen in the unconstrained case, with the exception that the momentum is no longer fixed, but may vary, dependent on the internal shape. The interaction between constraints and momentum terms is summarized in the following reduced equations of motion, which has been used to characterize a wide range of locomotion systems [95].

$$\begin{aligned} g^{-1}\dot{g} &= -\mathbb{A}(r)\dot{r} + I(r)^{-1}p, & (5.32)\\ \dot{p} &= \sigma(r,\dot{r},p), & (5.33)\\ M(r)\ddot{r} &= -C(r,\dot{r}) + N(r,\dot{r},p) + \tau & (5.34) \end{aligned}$$

In this formulation, a *nonholonomic momentum*, p, is defined, with an associated governing equation for p (Equation (5.34)), called the *nonholonomic momentum equation* [15]. The form of Equation (5.33) is very similar to that found in the unconstrained case (Equation (5.31)), with the replacement of the mechanical connection by a *nonholonomic connection*, $\mathbb{A}$. The nonholonomic connection plays a similar role as above by formally expressing the intuitive relationships between internal shape changes ($\dot{r}$) and their effect on locomotion ($\xi = g^{-1}\dot{g}$). Finally, we point out that the each of the equations is independent of g. In particular, the shape equation (Equation (5.34)) is used to calculate the cost function in terms of the energy input into the system.

Finally, we highlight the special limiting cases of the reduced formulation given by Equation (5.33) - (5.34). In many examples, such as with certain models of mobile robots, inchworms, or paramecia, the symmetries are completely annihilated by the constraints, and so inertial effects are not present. This case is called the *principal kinematic case*, for which the connection reduces to

$$g^{-1}\dot{g} = -\mathbb{A}(r)\dot{r}. \qquad (5.35)$$

On the other hand, in the case of the falling cat, some models of underwater vehicles, satellites with rotors, and platform divers, there are no external constraints to influence the system. In this case, the equations reduce to those found in Equation (5.31). Furthermore, many of these systems are assumed to have zero initial momentum. In this case, the governing equations again reduce to Equation (5.35), with the nonholonomic connection being replaced by the mechanical connection.

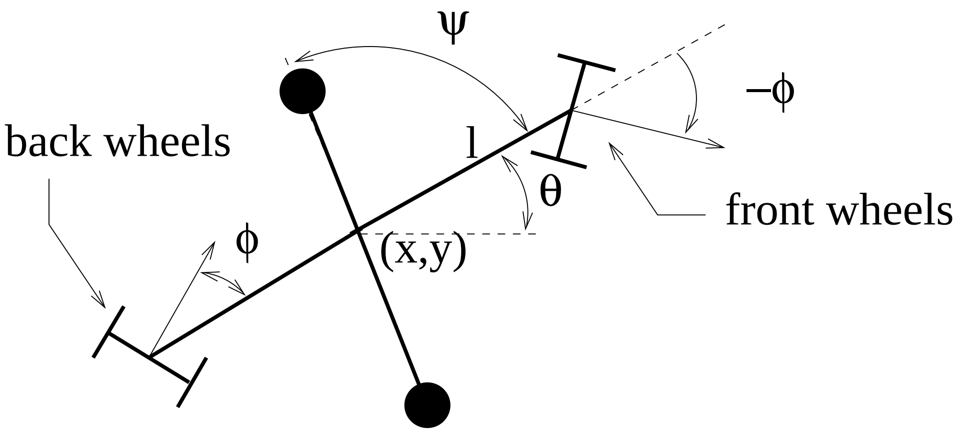

Figure 5.27: The snakeboard model.

5.7.4 The example of the snakeboard

The *Snakeboard* is a variant of the skateboard in which the passive wheel assemblies can pivot freely about a vertical axis. By coupling a twisting of the human torso with an appropriate turning of the wheels (where the turning is controlled by the rider's foot movements), a rider can generate a snake-like locomotion pattern *without* having to kick off the ground. A simplified model that captures the essential behavior of the snakeboard is shown in Figure 5.27. We assume for our problem that the front and rear wheel axles move through equal and opposite rotations. A momentum wheel rotates about a vertical axis through the center of mass, simulating the motion of a human torso.

The snakeboard's position variables, $(x, y, \theta) \in SE(2)$, are determined by a frame affixed to its center of mass. The shape variables are (ψ, ϕ), and so the configuration space is $Q = G \times M = SE(2) \times (\mathbb{S} \times \mathbb{S})$. The physical parameters for the system are the mass of the board, m; the inertia of the rotor, J_r; the inertia of the wheels about a vertical axis, J_w; and the half-length of the board, l. We make an additional assumption (see [15]) that the inertias of the system satisfy $J + J_r + 2J_w = ml^2$. This assumption greatly simplifies the equations of motion. The Lagrangian function for this problem is then:

$$L = \frac{1}{2}m(\dot{x}^2 + \dot{y}^2 + l^2\dot{\theta}^2) + \frac{1}{2}J_r(\dot{\psi}^2 + 2\dot{\psi}\dot{\theta}) + 2J_w\dot{\phi}^2 \qquad (5.36)$$

Control torques at the rotor and wheel axles are assumed, so $\tau = (0, 0, 0, \tau_\psi, \tau_\phi)$. The assumption that the wheels do not slip in the direction of the wheel axles determines two constraints of the form of Equation (5.29):

$$\begin{aligned} -\sin(\theta + \phi)\dot{x} + \cos(\theta + \phi)\dot{y} - l\cos(\phi)\dot{\theta} &= 0 \qquad (5.37) \\ -\sin(\theta - \phi)\dot{x} + \cos(\theta - \phi)\dot{y} + l\cos(\phi)\dot{\theta} &= 0 \qquad (5.38) \end{aligned}$$

One approach to finding the equations of motion for this system is to solve explicitly for the Lagrange multipliers. Instead, we can use the geometry of the system [95] to define the nonholonomic momentum as:

$$p = -2ml\cos^2\phi(\dot{x}\cos\theta + \dot{y}\sin\theta - l\dot{\theta}\tan\phi) + J_r\sin 2\phi\,\dot{\psi}. \tag{5.39}$$

The momentum, p is chosen to correspond to the snakeboard's total angular momentum about the instantaneous center of rotation (think about the circular path it would follow if the wheel angle, ϕ, were fixed).

Using the constraint equations and the formula for the nonholonomic momentum, we can then write down the equations governing $(\dot{x}, \dot{y}, \dot{\theta})$, involving the connection as in Equation (5.33):

$$g^{-1}\dot{g} = \begin{pmatrix} \frac{J_r}{2ml}\sin 2\phi & 0 \\ 0 & 0 \\ -\frac{J_r}{ml^2}\sin^2\phi & 0 \end{pmatrix}\begin{pmatrix} \dot{\psi} \\ \dot{\phi} \end{pmatrix} + \begin{pmatrix} -\frac{1}{2ml} \\ 0 \\ \frac{1}{2ml^2}\tan\phi \end{pmatrix} p \tag{5.40}$$

The first term on the right-hand side of this equation is the nonholonomic connection, $\mathbb{A}$.

Finally, the internal shape dynamics can be reduced as well [95], taking the form:

$$\begin{aligned} \left(J_r - \frac{J_r^2}{ml^2}\sin^2\phi\right)\ddot{\psi} & \quad -\frac{J_r^2}{2ml^2}\sin 2\phi\,\dot{\psi}\dot{\phi} + \frac{J_r}{2ml^2}\dot{\phi}p = \tau_\psi \\ 2J_w\ddot{\phi} & \quad = \quad \tau_\phi \end{aligned} \tag{5.41}$$

These six equations (Equation (5.40) - (5.41)) completely define the motion of the system. The use of reduction methods leads to a very concise and physically intuitive formulation of the dynamical relationships that govern the snakeboard's movement.

5.8 Numerical studies of snakeboard gaits

The example of a snakeboard for studying gait transitions is an important one for a variety of reasons. First, this example contains many of the features common to a wide spectrum of locomotion systems [97]. Also, the snakeboard provides an interesting array of gait possibilities. In previous works [79], detailed studies have been performed on several of these gaits (using open-loop inputs), and the system was shown to be locally controllable [95]. The results presented here provide insights into optimal motions and issues of gait transitions.

The physical parameters for all of the simulations shown here are:

$$\begin{array}{llll} m: & 6.0kg & l: & 0.2m \\ J_r: & 0.72kg\cdot m^2 & J_w: & 0.13kg\cdot m^2 \end{array} \tag{5.42}$$

These are approximate parameters taken from a prototype snakeboard used in the experiments. For each simulation, the initial and final values of the state variables are set to zero, unless otherwise specified. The cost function, C, is the time integral of the energy input into the system.

5.8.1 Forward motion

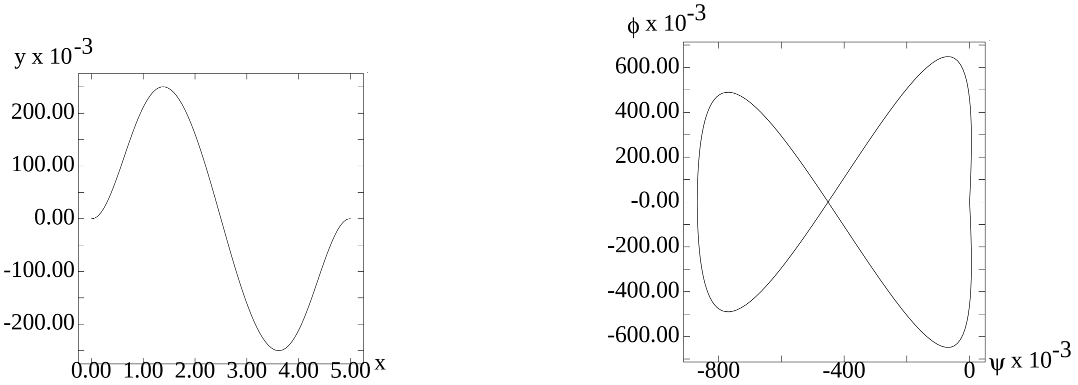

Figure 5.28: Forward motion of the snakeboard for zero initial and final momentum, along with the optimal inputs, ψ vs. ϕ.

The original studies of the snakeboard were inspired by the serpentine-like paths followed by the actual *Snakeboard* riders. It was observed that they generated a forward gait by synchronizing their torso and feet motions. This was called a “1-1” gait, implying the same frequency motion for each of the inputs. Coupling the inputs in this manner led to a net forward travel, with an associated generation of forward velocity, or momentum. In this section, we seek to optimize the cost function given a desired forward motion of $1m$. The optimal trajectory is given in Figure 5.28, along with a plot of the inputs used. Notice the figure-eight motion of the shape inputs. Many modern results in geometric analysis of nonholonomic systems suggest that so-called “area rules” play an important role in motion generation. Essentially, these rules relate closed loops (i.e., loops enclosing particular areas) in the shape space with proportional changes in position and orientation. The motions shown in Figure 5.28 are suggestive of a (signed) *area rule*— one that builds up and then takes away momentum, corresponding to a total signed area of zero— and a *moment rule*, corresponding to the net motion in the forward direction.

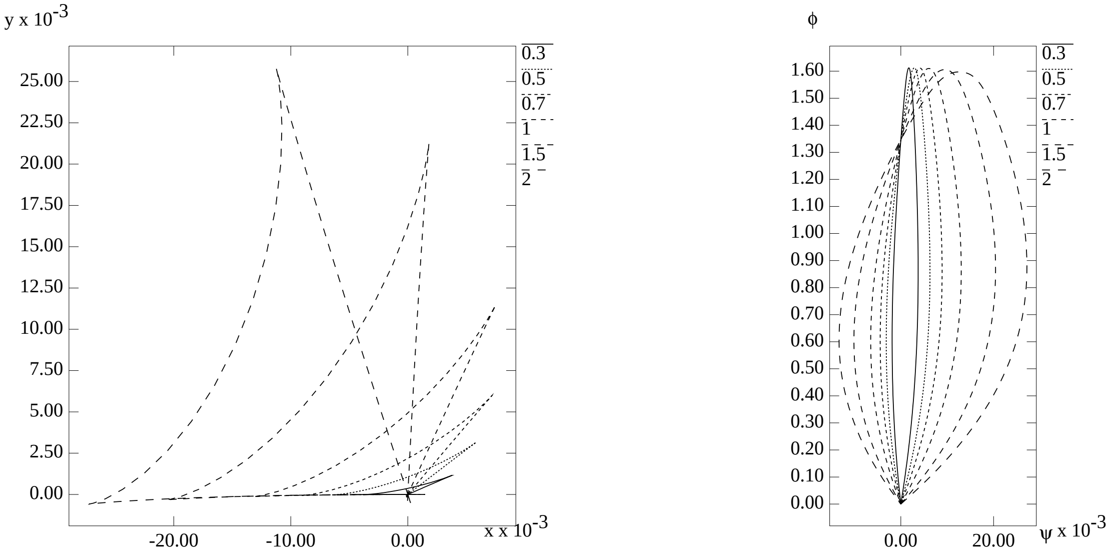

Figure 5.29: Rotate gait, with optimal inputs.

5.8.2 Rotate motion: "Three-point" turn

In the snakeboard, it was observed that a 2:1 input motion— that is, the rotor moving twice as fast as the wheels— resulted in a net rotation of the system. A parametric study of the optimal motions for generating such a rotation is shown in Figure 5.29, along with the optimal inputs. Notice, however, the scale on the axes— the motion is essentially rotating in place. This optimal gait is realized as a single, simple loop in shape space. First, the wheels are turned outward. Then, the rotor is turned, causing the body to rotate accordingly, until the system reaches the desired orientation, at which point the wheels are turned back in place (see Figure 5.29).

5.8.3 Lateral motion: Parallel parking

A very interesting motion found in the snakeboard (but not yet seen in *Snakeboard* riding) is a lateral motion, similar to a parallel parking maneuver. For the open loop system, this gait was generated using a 3:2 ratio of rotor movement to wheel movement. The optimal motion for a lateral motion of $1m$ is shown in Figure 5.30. In this motion, the final position of ψ is specified as $\pi/2$, explaining the lack of symmetry of this gait. The reader is reminded that while this type of motion may seem very natural for kinematic mobile robots, where the wheel angles are directly controlled, the motion in the snakeboard *cannot* occur without involving the *dynamics* of the system.

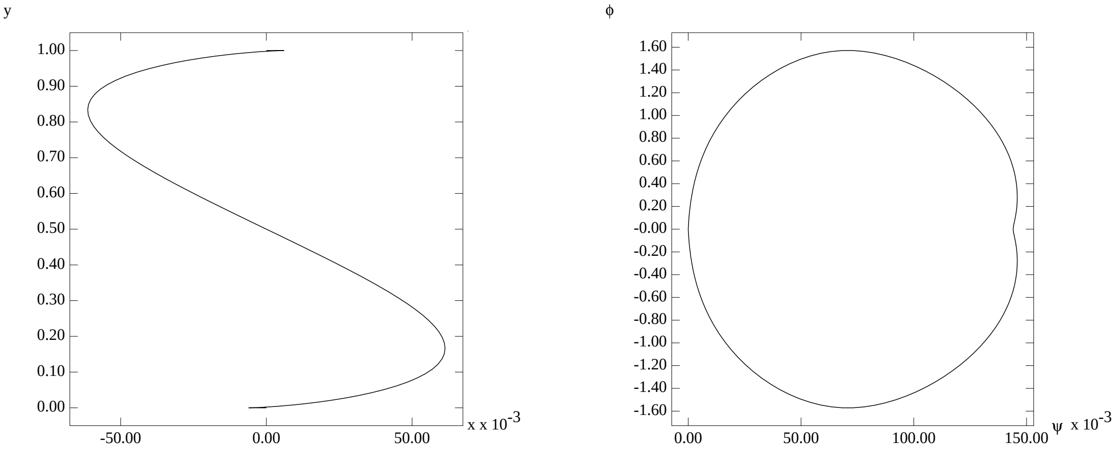

Figure 5.30: Parallel parking for the snakeboard.

5.8.4 Gait transitions

One of the most interesting aspects of combining optimal control theory with studies of locomotion is the ability to ask questions about optimality of gaits. With the methods developed here, it may be possible to analyze in simulation the dynamics of gait selection, such as that found in many biological creatures.

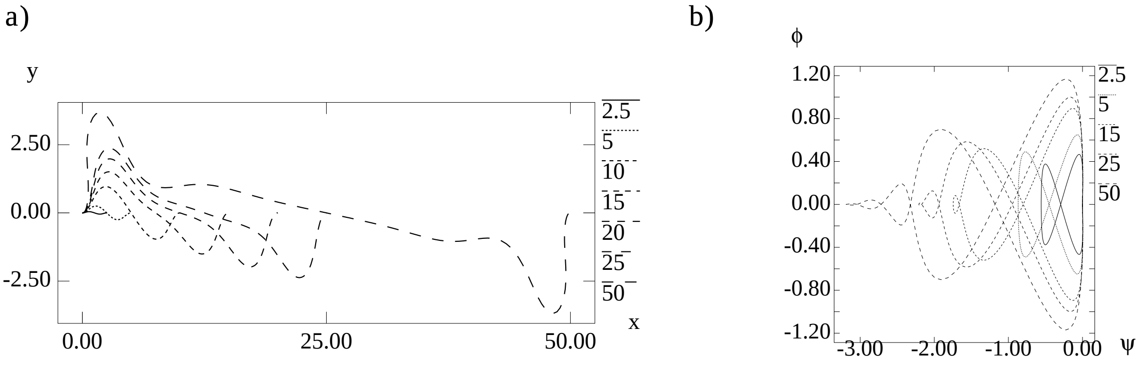

Figure 5.31: a) Gait transitions for increasing forward displacement and b) Inputs during gait transition in forward motion.

We examine three types of gait transitions. In Figure 5.31(a), the total forward distance traveled is varied. It is apparent that the optimal motion changes over the given range. More interesting is to observe the deformation of the input patterns. Figure 5.31(b) shows a very interesting series of changes, invoking parallels to a series of bifurcation points. This cascading pattern continues as the closed curve in shape space moves from two loops to three and even four loops.

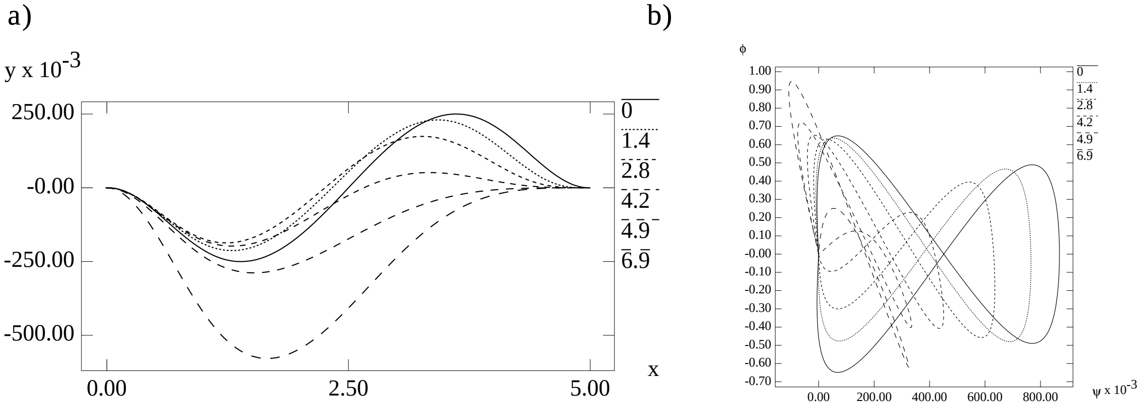

Figure 5.32: a) Gait transition when varying desired final momentum and b) Input patterns during gait transition.

The second parameter that was varied was the desired final momentum, for a fixed forward displacement (see Figure 5.32(a)). As the desired momentum value is increased, there is a definite shift between two types of input motions. The shifts in input patterns are shown in Figure 5.32(b), which shows the gradual distortion of a symmetric input pattern (for zero initial and final momenta) to a distorted figure-eight (although it does appear to retain the same basic structure throughout the deformation, suggesting a homotopic equivalence to this class of gaits). In Figure 5.33, we plot the cost function associated with various values of the final momentum. Notice that initially an increase in the desired final momentum leads to a *decrease* in the value of the cost function. This makes sense since a nonzero final momentum implies that the system does not have to expend energy in slowing down to zero momentum during the second half of the gait. Obviously, however, this effect cannot last indefinitely, since there must be a cross-over point at which the energy exerted to achieve a particular momentum is greater than the energy input in the initial start-and-stop motion. This cross-over point occurs around $p = 4.0$. At this point in Figure 5.32(a), the gait appears to have fully transitioned over to a unimodal shape.

Finally, we investigate the transition between two distinct types of gaits, by varying parameters between a forward, “drive” gait and a lateral motion, “parallel parking” gait. It is interesting to note that the smooth transition of the x-y pattern followed by the gaits is not as apparent in the input patterns, which are quite complex.

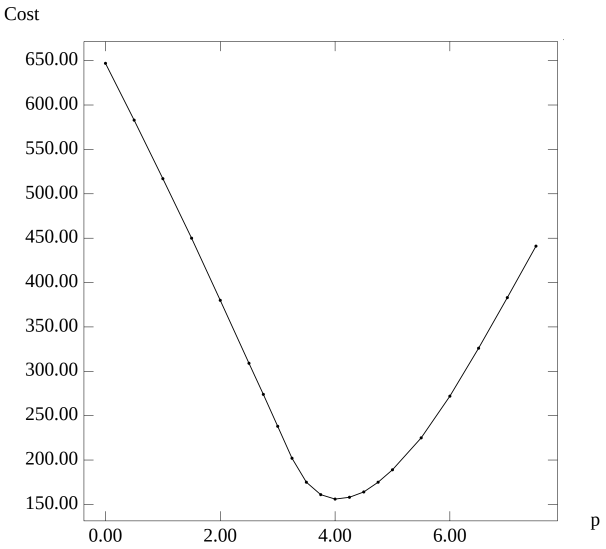

Figure 5.33: Cost function vs. desired final forward position.

5.9 Discussion

In this chapter, we presented a general approach for robot motion planning with geometric, kinematic and dynamic constraints and demonstrated applications for planning cooperative manipulation tasks with two cooperating mobile manipulators. The geometric constraints include constraints imposed by the presence of obstacles while the kinematic constraints may be holonomic or nonholonomic. The dynamic constraints include the equations of motion and constraints on contact forces. In addition, the constraints may be inequality constraints. We determine trajectories and actuator forces that minimize a suitably chosen integral cost function while satisfying the constraints.

The motion planning problem was formulated as a problem in the calculus of variations and solved using the integral form of the necessary conditions. There are several advantages to this method as presented in Section 3.7. In the problems considered in this chapter, the dimension of the extended state space vector X was as high as 58 and the number of mesh points $N = 100$. These solutions took approximately five minutes on a 200 MHz. Silicon Graphics Impact workstation. Approximate solutions with 20 mesh points can be computed in less than two minutes.

An example of the applicability of the proposed method for complex dynamical systems is the snakeboard. We studied the various gait patterns that generate the required motion. We also studied the effect of distance traveled on gait transitions for the snakeboard

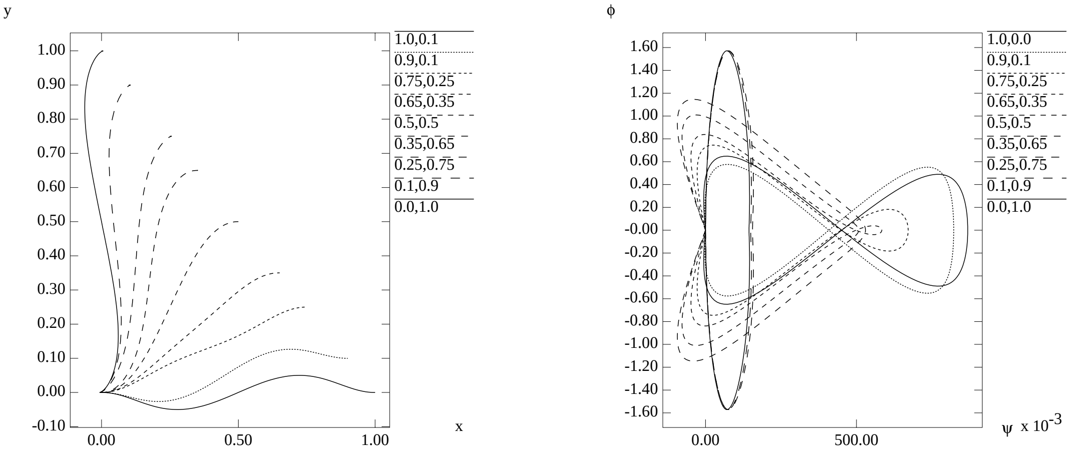

Figure 5.34: Gait transition between parallel parking and drive gaits.

and the effect of varying values of final momentum for a fixed travel distance. The results showed a gradual distortion in the symmetric input pattern for zero initial and final momenta to a distorted figure-eight while maintaining the same basic structure throughout the deformation for various values of final momenta.

One of the main disadvantages of this methodology is that the number of variables in the formulation increases rapidly for very complex problems. Thus, this technique does not lend itself to the solution of such general motion planning problems as the piano movers problem. Instead, we can use this approach to generate solutions for problems with a small number of robots/obstacles or to refine nominal solutions that are achieved by geometric planners in complicated environments [36, 67]. Since this method suffers from higher computational costs and also complexity for finding motion plans for larger formations of robots, it is meaningful to combine centralized and decentralized motion planning techniques in order to control a larger formation of robots. This is the essence of the material in the next chapter.

Chapter 6

Motion planning for formations of robots

6.1 Introduction

In the last chapter, we discussed how to obtain optimal solutions for a team of cooperating robots. Clearly, extension of this methodology to a larger number of robots (and in particular controlling a formation of robots) is quite straightforward. However, the search for optimal motion plans can be computationally very expensive, particularly as the number of robots or degrees of freedom of the system gets larger ($n > 2$). For this reason, we pursue in the current work easily computable feedback laws that can be used in conjunction with a higher level (but lower complexity) motion planner. The motion planning problem for such a team of robots can be stated as:

Given any team, $\mathcal{C}$, of n robots with a given lead robot, and initial positions and orientations and goal positions and orientations in workspace $\mathcal{W}$, generate a continuous trajectory for $\mathcal{C}$ avoiding contact with the obstacles, $\mathcal{O}_i$, or between robots subject to various geometric and kinematic constraints of the system.

For a team of mobile robots, geometric constraints are typically constraints due to obstacles in the environment and kinematic constraints are essentially nonholonomic constraints. Typically, these mobile robots have very good velocity controllers and are therefore able to follow a prescribed trajectory. It is therefore useful to model the control problem by tracking the velocity as the input variable. Thus dynamic modeling is not required. We are interested in finding a kinematic motion plan for such a system.

The multi-robot motion planning problem can be approached by assuming that the sensory information collected by each robot is available to a central planner. In Chapter 5, we used variational methods to obtain optimal motion plans for multiple robots using such a centralized planner. The main disadvantage is that the search for optimal motion plans can be computationally very expensive, particularly as the number of robots (the number of degrees of freedom) becomes large. The multi-robot control and planning problems are more tractable if each robot is assumed to be autonomous and controlled by an independent control law. Sensor-based planners for independently controlled robots are described in [141], while decentralized feedback control laws are developed in [117]. The main disadvantage is that it is difficult to establish guarantees or bounds on the system performance. In this chapter, we will show that independently controlled robots with appropriately designed control laws can maintain a desired formation with provable measures on the performance of the entire system. This is in contrast to the behavior based control paradigm [9, 10, 89] where such measures can only be obtained empirically.

In this chapter, we discuss several fundamental issues underlying the control and coordination of multiple autonomous robots. We focus on tasks in which the robots are required to follow a trajectory while maintaining a desired formation and avoiding obstacles. While these issues arise in any situation in which it is necessary to coordinate a large number of independently controlled robots, they are particularly important in military applications [3] where vehicles are often required to maintain a close formation. In autonomous vehicle highway systems [86, 123], there is a need to control a platoon of cars in formation while allowing the cars to perform such maneuvers as lane changes, merges, or avoiding obstacles on the road. In these applications, it is necessary for the team to accommodate constraints such as those imposed by obstacles, while maintaining the formation as best as possible. In a situation as the one shown in Figure 6.1, for example, it might be necessary to change the formation in order to negotiate the obstacle, and then reform the original formation.

We model any given formation of robots as a triple $(g, r, \mathcal{H})$, where g represents the ***group*** variables, $(x, y, \theta) \in SE(2)$, of the lead robot in the formation, r is a set of ***shape*** variables that describe the relative positions of the other robots in the team, and $\mathcal{H}$ is a ***control graph*** which describes the control strategy (or behavior) used by each robot, and the dependence of its trajectory on that of one or more of its neighbors. Thus, g denotes the gross motion of the team of robots, r governs the 'shape of the formation', and $\mathcal{H}$ is the discrete coordination protocol employed by the team. The ***formation*** of the team is described by the pair $(r, \mathcal{H})$.

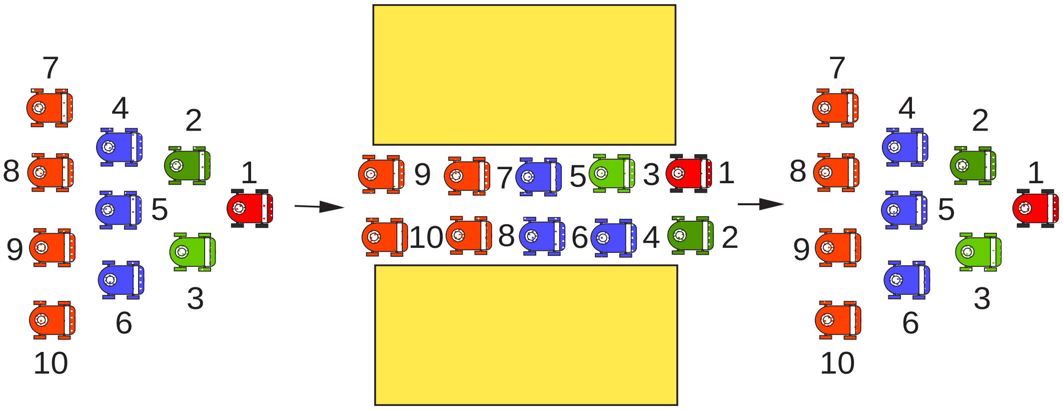

Figure 6.1: Formation of robots changing shapes.

The control of a system of autonomous robots that can perform such tasks requires coordination at several different levels. At the lowest level, it is necessary for each robot to control its motion, to avoid collisions with its neighbors, and to move along a desired trajectory. At an immediately supervisory level it is necessary to outline a strategy for maintaining a formation. In a system of two robots, this might be as simple as a leader-follower strategy where a leader plans and follows a preferred trajectory while the follower maintains a specified relative position and orientation with respect to the leader. This motion of the lead robot is specified by the group variable g and the motion of the follower robot is specified by the shape variable r. When the number of robots is more than two, there might be more than one leader-follower pairs or other more complex structure of interaction. This level of coordination is discrete. In contrast to the control laws for individual robots where the state variables and inputs change continuously, there are only a finite number of distinct formations and the changes in formations represent discrete changes in the organization of the team. Each such formation of robots represents the underlying control strategy and this control strategy is modeled by a directed graph $\mathcal{H}$. Finally, at the highest level, it is necessary to plan the trajectory for the entire team of robots based on the available terrain information. At this stage $(g, r, \mathcal{H})$ are planned concurrently based on the requirements of the task. Thus, based on the above problem formulation and the discussion, we can plan the motion for any given number of robots subject to geometric and kinematic constraints.

Since our method for controlling formations must rely on local sensor-based information, we must develop control laws that are based on such local information. We describe two scenarios for feedback control within a formation [32]. In the first scenario, one robot follows another by controlling the relative distance and orientation between the two. This

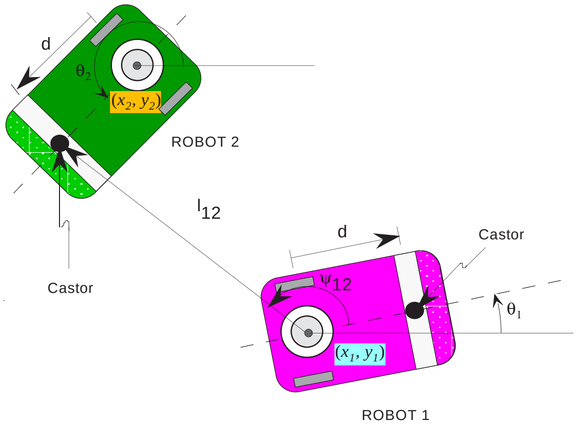

Figure 6.2: Notation for $l - \psi$ control.

situation is applicable to all formations in which each robot has one leader except for the lead robot. Thus it can be used, for example, for robots marching in a single file. We label such a controller as the $l - \psi$ controller or *Controller I*. A schematic of this controller is shown in Figure 6.2 where we have two non-holonomic mobile robots and the aim of this controller is to maintain the desired relative angle, ψ_{12}^d, and a desired separation, l_{12}^d between the leader and the follower.

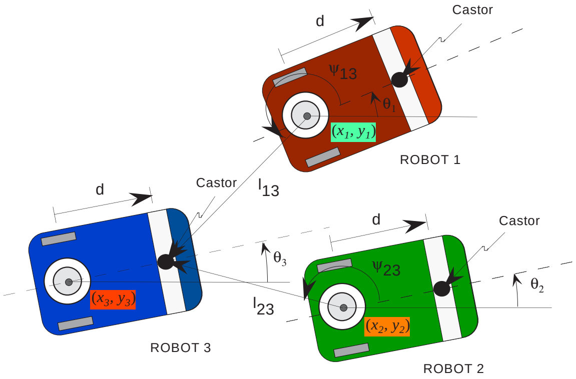

Figure 6.3: Notation for $l - l$ control.

In the second scenario, a robot maintains its position in the formation by maintaining a specified distance from two robots, or from one robot and an obstacle in the environment. This behavior is useful for robots that are constrained by more than one robot (or obstacle) in the formation. We label such a controller as the $l - l$ controller or *Controller II*. A schematic of this controller is shown in Figure 6.3 where we have two non-holonomic mobile robots and the aim of this controller is to maintain the desired separation l_{13}^d and l_{23}^d between the follower robot and its two leaders.

We can represent a formation of robots in a graph theoretic setting where each robot has a directional dependence on the other robots in the formation. A dependence of robot A on robot B represents the dependence of the position (and orientation) of robot A on that of B. Based on the two types of control laws proposed above, it is possible to maintain arbitrary formations such as a triangular formation G, a straight line formation H, or a rectangular formation, J as shown in Figure 6.4 depending on the constraints in the environment.

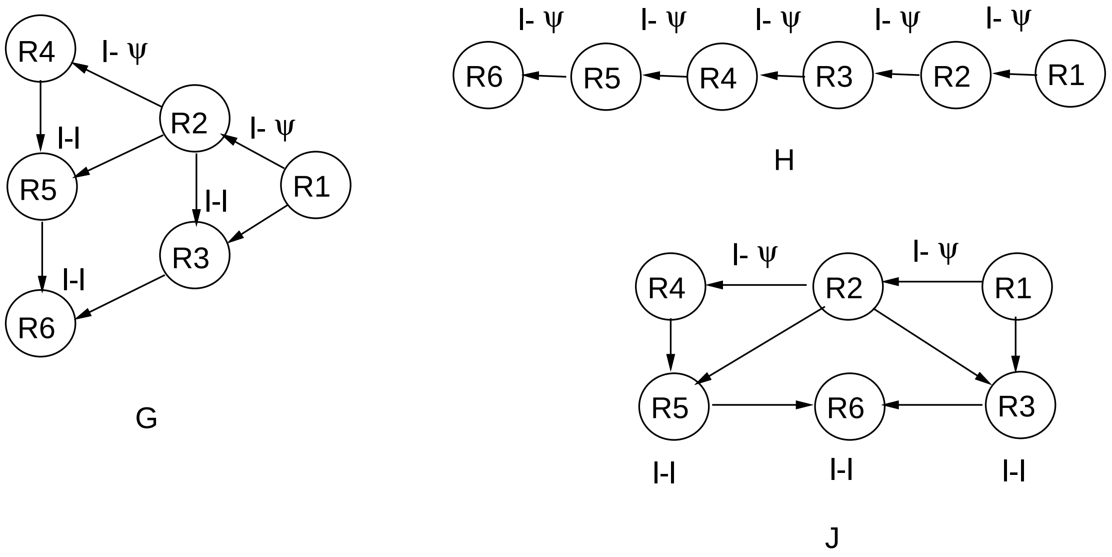

Figure 6.4: Isomorphic and Non-isomorphic digraphs.

In this chapter, we first describe our basic formulation of the problem where we model a team of robots as a triple consisting of a group element that describes the gross motion of the team, a set of shape variables that describe the relative positions of the robots, and a control graph that describes the behaviors of the robots in the formation. We then develop feedback control laws that are essential to maintain the desired position of a robot within the formation and the overall shape of a formation. Later, we characterize the different formations possible for a team of n robots by enumerating the control graphs. We prove an upper bound on the total number of possible control graphs and enumerate the equivalence classes of digraphs that are relevant to this problem. We describe a discrete coordination strategy for changing formations in terms of a sequence of transitions in the control graph that satisfy the constraints in the problem. We present several examples to demonstrate the general approach and show how a team of robots can automatically change the shape of the formation and/or its control graphs in order to circumvent obstacles (for example, to squeeze through a narrow passage).

6.2 Problem formulation

We now develop a framework for formalizing the basic issues addressed above. Before getting into details, it is useful to review the basic assumptions underlying this work. Each robot is equipped with the sensing and computational hardware that allow it to detect the distance to the nearest obstacle and the relative position and orientation of the neighboring robots in the team. Further, we assume that each team has a designated *lead robot* which has the responsibility of planning the trajectory for the team. The trajectory planning by the lead robot may be based on an approximate model of the team and the formation without specifics of the individual robots. We presented the framework for solving optimal control problems having equality and inequality constraints in Chapter 3 and applied it to solve the robot motion planning problem for two robots in Chapter 5. Based on the methodology of Chapter 5, the optimal motion plan for the kinematic motion planning problem of the lead robot is straightforward. Thus the lead robot follows this optimal trajectory and the other robots in the team follow the lead robot while maintaining the desired formation.

The configuration space for a team of n robots in the plane is given by the Cartesian product of n copies of $SE(2)$ *i.e.*, $SE(2) \times SE(2) \times \cdots \times SE(2)$. We model the team of robots by a triple $(g, r, \mathcal{H})$, where g represents the ***group*** variables, $(x, y, \theta) \in SE(2)$, of the lead robot in the formation, r is a set of ***shape*** variables that describe the relative positions of the other robots in the team, and $\mathcal{H}$ is a ***control graph*** which describes the control strategy (or behavior) used by each robot, and the dependence of its trajectory on that of one or more of its neighbors. When viewed in this framework, the problem of locomotion of a team of robots can be broken down into three sub-problems.

1. *Trajectory planning*: The determination of the optimal trajectory $g(t) \in SE(2)$ for the lead robot;

2. *Robot control*: The design of continuous robot control laws that enable the robots to maintain the desired shape variables using the strategy prescribed by the control graph $\mathcal{H}$ and also being able to modify the shape of the formation; and

3. *Formation control*: The design of the protocol for effecting changes in the control graph, $\mathcal{H}$, and the changes in shape r, to accommodate constraints imposed by the

task, including those associated with obstacles.

In Chapters 3 and 5, we developed a set of algorithms grounded in the methods of variational calculus for solving the trajectory planning problem with state and input constraints. This provides a solution to Item 1 above. In this chapter, we will focus on Items 2 and 3. We first develop a set of decentralized control laws that allow each robot to maintain a desired position within a formation and enables changes in the shape of the team formation. These control laws have the advantage of providing easily computable control inputs in real time, with provable performance for the entire system. We then focus on changes in formation which do not require a change in the control graph and later develop tools to incorporate changes in coordination protocol or control graph that describes the behavior of each robot in relation to its neighbors. In particular, we enumerate the set of possible control graphs for a team of n robots, and develop an algorithm that yields a sequence of intermediate control graphs for transitions between two arbitrary formations.

6.3 Control laws for shape variables

In this section we develop two types of feedback controllers for maintaining a formation of a team of mobile robots. These control laws are useful for maintaining either: (a) the desired separation and relative angle between the leader and the follower robot as shown in Figure 6.2 or (b) the desired separation of the follower robot from its two leaders as shown in Figure 6.3.

The standard kinematic equations for a nonholonomic robot is given by:

$$\begin{aligned} \dot{x}_i &= v_i \cos\theta_i \\ \dot{y}_i &= v_i \sin\theta_i \\ \dot{\theta}_i &= \omega_i \end{aligned} \tag{6.1}$$

where the subscript, i, refers to the i^{th} robot. With this notation, v_i is the generalized velocity along the allowable direction of motion for the mobile robot, and ω_i is the angular velocity about the center of the wheel base. Each robot is driven by two coaxial, powered wheels with a passive, front, castor-like wheel. The theory and the methods presented here are easily adapted to other nonholonomic models of wheeled robots. Our motivation for using Equation (6.1) comes from our experimental testbed shown in Figure 5.2.

6.3.1 Controller I: $l - \psi$ control

Figure 6.2 shows a system of two nonholonomic mobile robots separated by a distance of l_{12} between the center of the first robot and the front castor of the second robot. The distance between the castor and the center of axis of the wheels of each robot is denoted by d. Each robot has two actuated degrees-of-freedom. The two robots are not physically coupled in any way and the aim of the feedback controller is to maintain the desired formation. The state of the follower robot is given by: $[l_{12}, \psi_{12}, \theta_2]^T$

In the $l - \psi$ control mode for two mobile robots, the aim is to maintain a desired length, l_{12}^d and a desired relative angle, ψ_{12}^d between the two robots. The kinematic equations for the system of two mobile robots shown in Figure 6.2 is given by Equation (6.1) for the first robot and:

$$\begin{aligned} \dot{l}_{12} &= v_2 \cos\gamma_1 - v_1 \cos\psi_{12} + d\omega_2 \sin\gamma_1 \\ \dot{\psi}_{12} &= \frac{1}{l_{12}} \left(v_1 \sin\psi_{12} - v_2 \sin\gamma_1 + d\omega_2 \cos\gamma_1 - l_{12}\omega_1 \right) \\ \dot{\theta}_2 &= \omega_2 \end{aligned} \tag{6.2}$$

for the second robot where, $\gamma_1 = \theta_1 + \psi_{12} - \theta_2$ and v_i, ω_i $(i = 1, 2)$, are the linear and angular velocities at the center of the axle of each robot. In order to avoid collisions between robots, we will require that $l_{12} > 2d$.

We use standard techniques of I/O linearization [4] to generate a control law that gives exponentially convergent solutions in the internal shape variables l_{12} and ψ_{12}. These control laws are given by:

$$\begin{aligned} \omega_2 &= \frac{\cos\gamma_1}{d} \Big(\alpha_2 l_{12}(\psi_{12}^d - \psi_{12}) - v_1 \sin\psi_{12} + l_{12}\omega_1 + \rho_{12} \sin\gamma_1 \Big) \\ v_2 &= \rho_{12} - d\omega_2 \tan\gamma_1 \end{aligned} \tag{6.3}$$

where

$$\rho_{12} = \frac{\alpha_1(l_{12}^d - l_{12}) + v_1 \cos\psi_{12}}{\cos\gamma_1}$$

The linear velocity, v_2 shows a dependence on the error in the desired and actual separation between the two robots, while the angular velocity, ω_2 in addition to the error in the desired and actual separation, also depends on the error in the relative orientation, *i.e.*, ψ_{12}. The above control law leads to the following dynamics in the $l - \psi$ variables:

$$\begin{aligned} \dot{l}_{12} &= \alpha_1(l_{12}^d - l_{12}) \\ \dot{\psi}_{12} &= \alpha_2(\psi_{12}^d - \psi_{12}) \end{aligned} \tag{6.4}$$

Theorem 6.3.1 *Assume the system of two mobile robots shown in Figure 6.2 and the associated control law in Equation (6.3). For the motion of the lead robot following a circular path, $v_1 = K_1$, $\omega_1 = K_2$, θ_2 is locally asymptotically convergent to $\theta_1(t) + \psi_{12}^d - \beta_2 - \arccos(K_2/\beta_1)$, where*

$$\beta_1 = \sqrt{\left(\tfrac{K_2 l_{12} + K_1 \sin\psi_{12}^d}{d}\right)^2 + \left(\tfrac{K_1 \cos\psi_{12}^d}{d}\right)^2} \qquad \textit{and} \qquad \beta_2 = \arctan\left(\tfrac{K_1 \cos\psi_{12}^d}{K_2 l_{12} + K_1 \sin\psi_{12}^d}\right)$$

Proof: Since l_{12} and ψ_{12} are exponentially stabilized, we focus our attention on the zero dynamics of the system, which for this case is given in terms of θ_2. We seek to show that the zero dynamics of the system are asymptotically stable, *i.e.*, the system is *asymptotically minimum phase* [4].

The differential equation for θ_2 is given by:

$$\begin{aligned}\dot{\theta}_2 = \quad & \frac{K_2 l_{12} + K_1 \sin\psi_{12}^d}{d}\cos(K_2 t - \theta_2 + \theta_{10} + \psi_{12}^d) + \\ & \frac{K_1 \cos\psi_{12}^d}{d}\sin(K_2 t - \theta_2 + \theta_{10} + \psi_{12}^d)\end{aligned}$$

The above equation can be re-written as:

$$\dot{\theta}_2 = \beta_1 \cos(K_2 t - \theta_2 + \theta_{10} + \psi_{12}^d - \beta_2)$$

where β_1 and β_2 are as given in the proposition.

We make a change of variables to $\delta_{12} = \theta_1 - \theta_2 = K_2 t + \theta_{10} - \theta_2$, which gives the following differential equation:

$$\dot{\delta}_{12} = K_2 - \beta_1 \cos(\delta_{12} + \psi_{12}^d - \beta_2). \tag{6.5}$$

The equilibrium point, $\delta_{12}^e = -\psi_{12}^d + \beta_2 + \arccos(K_2/\beta_1)$, is easily shown to be asymptotically stable, using a linearization of Equation (6.5). Thus, the orientation of the lead robot will locally converge to $\theta_2^e = \theta_1 - \delta_{12}^e = \theta_1(t) + \psi_{12}^d - \beta_2 - \arccos(K_2/\beta_1)$. ■

Corollary 6.3.2 *For the case that the lead robot follows a straight line ($\omega_1 = K_2 = 0$), θ_2 converges exponentially to θ_{10}.*

Proof: For $v_1 = K_1$ and $\omega_1 = 0$, we obtain the following expression for ω_2 after some algebraic manipulations:

$$\omega_2 = \dot{\theta}_2 = \frac{K_1}{d}\sin(\theta_{10} - \theta_2) \tag{6.6}$$

where, θ_{10} is the initial orientation of the first robot at the beginning of the motion (and maintained as a constant throughout the motion since $\omega_1 = 0 \Rightarrow \theta_1(t) = \theta_{10}$). Integrating the expression and assuming $\theta_2(0) = \theta_{20}$, we get:

$$\tan\left(\frac{\theta_2 - \theta_{10}}{2}\right) = \tan\left(\frac{\theta_{20} - \theta_{10}}{2}\right) \exp^{-\frac{K_1}{d}t}$$

As $t \to \infty$, θ_2 exponentially converges to θ_{10}. ■

Remarks:

1. Note that we can solve explicitly for the motion of θ_2 as

$$\frac{\tan\frac{\delta_{12}+\psi_{12}^d-\beta_2}{2} + \sqrt{\frac{\beta_1-K_2}{\beta_1+K_2}}}{\tan\frac{\delta_{12}+\psi_{12}^d-\beta_2}{2} - \sqrt{\frac{\beta_1-K_2}{\beta_1+K_2}}} = C_0 \exp^{-\sqrt{\beta_1^2-K_2^2}\,t}$$

   where $\delta_{12} = K_2 t + \theta_{10} - \theta_2$, β_1 and β_2 are as defined in the proposition and $\beta_1^2 > K_2^2$ (since $l_{12} > 2d$). C_0 is the constant of integration.

2. The optimal point-to-point paths of a nonholonomic car with constraints on the turning radius were shown by Reeds and Shepp [104] to be composed of straight lines and circular arcs. This is seen to be generally true even for more complicated systems from the numerical results presented in Chapter 5. In this section the convergence of solutions has been obtained without placing any restrictions on K_1 and K_2. Thus, the $l - \psi$ controlled robot can track, with exponential convergence, any straight line, circular arc, or rotate in place motion of the lead robot thereby following the optimal trajectory generated by the lead robot.

6.3.2 Controller II: $\mathrm{l} - \mathrm{l}$ control

Figure 6.3 shows a system of three nonholonomic mobile robots. The task involves stabilizing the distance of the third robot from the other two robots. Thus the state of the third robot is specified by $[l_{13}, l_{23}, \theta_3]^T$. Distances are measured from the center of the axle of the first two robots to the castor of the third robot which is offset by d from its axle. Each robot has two actuated degrees-of-freedom. The three robots are not physically coupled in any way and the aim of the feedback controller is to maintain the desired formation.

In the $l - l$ control, the aim is to maintain the desired lengths, l_{13}^d and l_{23}^d of the third robot from its two leaders. Again, we assume that both l_{13} and $l_{23} > 2d$. Additionally, we require that the follower robot never lies on the line connecting the two lead robots as the control law for ω_3 becomes singular (see Equation (6.8)). In particular, this requires that the two lead robots never separate by a distance greater than or equal to $l_{13}^d + l_{23}^d$.

The kinematic equations for the system of three mobile robots shown in Figure 6.3 is given by Equation (6.1) for $i = 1, 2$ and

$$\begin{aligned} \dot{l}_{13} &= v_3 \cos\gamma_1 - v_1 \cos\psi_{13} + d\omega_3 \sin\gamma_1 \\ \dot{l}_{23} &= v_3 \cos\gamma_2 - v_2 \cos\psi_{23} + d\omega_3 \sin\gamma_2 \\ \dot{\theta}_3 &= \omega_3 \end{aligned} \tag{6.7}$$

for the third robot; where, $\gamma_i = \theta_i + \psi_{i3} - \theta_3$ $(i = 1, 2)$.

Again, we use I/O linearization to generate a feedback control law:

$$\begin{aligned} \omega_3 &= \frac{\alpha_1(l_{13}^d - l_{13})\cos\gamma_2 + v_1\cos\psi_{13}\cos\gamma_2 - \alpha_2(l_{23}^d - l_{23})\cos\gamma_1 - v_2\cos\psi_{23}\cos\gamma_1}{d\sin(\gamma_1 - \gamma_2)} \\ v_3 &= \frac{\alpha_1(l_{13}^d - l_{13}) + v_1\cos\psi_{13} - d\omega_3\sin\gamma_1}{\cos\gamma_1} \end{aligned} \tag{6.8}$$

This gives exponential convergence for the controlled variables:

$$\begin{aligned} \dot{l}_{13} &= \alpha_1(l_{13}^d - l_{13}) \\ \dot{l}_{23} &= \alpha_2(l_{23}^d - l_{23}). \end{aligned} \tag{6.9}$$

As seen from the expressions of the control laws, ω_3 and v_3, it is clear that the error in the separation between the robots appears in the equations.

Theorem 6.3.3 *Assume a system of three mobile robots as in Figure 6.3 and the associated control law given by Equation (6.8). For a straight line parallel motion of the first two robots (constant velocity $v_1 = v_2 = K$, $\omega_1 = \omega_2 = 0$ and $\theta_{10} = \theta_{20} = \theta_0$), θ_3 locally converges exponentially to $\theta_3^e = \theta_0$ and $\psi_{13}(t) = \psi_{13}(0)$ and $\psi_{23}(t) = \psi_{23}(0)$.*

Proof: The proofs are similar in nature to those presented for $l - \psi$ feedback control. Restricting our attention to the zero dynamics, the above $l - l$ control law gives

$$\begin{aligned} \dot{\theta}_3 &= \frac{K}{d}\sin(\theta_0 - \theta_3) \\ \dot{\psi}_{13} &= 0 \\ \dot{\psi}_{23} &= 0 \end{aligned} \tag{6.10}$$

Integrating the expression and assuming $\theta_3(0) = \theta_{30}$, we get:

$$\tan\left(\frac{\theta_3 - \theta_0}{2}\right) = \tan\left(\frac{\theta_{30} - \theta_0}{2}\right)\exp^{-\frac{K}{d}t}.$$

As $t \to \infty$, θ_3 exponentially converges to θ_0. Hence, the solution $\theta_3^e = \theta_0$ is a stable equilibrium point. Similarly the other two differential equations for ψ_{13} and ψ_{23} gives constant value for all time, t, namely, $\psi_{13}(0)$ and $\psi_{23}(0)$ respectively. ■

Thus, based on the type of control law employed to control the motion of a robot, the configuration variables of the robot will change. For example in the case of $l-\psi$ control, the configuration variables of the controlled robot are given by $(l_{ik}, \psi_{ik}, \theta_k)$ and in the case of $l-l$ control, it is given by $(l_{ik}, l_{jk}, \theta_k)$, where i, j refers to the subscript for the leader robot/robots and k refers to the subscript for the follower robot. For a long chain of robots moving with a particular formation, the lead robot in the chain plans an optimal path, avoiding any obstacles. Next, the velocity and the trajectory information of the lead robot is conveyed down the chain to the first generation of robots and they in turn act as leaders for the subsequent generations. Alternatively, the follower robots could estimate the velocities of the lead/leader robots based on sensory information. This will allow each of the robots in the formation to maneuver fully autonomously.

6.4 Modeling formations

A formation of n robots has one designated ***lead robot*** called R_1 that directly or indirectly controls all other (follower) robots in the formation. Within the formation, the follower robots will be dependent on other robots for their motion. We term these robots as ***leaders*** to designate that they lead other follower robots, but distinguish them from unique lead robot R_1. For example, in Figure 6.4, R_1 is the lead robot and R_2, R_3, R_4 and R_5 are all leaders each with one or more followers.

6.4.1 Control graphs

We can think of a formation as being made up of two components: a) the ***shape variables***, r, that describe its internal state and b) a directed relational graph structure, the ***control graph*** or ***digraph***, $\mathcal{H}$, that represents the internal topology of the formation. Two digraphs are said to be isomorphic if there is a one-to-one correspondence between their point sets and preserves adjacency and directionality between vertices (refer Equation (6.20) for a precise definition).

A transition from one formation to another may or may not involve a change in the control graph. Consider digraphs, G and H shown in Figure 6.4 to illustrate the point. R_5 in G follows R_2 and R_4 while R_5 in H follows R_4. G is not isomorphic to H since the connectivity of R_3, R_4, R_5 and R_6 is different in both the control graphs. In the same figure is another control graph J, which is isomorphic to G.

One possible way of representing a digraph is through an $n \times n$ adjacency matrix. For example, the digraph G in Figure 6.4 is given by:

$$G = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

In the representation above, the (i, j) element represents an incoming edge from robot R_i to R_j. We represent the rows and columns of G by the vertices which represent the robots, *i.e.*, R_1 through R_6. Now for example, since there is an arrow from R_1 to R_2 and R_1 to R_3, the first row for R_1 has 1 in the $(1, 2)$ and $(1, 3)$ entry of the matrix. It is important to note that if there are q rows identically zero, there are $n - q - 1$ leaders in the given formation and one lead robot. Thus, there are 4 leaders and one lead robot in the above formation since the sixth row is identically zero. Associated with digraphs, is also the concept of connectedness, which plays an important role in our formation transition algorithm. A digraph G, is *strongly connected* if there exists at least one directed path from every vertex to every other vertex in the graph. It is *weakly connected* if the corresponding undirected graph is connected but G is not strongly connected. A digraph that is neither strongly nor weakly connected is *disconnected*.

There are two constraints for formation changes. The first constraint is there can be no disconnected digraphs during the transition from one graph to another. This requires every vertex in the control graph has at least one incoming edge for $l - \psi$ control and at most two incoming edges for $l - l$ control. A disconnected graph requires more than one lead robot. While this may be desirable, it effectively breaks the team up into sub-teams and is beyond the scope and assumptions of this work. Secondly, we require a numbering system for robots to facilitate the construction of the adjacency matrix. Formally, these two constraints on allowable control graphs can be written as:

Constraint A: Every vertex $i > 1$, in the digraph has at least one incoming edge (for $l - \psi$ control) and at most two incoming edges (for $l - l$ control). This implies that there is no isolated robot in any control graph.

Constraint B: Every directed edge in the digraph goes from a lower vertex label to a higher vertex label. This implies that the adjacency matrix is always upper-triangular.

While these two constraints impose restrictions on the robot controllers, they do not restrict the relative positions of robots in a given formation. Depending on the types of sensors used, it may be difficult for one robot to follow one or more leaders when the leaders are far away from the follower. In fact, when the sensors are line-of-sight sensors (for example, optical or acoustic sensors), the restrictions are more severe. The follower must be able to "see" the leader in the formation. When the formation changes, and this includes changes in the control graph and the shape of the formation, it may be necessary to guarantee the "line of sight arrangement" between the leader and its follower(s). In fact, in this chapter, the examples in Section 6.5.3 will illustrate this point. The target formations and the strategies used to change formations are carefully selected not to violate such ***sensor constraints***. In general, these sensor constraints may be quite complex. In this thesis, we do not use any sensor constraints to limit our modeling of control graphs primarily because sensor constraints are very specific to applications.

In the following sections, we will: (a) describe the methodology used to transition from one formation to another without having disconnected digraphs in the transition process; and (b) prove the expression for an upper-bound on the total number of allowable control graphs given n robots using the technique of mathematical induction; and (c) use *Polya's enumeration theorem* to obtain a polynomial for digraphs which enumerates the equivalence classes of digraphs with a given number of directed edges.

6.4.2 Transition matrix

The transition from one control graph to another is modeled by a ***transition matrix***. The transition matrix, T, is the difference between the final and the initial adjacency matrices. Let us consider the following example in Figure 6.5. The adjacency matrix for the initial

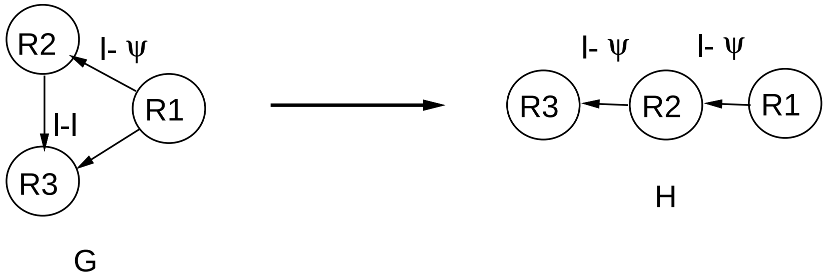

Figure 6.5: Change in formation.

and final control graphs, *i.e.*, G and H respectively, is given by:

$$G = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad H = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \tag{6.11}$$

The appearance of a 1 in a column for a robot immediately defines its controller.

$$\sum_{column} 1's = \begin{cases} 0 & \text{Lead robot} \\ 1 & \text{follower robot with} \quad l-\psi \quad \text{control} \\ 2 & \text{follower robot with} \quad l-l \quad \text{control} \end{cases}$$

Thus, $T = H - G$, is given by:

$$T = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{6.12}$$

The appearance of (-1) in the $(1,3)$ entry of the matrix denotes that the edge connecting the vertices 1 and 3 needs to be broken to achieve the transition. The formation of a new edge (not present in this example) is signalled by the appearance of a $(+1)$ in the transition matrix.

6.4.3 Enumeration of graphs

In this section, we will focus on proving the upperbound for the total number of allowable control graphs for a given number of vertices (robots).

Theorem 6.4.1 *Given n vertices in a digraph and constraints A and B, there are at most $N(n) = \frac{n!(n-1)!}{2^{n-1}}$ distinct control graphs.*

Proof: We prove the theorem by mathematical induction. For $n = 1$, it is clear that there is just one possible digraph and that is a single vertex. For $n = 2$, there is again only one possible digraph with a directed edge from 1 to 2. Thus the formula is true for $n = 1, 2$.

Assume the formula is true for some $n = k$, *i.e.*, the total number of distinct digraphs is, $N(k) = \frac{k!(k-1)!}{2^{k-1}}$. It is necessary to prove that it is also true for $n = k + 1$, *i.e.*, the total number of distinct digraphs is, $N(k + 1) = \frac{k!(k+1)!}{2^k}$.

Consider the $(k + 1)^{th}$ vertex. Based on constraints A and B, if vertex $(k + 1)$ has one incoming edge, there are $\binom{k}{1}$ possible vertices from which the incoming edge could be incident. If it has two incoming edges, there are $\binom{k}{2}$ possible choices of vertices from which the incoming edges could be incident. Thus there are totally, $\binom{k}{1} + \binom{k}{2} = \frac{k(k+1)}{2}$ possibilities of incoming edges for vertex $(k+1)$. Thus the total number of digraphs for $(k+1)$ vertices based on constraints A and B is:

$$N(k+1) = N(k) \times \frac{k(k+1)}{2} = \frac{k!(k+1)!}{2^k}$$

Thus we have proved the theorem for $n = k+1$. Since k is arbitrary, the above formula is true for all n. ■

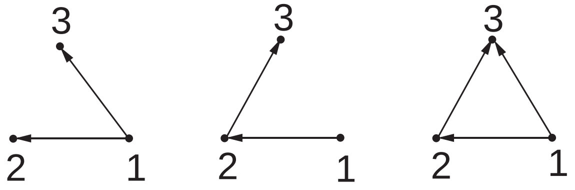

Figure 6.6: Control graphs for $n = 3$.

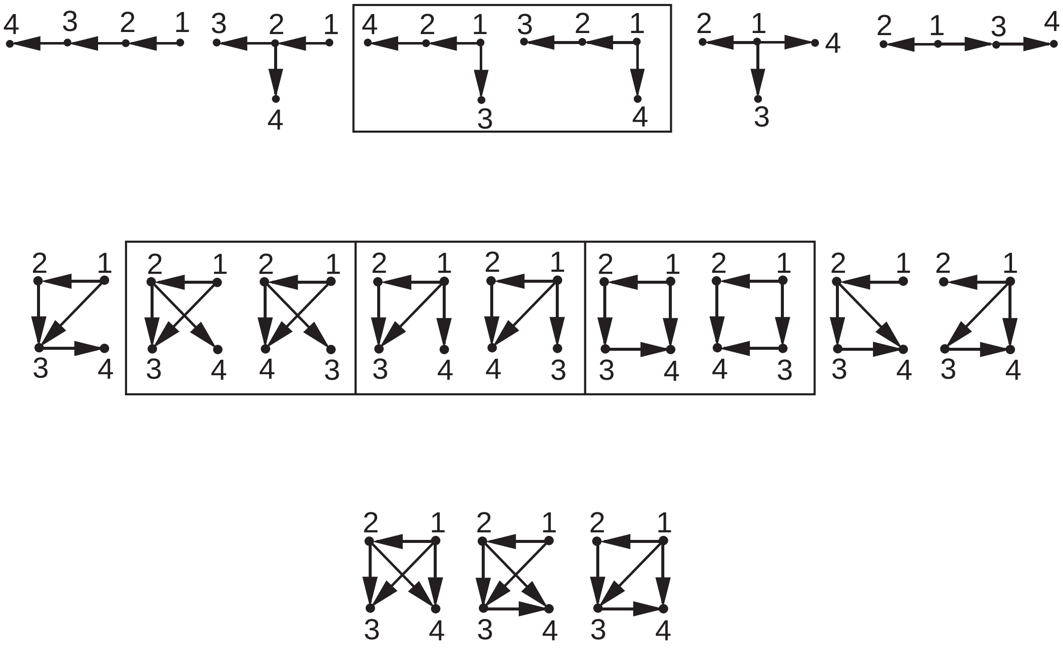

Figure 6.7: Control graphs for $n = 4$.

As an illustration of constraints A and B and the above theorem, Figures 6.6 and 6.7 show all allowable control graphs for $n = 3$ and $n = 4$ respectively. Clearly there are 3 allowable control graphs for $n = 3$ and 18 allowable control graphs for $n = 4$ as seen from the figure. We will return to describe why certain digraphs in Figure 6.7 are boxed.

We will now prove a corollary to the above theorem when all the robots in the team have $l - \psi$ control, except the lead robot. The theorem below gives an upperbound on all allowable *trees*[1] in a team of n robots.

Corollary 6.4.2 *Given n vertices in a digraph and constraints A and B, there are at most $(n-1)!$ distinct trees.*

Proof: We prove the theorem by mathematical induction. For $n = 1$, it is clear that there is just one possible tree and that is a single vertex. For $n = 2$, there is again only one possible tree with a directed edge from 1 to 2. Thus the formula is true for $n = 1, 2$.

Assume the formula is true for some $n = k$, *i.e.*, the total number of distinct trees is, $(k-1)!$. It is necessary to prove that it is also true for $n = k+1$, *i.e.*, the total number of distinct trees is, $k!$.

Consider the $(k+1)^{th}$ vertex. Based on constraints A and B, vertex $(k+1)$ has k possible vertices from which the incoming edge could be incident. Thus the total number of distinct trees is $(k-1)!\binom{k}{1} = k!$.

Thus we have proved the theorem for $n = k+1$. Since k is arbitrary, the above formula is true for all n. ■

Though Theorem 6.4.1 gives us an upperbound on the allowable control graphs, and Corollary 6.4.2 gives us an upperbound on the total number of trees in a formation, they are not particularly useful for enumerating or classifying them.

In the present problem, due to constraint B all allowable control graphs have only one directed edge from vertex 1 to vertex 2. Thus robot R_2 has $l - \psi$ control with robot R_1 in all allowable control graphs. We will now develop the necessary tools for enumerating digraphs which in conjunction with our theorem and the proposed algorithm will satisfy constraints A and B. However, the control graphs that we obtain will have a unique vertex labeled R_1 for the lead robot and there could be multiple choices for R_2 (in the case of an allowable control graph being a *tree*). Similarly the labeling on the other vertices will not be unique and they can be numbered based on constraint B. Thus in Figure 6.7, the boxed control graphs refer to two allowable control graphs each involving a permutation on the two vertices with labels 3 and 4. For a larger number of vertices (robots), though the unlabeled digraph may be isomorphic, the labeled digraphs will differ by a permutation on at most $n-1$ vertices (since R_1 is uniquely identified). During the process of enumerating the allowable control graphs, we will create a library of these control graphs, each of

[1] A tree is a connected acyclic digraph.

which can be recalled based on the constraints in the environment. We will now briefly review some mathematical preliminaries before we state Polya's enumeration theorem for digraphs.

6.4.4 Mathematical preliminaries

The study of permutation groups and graph theory are closely related to each other as graphs provide a pictorial representation of its automorphism group. Consider a set, $X = \{1, 2, \ldots, n\}$ and let G be the set of all permutations of X. This collection of elements of G forms a group and is called the *symmetric group*, S_n. The order of S_n is denoted by $o(S_n)$ and is equal to $n!$ as it represents all possible permutations of n objects. The degree of a permutation is the total number of elements in the set X, *i.e.*, n on which an element of S_n acts.

A permutation $\binom{a\ b\ c\ d}{b\ c\ a\ d}$ can be represented compactly as $(a\ b\ c)(d)$ to denote the same permutation as above where $a \to b,\ b \to c,\ c \to a,\ d \to d$. Such a representation of a permutation is called the *cycle representation of a permutation*. It is well known that every permutation can be written as a product of disjoint cycles [54][2]. Thus, if a permutation $\gamma \in G$, acts on the set X, let $\sigma_k(\gamma)$ denote the number of cycles of length k in the disjoint cycle decomposition of γ. The *cycle index* denoted by $Z(G)$, is defined as a polynomial using variables $s_1, s_2, \ldots, s_n$. It is given by:

$$Z(G) = \frac{1}{o(G)} \sum_{\gamma \in G} \prod_{k=1}^{n} s_k^{\sigma_k(\gamma)} \tag{6.13}$$

To illustrate Equation (6.13), let $G = S_3$, the symmetric group of degree 3. Before we go any further to compute the cycle index of S_3, we will describe the elements of S_3 which consist of functions mapping elements of a set $X = \{1, 2, 3\}$ of three elements to itself. Since the total number of permutations of n elements is $n!$, there are 3! possible functions which will map elements of X to itself uniquely. These functions are as stated below:

$$\begin{array}{rlrlrl} e: & \{1,2,3\} \to \{1,2,3\} & \phi: & \{1,2,3\} \to \{2,1,3\} & \psi: & \{1,2,3\} \to \{2,3,1\} \\ \psi \circ \psi: & \{1,2,3\} \to \{3,1,2\} & \phi \circ \psi: & \{1,2,3\} \to \{1,3,2\} & \psi \circ \phi: & \{1,2,3\} \to \{3,2,1\} \end{array}$$

where e is the identity map and ϕ and ψ are called the *generators* of S_3. Clearly $\psi^2 \circ \phi = \phi \circ \psi$ and $\phi \circ \phi = \psi \circ \psi \circ \psi = e$. Any of the above permutations of S_3 can be written as a product of disjoint cycles. Thus for example, the map ϕ can be written compactly as $(12)(3)$ which

[2] Two cycles are said to be disjoint if there are no common elements among them. Thus the permutation $(12)(34)$ which maps $\{1\ 2\ 3\ 4\} \to \{2\ 1\ 4\ 3\}$, is composed of two disjoint cycles (12) and (34).

indicates the same information as the mapping for ϕ above. We define any permutation, $\gamma \in S_n$ to be of type, $(\sigma) = (\sigma_1, \sigma_2, \ldots, \sigma_n)$ if γ has σ_k cycles of length k for $k = 1, 2, \ldots, n$. Thus in S_3, the identity permutation $e = (1)(2)(3)$, has 3 cycles of length 1 and hence is of the type $(3, 0, 0)$. Similarly there are three permutations, $\phi = (12)(3)$, $\phi \circ \psi = (1)(23)$ and $\psi \circ \phi = (2)(13)$, each having one cycle of length 1 and one cycle of length 2, *i.e.*, each permutation is of the type $(1, 1, 0)$. Finally, there are two permutations, $\psi = (123)$ and $\psi \circ \psi = (132)$ of length 3 and are of the type $(0, 0, 1)$. Thus, the cycle index for S_3, using Equation (6.13) is given by:

$$Z(S_3) = \frac{1}{3!}(s_1^3 + 3s_1 s_2 + 2s_3) \tag{6.14}$$

Notice that the sum of the coefficients of the polynomial comes to $1 + 2 + 3 = 6 = 3!$, the number of permutations of three elements. The general expression for $Z(S_n)$ can be derived based on various partitions of n. With the above notation, each permutation γ of n objects can be associated with a partition of n which has $\sigma_k(\gamma)$ disjoint cycles of length k for each k from 1 through n. For a given permutation γ, the partition of n can be written as a vector $(\sigma) = (\sigma_1, \sigma_2, \ldots, \sigma_n)$ where σ_k is the number of parts of length k. Each such permutation must satisfy:

$$n = 1\sigma_1 + 2\sigma_2 + \cdots + n\sigma_n = \sum_{k=1}^{n} k\sigma_k \tag{6.15}$$

Based on the above notation, if we denote $h(\sigma)$ to be the number of permutations in S_n whose cycle decomposition gives the partition (σ), then it can be shown that [52]:

$$h(\sigma) = \frac{n!}{\prod_{k=1}^{n} k^{\sigma_k} \sigma_k!} \tag{6.16}$$

Thus in the case of S_3, the identity permutation has $(\sigma) = (3, 0, 0)$ which gives $h(\sigma) = 1$ while $\sigma = (1, 1, 0)$ gives $h(\sigma) = 3$. The cycle index for a symmetric group of order n can then be shown to be [52]:

$$Z(S_n) = \frac{1}{n!} \sum_{(\sigma)} h(\sigma) \prod_{k=1}^{n} s_k^{\sigma_k} \tag{6.17}$$

where the summation is over all partitions, (σ) of n and $h(\sigma)$ is as given by Equation (6.16).

We now define the *power group*. Let G_1 be a permutation group with object set $X = \{1, 2, \cdots, n\}$ and G_2 be a permutation group with object set $Y = \{1, 2, \cdots, m\}$ of at least two elements. Since each of the n elements of X can be mapped into any one of the m elements of Y, the total number of different functions from X into Y is m^n. These collection of m^n functions is defined as the object set Y^X, for the power group $G_2^{G_1}$. The permutations of

$G_2^{G_1}$ consist of all ordered pairs $(\gamma;\delta)$, of permutations $\gamma \in G_1$ and $\delta \in G_2$. The image of any function f in Y^X under $(\gamma;\delta)$ is given by:

$$((\gamma;\delta)f)(x) = \delta(f(\gamma x)) \qquad \forall \ x \in X$$

In order to state Polya's enumeration theorem, we will take $G_2 = I$, the identity group on Y. Thus the power group I^{G_1}, acts on Y^X.

Next, let $w : Y \to \{0, 1, 2, \cdots\}$ be a function whose range is the set of all nonnegative integers. For each k, we define

$$c_k = |w^{-1}(k)|$$

to be the number of elements of Y with weight k. Then the elements $y \in Y$ which have $w(y) = k$ are said to have weight k and w is called the *weight function*.[3] In Polya's terminology, f from X to Y are called "configurations" and the elements y of Y are called "figures". The weights of the elements of Y can be expressed as powers of some common quantity x. In that case, the weight assignment to the elements of Y can be described by the "figure counting series" $c(x)$, and is given by:

$$c(x) = \sum_{k=0}^{\infty} c_k x^k \tag{6.18}$$

where c_k is the number of elements in the set Y with weight k. In other words, given a polynomial of the form

$$c(x) = 3 + 4x + 5x^2 + \cdots$$

implies that Y has three elements with weight 0, four elements with weight 1, and so on. Notice that the sum of the coefficients in the polynomial should give the total number of elements in the set Y.

Now the *weight of the function* f in Y^X is defined as:

$$w(f) = \sum_{x \in X} w(f(x))$$

A relation, $\sim$, is said to be an equivalence relation if it is reflexive, symmetric and transitive. An equivalence class is a collection of objects which are equivalent. Thus in the context of functions f in Y^X, we define two functions to be equivalent if they have the same weight *i.e.*, $f \sim g$ if $w(f) = w(g)$. These functions form an equivalence class. An *orbit* is defined as the collection of elements in the same equivalence class. Thus, functions with different weights fall into different orbits. Since pre-image under w of any non-negative

[3] In general, the weight $w(y)$ could be a symbol or a real number.

integer k is finite, there are only a finite number of orbits for each k. If we denote C_k to be the total number of elements in each orbit whose functions have weight k, the series in the indeterminate x is called the "configuration counting series" and is given by:

$$C(x) = \sum_{k=0}^{\infty} C_k x^k$$

Thus, the configuration counting series gives the expression for the total number of different configurations. For example, in the case of enumerating graphs, the polynomial $C(x) = 1 + 2x + 3x^2$ gives an expression of possible configurations of graphs with: a) no edges - only one graph is possible since the coefficient of x^0 is 1, b) one edge - two graphs are possible since the coefficient of x^1 is 2 and c) two edges - three graphs are possible since the coefficient of x^2 is 3. The main essence of Polya's enumeration theorem is that it counts the number of equivalence classes with various weights, given X, Y, permutation group G_1 on X, and the weights $w(y)$ for each $y \in Y$.

Theorem 6.4.3 (Polya's enumeration theorem) *[52] The configuration counting series $C(x)$ is obtained by substituting the figure counting series $c(x^i)$ for each s_i in the cycle index $Z(G_1; s_1, s_2, \cdots, s_n)$ of the permutation group G_1. That is,*

$$C(x) = Z(G_1; \sum c_k x^k, \sum c_k x^{2k}, \sum c_k x^{3k}, \cdots).$$

We will now illustrate Polya's enumeration theorem with the aid of a simple example. In this example, we will enumerate all possible "necklaces" that can be made by placing beads on the vertices of a triangle. Suppose we are given two colors, white and black, for the beads. Then all possible necklaces are shown in Figure 6.8. We now show that the configurations in Figure 6.8 are indeed the only possible configurations. We define a set $X = \{1, 2, 3\}$ which contains three beads with unassigned colors to them. If we want to obtain the total number of unlabeled necklaces, we must identify labeled necklaces which differ from each other by a rotation about the centroid of the equilateral triangle or reflection about the line passing through a vertex and the midpoint of the opposite edge. The set, $Y = \{W, B\}$ consists of two elements "white" and "black" which denotes the color of a bead in X. A function f from X to Y corresponds to a labeled necklace in which bead number k in X has "color" $f(k)$. Thus, the necklace represented by f has $|f^{-1}(W)|$ beads of white color and $|f^{-1}(B)|$ beads of black color. I is the identity group acting on Y and to remove the labels on the beads we identify two necklaces to be equivalent whenever they are in the same orbit of the power group I^{D_3}, where D_3 is called the *dihedral group* of degree 3. The dihedral group, D_n is the group of all symmetries of the regular polygon.

Mathematically, a dihedral group D_n, of degree n, is generated by elements r and d such that $r^n = e$ and $dr = rd^{-1}$. The order of the dihedral group is $o(D_n) = 2n$.

If we define the weights $w(W) = 0$ and $w(B) = 1$, then $c(x) = 1+x$ is the figure counting series for Y and a function of weight k represents a necklace with k black beads and $3 - k$ white beads. Thus the configuration counting series $C(x)$ enumerates the unlabeled necklaces and the coefficient of x^k is the number of necklaces with k black beads. By Polya's enumeration theorem:

$$C(x) = Z(D_3, 1 + x)$$

Since $Z(D_3)$ is the same as $Z(S_3)$ (D_3 is isomorphic to S_3), the configuration counting series is obtained by substituting $(1+x)$ for s_1, $(1+x^2)$ for s_2 and $(1+x^3)$ for s_3 in Equation (6.14). After some simplifications, we get:

$$C(x) = 1 + x + x^2 + x^3 \tag{6.19}$$

indicating that there is one necklace with all white beads ($1 \cdot x^0$), one necklace with one black and two white beads ($1 \cdot x^1$) and so on. The expression in Equation (6.19) is also easily verified from Figure 6.8 as we scan from left to right.

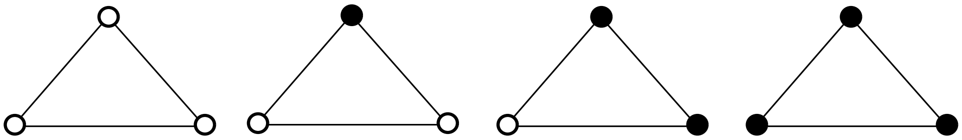

Figure 6.8: The necklace problem.

Since our ultimate goal is to obtain an expression for the equivalence classes of figures for a digraph, it is important to understand the principle of G-equivalence classes of r-sets. We define two subsets $P_1 = \{x_1, x_2, \ldots, x_r\}$ and $P_2 = \{x_1^{'}, x_2^{'}, \ldots, x_r^{'}\}$ of $X = \{1, 2, \ldots, n\}$ to be G-equivalent if there exists a permutation $\gamma \in G$ such that, $\gamma P_1 = P_2$. We will call P_1 and P_2 as r-sets of X since they contain r elements. We now state and prove the corollary that gives the total number of $G - equivalence$ classes of r-sets of X.

Corollary 6.4.4 *[52] The coefficient of x^r in $Z(G, 1 + x)$ is the number of G-equivalence classes of r-sets of X.*

Proof: The figure counting series $c(x) = 1 + x$ can be derived based on the assumption that $1 = x^0$ indicates the absence of an object in X while $x = x^1$ indicates its presence. Clearly x^r indicates the presence of r distinct objects of X thus forming an r-set P_1 or P_2 as defined above. Now based on Polya's enumeration theorem the coefficient of x^r indicates the number of G-equivalence classes of r-sets of X. ■

We now state the generalization of Polya's enumeration theorem for N variables:

Theorem 6.4.5 (Polya's enumeration theorem for N variables) *[52]*
If $c(x_1, x_2, \ldots, x_N)$ is the figure counting series for Y, then the orbits of functions in Y^X determined by the power group I^{G_1} are enumerated by weight with $C(x_1, x_2, \ldots, x_N)$ and

$$C(x_1, x_2, \ldots, x_N) = Z(G_1; c(x_1, x_2, \ldots, x_N), c(x_1^2, x_2^2, \ldots, x_N^2), \cdots).$$

Before we proceed to state the theorem of Polya for enumerating directed graphs of order n, we need to introduce some definitions. The number of edges incident out of a vertex v_i, is called the *out-degree* of v_i and is denoted by $d^+(v_i)$. The number of edges incident into v_i, is called the *in-degree* of v_i and is denoted by $d^-(v_i)$. Thus, the out-degree of v_1 (R_1) in digraph G of Figure 6.5 is $d^+(v_1) = 2$, while its in-degree is $d^-(v_1) = 0$. Similarly, the out-degree of v_3 (R_3) is $d^+(v_3) = 0$, while its in-degree is $d^-(v_3) = 2$.

Let $X = \{1, 2, \ldots, n\}$ and let $X^{[2]}$ denote all ordered pairs of different elements of X. Let $Y = \{0, 1\}$ and the functions, $f : X^{[2]} \to Y$ represent labeled digraphs of order n. Based on the function f, we denote $D(f)$ to be the corresponding digraph with a directed edge from vertex i to vertex j if and only if $f(i, j) = 1$. Thus the presence or absence of a directed edge between ordered vertices is obtained by their function value 0 or 1. The figure counting series is thus $(1 + x)$ as the arc connecting two given vertex pairs (v_i, v_j) is either present or absent. Two functions f and g represent the same digraph if there exists a permutation $\gamma \in G$ such that, if i is adjacent to j in $D(f)$ then γi is adjacent to γj in $D(g)$. Therefore $D(f)$ and $D(g)$ are isomorphic if and only if

$$f(i, j) = g(\gamma i, \gamma j) \quad \forall \ (i, j) \in X^{[2]}, \qquad \text{for some} \quad \gamma \in G \tag{6.20}$$

We next introduce the concept of the *reduced ordered pair group* of G denoted by $G^{[2]}$. The reduced order pair group of a permutation group G with object set X is defined as a group of permutations induced by permutations in G such that for every ordered pair (i, j) in $X^{[2]}$ the image under $\gamma' \in G^{[2]}$ is given by:

$$\gamma'(i, j) = (\gamma i, \gamma j)$$

The object set of $G^{[2]}$ is $X^{[2]}$. Since there are at most $2 \times \binom{n}{2}$ possible orderings in $X^{[2]}$, the degree of $G^{[2]}$ is $n(n-1)$. For example, in the case of allowable control graphs with n robots, the set $X^{[2]}$ denotes the pairs of vertices connected by a directed edge and $G^{[2]}$ is simply $S_n^{[2]}$. Corollary 6.4.4 gives the total number of formations with r directed edges in a control graph.

As an example of the induced term in the cycle index of $Z(S_3^{[2]})$, consider the permutation $\gamma = (1)(23)$ of S_3. The permutation

$$\gamma = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 3 & 2 \end{pmatrix}$$

induces the following permutation on the six ordered vertex pairs

$$\gamma^{'} = \begin{pmatrix} 12 & 13 & 23 & 21 & 31 & 32 \\ 13 & 12 & 32 & 31 & 21 & 23 \end{pmatrix}$$

Thus, $\gamma = (1)(23) \in S_3$ induces the permutation $\gamma^{'} = (12\ 13)(23\ 32)(21\ 31) \in S_3^{[2]}$ in cycle notation. Based on the permutation γ, the term $s_1 s_2$ in the cycle index $Z(S_3)$ induces the term s_2^3 in the cycle index $Z(S_3^{[2]})$ of the reduced ordered pair group. Similarly, if we choose other permutations belonging to the group S_3, we can derive the other induced terms in the cycle index of $Z(S_3^{[2]})$. Following the calculations for S_3, we obtain the following cycle index for the reduced ordered pair group $S_3^{[2]}$:

$$Z(S_3^{[2]}) = \frac{1}{3!}\left(s_1^6 + 3s_2^3 + 2s_3^2\right) \tag{6.21}$$

Similar computation for S_4 gives the following expression for the cycle index for the reduced ordered pair group, $Z(S_4^{[2]})$:

$$Z(S_4^{[2]}) = \frac{1}{4!}\left(s_1^{12} + 6s_1^2 s_2^5 + 8s_3^4 + 3s_2^6 + 6s_4^3\right) \tag{6.22}$$

Each permutation of S_4 induces a term in the reduced ordered pair group and is given in Table 6.1 along with the total number of permutations of S_4 which give the following partition $(\sigma) = (4,0,0,0),\ (2,1,0,0),\ (1,0,1,0),\ (0,2,0,0),\ (0,0,0,1)$ for $n = 4$ (refer to Equation (6.15)). The total number of permutations $h(\sigma)$, with a particular partition (σ), is computed using Equation (6.16).

Table 6.1: Cycle structure of permutations in S_4 and the corresponding induced term in $S_4^{[2]}$.

Term in $Z(S_4)$	Induced term of $Z(S_4^{[2]})$	No. of permutations ($h(\sigma)$)
s_1^4	s_1^{12}	1
$s_1^2 s_2$	$s_1^2 s_2^5$	6
$s_1 s_3$	s_3^4	8
s_2^2	s_2^6	3
s_4	s_4^3	6

We will now proceed to state the theorem for the total number of digraphs of order n using the modified form of Polya's enumeration theorem. The following theorem enumerates digraphs $d_n(x)$, of order n.

Theorem 6.4.6 (Polya's enumeration theorem for digraphs) *[52]*
The counting polynomial $d_n(x)$ for digraphs of order n is given by

$$d_n(x) = Z(S_n^{[2]}, 1+x) = \sum_{k=1}^{n(n-1)} a_k x^k$$

where

$$Z(S_n^{[2]}) = \frac{1}{n!}\sum_{(\sigma)} \frac{n!}{\prod_{k=1}^{n} k^{\sigma_k}\sigma_k!} \prod_{k=1}^{n} s_k^{(k-1)\sigma_k+2k\binom{\sigma_k}{2}} \prod_{r<t} s_{\lfloor r,t\rfloor}^{2\lceil r,t\rceil\sigma_r\sigma_t}$$

and $\lfloor r,t\rfloor$ and $\lceil r,t\rceil$ denote the least common multiple (l.c.m) and the greatest common divisor (g.c.d) of r and t respectively.

The upper limit in the summation for $d_n(x)$ in the above theorem is $n(n-1)$ since the set $X^{[2]}$, of all ordered pairs of elements of X has $n(n-1)$ elements. Now since any term in the cycle index of the reduced ordered pair group $G^{[2]}$, consists of the polynomial in $s_k^{\sigma_k}$ such that $\sum k\sigma_k = n(n-1)$, we conclude that the highest exponent of x in $d_n(x)$ is $n(n-1)$. As an example, the expression enumerating digraphs of order $n = 3$, is given by the polynomial $d_3(x)$:

$$d_3(x) = 1 + x + 4x^2 + 4x^3 + 4x^4 + x^5 + x^6$$

Summing the coefficients of the various powers of x in $d_3(x)$ gives the total number of digraphs of order 3 (*i.e.*, $d_3(1) = 16$). The term $4x^2$ in the above expression for $d_3(x)$, suggests that there are 4 possible digraphs with 2 directed edges. These digraphs belong to the same equivalence class. These are shown in Figure 6.9. It can be easily verified

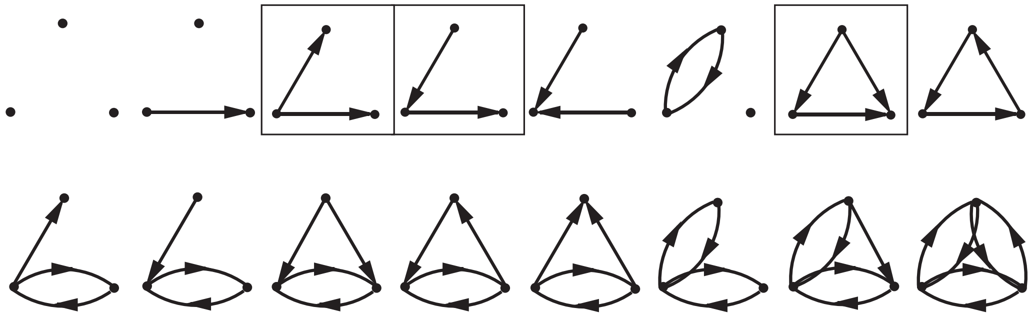

Figure 6.9: All unlabeled digraphs for $n = 3$.

from Figure 6.9 that the above formula is correct when we scan the figure from the top row, left to right. The boxed digraphs in Figure 6.9 are the required control graphs and we will derive the algorithm to obtain these digraphs from all possible digraphs later in this section. For $n = 4$, the expression for $d_4(x)$ is given by:

$$d_4(x) = 1 + x + 5x^2 + 13x^3 + 27x^4 + 38x^5 + 48x^6 + 38x^7 + 27x^8 + 13x^9 + 5x^{10} + x^{11} + x^{12} \quad (6.23)$$

Summing the coefficients for $d_4(x)$ or evaluating $d_4(1)$ gives 218 possible unlabeled digraphs of order 4. Since the present problem has only one lead robot (R_1), every vertex except v_1, has at least one and at most two incoming edges (constraint A) which greatly simplifies the enumeration task. The next theorem gives an upper-bound on the allowable unlabeled control graphs for arbitrary order n.

Theorem 6.4.7 *An upper-bound on the allowable unlabeled control graphs for n robots is given by the polynomial:*

$$R_n(x) = \sum_{k=(n-1)}^{(2n-3)} a_k x^k$$

There are $(n-1)$ equivalence classes of digraphs, where the k^{th} equivalence class has a_k members, and a_k is given by Theorem 6.4.6.

Proof: We will first prove that the lower bound on the summation is indeed $(n-1)$. Since there is at least one directed edge incident on any vertex (except one vertex which corresponds to the lead robot) based on constraint A, the lower bound on the total number of directed edges in the control graph to make it at least weakly connected is $(n-1)$. Thus the lower bound on the required polynomial is $(n-1)$. Similarly, since there can be at most 2 directed edges incident on any vertex, except two vertices v_1 and v_2, [4] the upper-bound on the summation is $2(n-2)+1 = (2n-3)$.

Since k varies from $(n-1)$ through $(2n-3)$, there are $(n-1)$ equivalence classes and the coefficient of the polynomial x^k is a_k, which denotes the members in the equivalence class with k directed edges.

The expression for $R_n(x)$ can be extracted from the expression for $d_n(x)$ stated in Theorem 6.4.6 based on the lower and upper bounds on k. ■

Thus $R_n(1)$ gives us an upper-bound on unlabeled allowable control graphs for any given n. To illustrate the theorem for $n = 3$, we get:

$$R_3(x) = 4x^2 + 4x^3$$

[4] The vertex corresponding to the lead robot (v_1) has in-degree, $d^-(v_1) = 0$ and the other vertex labeled v_2 has in-degree, $d^-(v_2) = 1$ based on constraints A and B.

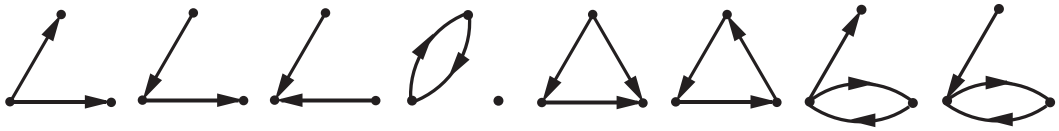

Figure 6.10: The two equivalence classes of digraphs for $n = 3$ enumerated by Theorem 6.4.7.

For example, the coefficient 4 of x^2 implies that there are four unlabeled digraphs in the equivalence class with 2 directed edges as shown in Figure 6.10. It is clear that the task of enumerating all digraphs of order three given by Polya's enumeration theorem 6.4.6 is reduced to $(n-1)$ equivalence classes and the search space for allowable control graphs is far less than before. In the case of $n = 3$, we can narrow our search for allowable control graphs to only eight ($a_2 + a_3 = 4 + 4 = 8$) as shown in Figure 6.10 compared to sixteen ($d_3(1) = 16$) enumerated in Figure 6.9. Similarly in the case for $n = 4$ from 218 ($d_4(1) = 218$) possible digraphs we need to search only 78 ($a_3+a_4+a_5 = 13+27+38 = 78$). The three control graphs boxed in Figure 6.9 are the allowable control graphs since there is only one vertex with in-degree zero and all other vertices have at least one incoming edge (*i.e.*, their in-degree is at least one).

Before we state the algorithm to obtain allowable control graphs from Theorem 6.4.7, we need to define a few more terms. A *walk* of length q in a digraph is determined by the sequence of points $v_1, v_2, \ldots, v_q$ in which v_i is adjacent to v_{i+1} for $i < q$. A *closed walk* has the same first and last point, *i.e.*, $v_1 = v_q$. A *cycle* is a non-trivial closed walk with all points distinct except the first and the last vertex. A digraph is said to be *acyclic* if it has no cycles. We will now state some theorems for acyclic digraphs as they are central to extracting the allowable control graphs enumerated by Theorem 6.4.7.

Theorem 6.4.8 *[28] Given the $n \times n$ adjacency matrix X of a digraph G of order n, the digraph G is acyclic if and only if Det($I - X$) is not equal to zero, where I is the $n \times n$ identity matrix.*

Since any acyclic digraph has Det($I - X$) non-zero, is it guaranteed that the adjacency matrix can be reduced to an upper or a lower triangular matrix? Since constraint B in our formulation requires that the edge is incident from a lower index to a higher index, the adjacency matrix is always upper triangular. Thus, the following theorem proves that given an acyclic digraph, we can always reorder the vertices so that the adjacency matrix is upper (lower) triangular. In our case, we will not have a lower triangular matrix due to constraint B.

Theorem 6.4.9 *[28] A digraph G, is acyclic if and only if its vertices can be ordered such that the adjacency matrix is an upper (lower) triangular matrix.*

Theorem 6.4.8 is the central result for extracting allowable control graphs. We need the above theorem since constraint B requires that the edges are directed from a lower index to a higher index. If that is the case, there can be no cycles in the allowable control graphs since a cycle has the same first and last vertex. We now state the algorithm used to extract the allowable unlabeled digraphs from those enumerated by Theorem 6.4.7.

Algorithm for choosing allowable control graphs: The algorithm can be broken down into the following steps:

1. Use Polya's theorem to enumerate all digraphs of order n, *i.e.*,

$$d_n(x) = 1 + a_1 x + a_2 x^2 + \ldots + a_{n(n-1)} x^{n(n-1)}$$

2. Use Theorem 6.4.7 to obtain an upper-bound on allowable control graphs, *i.e.*, $R_n(x)$ given by:

$$R_n(x) = a_{(n-1)} x^{(n-1)} + a_n x^n + \ldots + a_{(2n-3)} x^{(2n-3)}$$

3. Choose any digraph G, enumerated by $R_n(x)$ and label the vertices arbitrarily. Based on this labeling construct the adjacency matrix X.

4. Check if there is only one vertex in G that has in-degree 0 *i.e.*, check if X has only one column with all its entries 0. If there exists a zero column, re-label the vertex corresponding to that column as vertex v_L. This is the vertex for the lead robot R_1, since our formulation allows for only one lead robot in the team of n robots. If not, goto step 3 for a new digraph..

5. If v_L exists, check if all the other vertices have in-degree of at least 1 and no vertex has in-degree greater than 2 based on constraint A, *i.e.*, $1 \leq d^-(v_i) \leq 2$ for all $i = 1, \ldots, n$ ($i \neq L$). If not, goto step 3 for a new digraph.

6. If condition (5) is satisfied evaluate Det($I - X$), where X is the adjacency matrix for G. If G is acyclic, Det($I - X$) evaluates to a non-zero value. If not, goto step 3 for a new digraph.

7. If G satisfies all the above steps, we can re-label the lead robot denoted by vertex v_L as v_1.

8. Repeat step 3 until all the digraphs enumerated by $R_n(x)$ are examined.

The above algorithm can be illustrated by the flowchart shown in Figure 6.11.

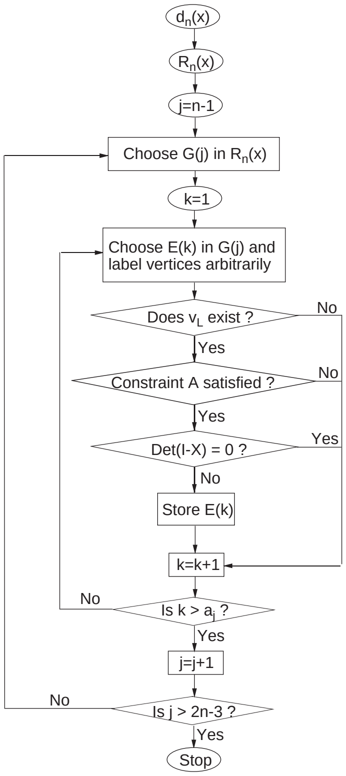

Figure 6.11: Flowchart for choosing allowable control graphs.

Corollary 6.4.10 *All the robots in the team (except the lead robot, R_1) have $l - \psi$ control if and only if the total number of directed edges in the allowable control graph obtained by the above algorithm is $(n-1)$.*

Proof: We first prove that if all the robots in the team apart from R_1 have $l-\psi$ control, then the total number of edges in the allowable control graphs obtained by the above algorithm is $(n-1)$. Since every robot apart from R_1 has $l-\psi$ control there is only one edge incident on every vertex. Since there are a total of $(n-1)$ vertices, there are $(n-2)$ edges connecting them and there is one edge incident from the lead robot R_1 to R_2 which makes the total number of edges $(n-1)$.

To prove the converse, we need to show that $(n-1)$ edges guarantee $l-\psi$ control for all the robots, except R_1. Assume that this is false. Thus, there is at least one robot with $l-l$ control, *i.e.*, 2 incoming edges $(1 \leq d^-(v_i) \leq 2$ for $i = 2, \ldots, n)$. If there is at least one robot with two incoming edges and since the digraph is at least weakly connected, the total number of edges in the control graph would be at least $(n-2)+2 = n$, where $(n-2)$ edges are required to connect $(n-1)$ remaining robots and 2 edges are incident on the robot with $l-l$ control. Clearly the sum of edges contradicts the assumption of $(n-1)$ edges. ∎

Corollary 6.4.11 *All the robots in the team (except the lead robot, R_1 and the robot with only one incoming edge say R_2) have $l-l$ control if and only if the total number of directed edges in the allowable control graphs obtained by the above algorithm is $(2n-3)$.*

Proof: We first prove that if all the robots in the team apart from R_1 and R_2 have $l-l$ control, then the total number of edges in the allowable control graphs obtained by the above algorithm is $(2n-3)$. Since every robot apart from R_1 and R_2 have $l-l$ control, there are two incident edges on every vertex (*i.e.*, $d^-(v_i) = 2$ for $i = 3, \ldots, n$). Since there are a total of $(n-2)$ vertices, there are $(2n-4)$ edges connecting them and there is one edge incident from the lead robot R_1 to R_2 which makes the total number of edges $(2n-3)$.

To prove the converse, we need to show that $(2n-3)$ vertices guarantee $l-l$ control for all the robots, except R_1 and R_2. Assume that this is false. There is only one lead robot R_1 in the formation and the digraph is at least weakly connected and no robot has more than two incoming edges by constraint A . Thus, if the converse is false, there exists at least one vertex v_k with in-degree 1 $(1 \leq d^-(v_k) \leq 2)$. Thus, the total number of edges in the allowable control graph will be at most $2(n-3) + 1(for\ \ v_k) + 1(R_1\ \ to\ \ R_2) = 2n-4$. However, there are $(2n-3)$ edges, hence no such v_k exists. ∎

The above two theorems suggest that as the exponent of x in the expression for $R_n(x)$ varies from $(n-1)$ to $(2n-3)$, there is a spectrum of control graphs ranging from pure $l-\psi$ control to pure $l-l$ control with a medley of combined $l-\psi$ and $l-l$ control falling in between the two exponents (for $n > 3$ as $n = 3$ has only two equivalence classes, pure $l-\psi$ and pure $l-l$ control). It is also important to note that the above algorithm was specialized to the case for only one lead robot due to constraint A. However, if we have multiple lead robots in the formation, we can easily adapt the above algorithm to choose those allowable formations where the only condition that changes is that of the in-degree of the vertices. Say for example in the case of two lead robots, we need to search for control graphs with only two vertices having in-degree zero and all other vertices having in-degree of at least one and at most two. Such a constraint would allow disconnected digraphs as allowable control graphs for a team of robots. Thus, the above algorithm combined with the theorems gives us a strategy to choose allowable control graphs from the vast array of control graphs enumerated by Polya's Theorem 6.4.6.

6.4.5 Mixed graphs:

A graph is said to be a mixed graph if it consists of any number (including zero) of directed and/or undirected edges. In this paragraph, we will discuss mixed graphs as a natural extension of ordinary and directed graphs. Though graphs in our problem are directed graphs, and have been discussed above, we present the following material on mixed graphs for the sake of completeness. It is clear from Figure 6.12 that there are 10 equivalence classes of graphs of order 3. The first class has no edges and 1 member, the second has 1 directed edge and 1 member, the third has 2 directed edges and 3 members and so on. In Figure 6.13, the notation m_{nqr} denotes the mixed graph with n vertices, q directed edges and r undirected edges. Figure 6.13 enumerates equivalence classes with only one directed edge. There are two equivalence classes with $m_{311} = 2$ and $m_{312} = 1$ members.

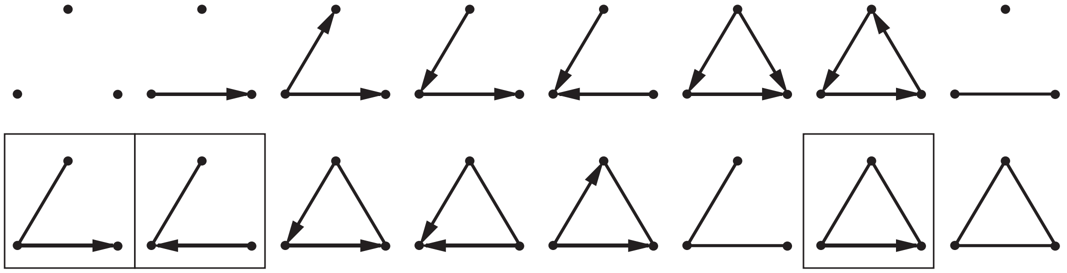

Figure 6.12: All mixed graphs of order 3.

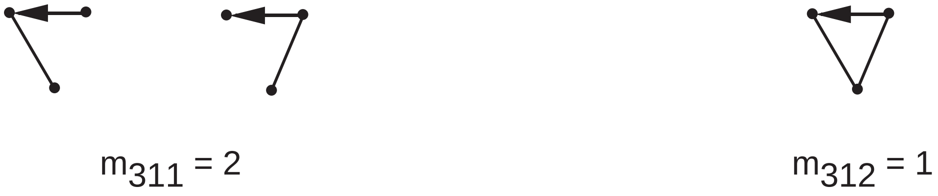

Figure 6.13: The two equivalence classes of mixed graphs with one directed edge for $n = 3$.

A similar enumeration for one directed edge for $n = 4$ yields three equivalence classes: 5 members with 2 undirected edges, 6 members with 3 undirected edges, and 3 members with 4 undirected edges as shown in Figure 6.14.

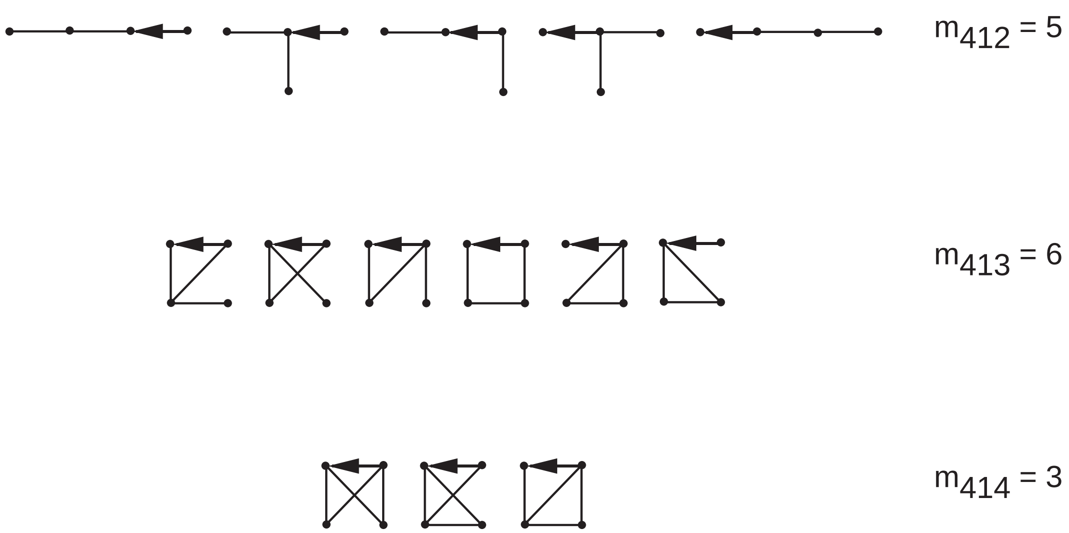

Figure 6.14: The three equivalence classes of mixed graphs with one directed edge for $n = 4$.

Before we state the theorem for the enumeration of mixed graphs, we define the notions of *converse* and *self-converse* of a cycle in the disjoint cycle decomposition of a given permutation. Let γ' be a permutation in $S_n^{[2]}$ induced by γ. The converse of any given cycle z' in the disjoint cycle decomposition of γ' is that cycle z'' of γ' which permutes all ordered pairs $(i, j) \in X^{[2]}$ such that (j, i) is permuted by z'. A cycle z' of γ' is *self-converse* if (i, j) is permuted by z' whenever (j, i) is permuted. As an example consider the permutation $\gamma = (1)(23) \in S_3$ which induces the permutation $\gamma' = (12\ 13)(23\ 32)(21\ 31)$ as discussed before. The converse of the cycle $z' = (12\ 13)$ in the disjoint cycle decomposition of γ' is $z'' = (21\ 31)$ since z'' permutes the ordered pairs $(2, 1)$ and $(3, 1)$ such that the ordered pair $(1, 2)$ and $(1, 3)$ is permuted by z'. Note that z' is not self-converse. However, if $z' = (23\ 32)$, then it is clearly self-converse as $z'' = z'$.

We will now state the theorem which enumerates mixed graphs of order n with q directed edges and r undirected edges. The figure counting polynomial which enumerates mixed graphs of order n with both directed and ordinary edges is given by:

$$m_n(x, y) = \sum_{q,r=1} m_{nqr} x^q y^r$$

where $q + r \leq \binom{n}{2}$. The theorem stated below can be derived using Polya's enumeration theorem for N variables stated in Theorem 6.4.5.

Theorem 6.4.12 (Polya's enumeration theorem for mixed graphs) *[52]*

The counting polynomial for mixed graphs of order n is given by:

$$m_n(x,y) = Z(S_n^{[2]}; (1+2x+y)^{1/2}, 1+y^{1/2}) \tag{6.24}$$

where

$$Z(S_n^{[2]}; s_k, t_k) = \frac{1}{n!}\sum_{(\sigma)} \frac{n!}{\prod_{k=1}^{n} k^{\sigma_k}\sigma_k!} \prod_{k\ odd} s_k^{(k-1)\sigma_k} \prod_{k\ even} (s_k^{(k-2)}t_k)^{\sigma_k} \prod_{k=1}^{n} s_k^{2k\binom{\sigma_k}{2}} \prod_{r<w} s_{\lfloor r,w\rfloor}^{2\lceil r,w\rceil\sigma_r\sigma_w} \tag{6.25}$$

where s_k denotes pairs of converse cycles and t_k for self-converse cycles.

We substitute $(1+2x^k+y^k)^{1/2}$ for s_k (converse cycles) in the cycle index of the reduced ordered pair group and $(1+y^{k/2})$ for t_k.

The figure counting series corresponding to converse cycles is given by $(1+2x+y)$ where 1 indicates the absence of an edge between two given vertices, $2x$ indicates two possible orientations and y an undirected line. The power of 1/2 in $(1+2x+y)^{1/2}$ vanishes in $m_n(x,y)$ since s_k always occurs with even powers as converse cycles in a given permutation necessarily appear in pairs. Similarly, the figure counting series $(1+y)$ indicates the presence or absence of an undirected edge in self-converse cycles. Also the power of 1/2 in $(1+y^{1/2})$ vanishes in $m_n(x,y)$ as the only term which contains t_k occurs in even powers of k. Since the theorem stated above is general, we can easily obtain expressions for digraphs $d_n(x)$, and ordinary graphs $g_n(x)$ of order n as shown below:

$$d_n(x) = m_n(x,x^2) \qquad g_n(y) = m_n(0,y)$$

As an example to illustrate the applicability of the above theorem, consider a mixed graph of order $n=3$. The expression for $Z(S_3^{[2]}; s_k, t_k)$ is given by:

$$Z(S_3^{[2]}; s_k, t_k) = \frac{1}{3!}(s_1^6 + 3s_2^2t_2 + 2s_3^2)$$

Now to obtain the expression for $m_3(x,y)$, we substitute $(1+2x+y)^{1/2}$ for s_1, $(1+2x^2+y^2)^{1/2}$ for s_2, $(1+2x^3+y^3)^{1/2}$ for s_3 and $(1+y^{2/2})$ for t_2. After some simplifications the formula for $m_3(x,y)$ is given by:

$$m_3(x,y) = 1+x+3x^2+2x^3+y+2xy+3x^2y+y^2+xy^2+y^3$$

It can be easily verified from Figure 6.12 that the above formula is correct when we scan the figure from the top row, left to right.

6.4.6 Formation changes

Depending on the task, the environment, and the sensor constraints, it may be necessary to change the formation of the team of robots. The choice of formation for a particular task is a challenging problem which is not addressed here. Instead, we focus on the problem of achieving a desired formation. We have addressed the enumeration of control graphs in the previous section. In this section, we explain how a transition between a given control graph and a target control graph can be achieved.

Figure 6.15 is an exhaustive list of all possible transitions in the control graph at the j^{th} vertex. The list in the figure is obtained by considering the possible switches in control laws for robot R_j ($j \neq 1$). In transitions 1, $2(a)$, $3(a)$ and $4(a)$, the robot changes its controller dependencies instantaneously on the initiation of transition. Thus if R_j is to undergo a transition of the type $2(a)$ shown in Figure 6.15, it will simultaneously break connection with R_i and make connection with R_h. The simultaneous "make and break" is possible as R_j is at no instant an isolated robot in the control graph (R_j preserves the link with the common robot, R_k, which is present in the initial and final control graph). However, transitions of the type $2(b)$, $3(b)$ and $4(b)$ have an intermediate step since the intersection set of the leaders of R_j before and after transition is a null set. Thus, for example, if R_j in $2(b)$ were to form two new connections, it will not break connections with both R_i and R_k simultaneously, but instead break it with R_i (or R_k) and then form a link with R_p (or R_h) and so on for the remaining set of robots. This procedure is enforced based on constraints A and B mentioned before.

Thus, it is clear from the flowchart in Figure 6.16 that as you scan the columns of the transition matrix sequentially from T_2 through T_n, you will encounter one of the seven possible enumerations shown in Figure 6.15 which is an exhaustive list of all possible transitions for the robot R_j. Based on the algorithm in the flowchart and the discussion above, the process of transitioning from one control graph to another is indeed complete, *i.e.*, the algorithm in Figure 6.16 will guarantee that at the end of the transition process, we will have the desired final control graph.

6.5 Simulation results

In this section, we will demonstrate the use of the proposed control laws both in the presence and absence of obstacles and show how we can transition from one formation to another based on the transitions enumerated in Figure 6.15. The concepts discussed

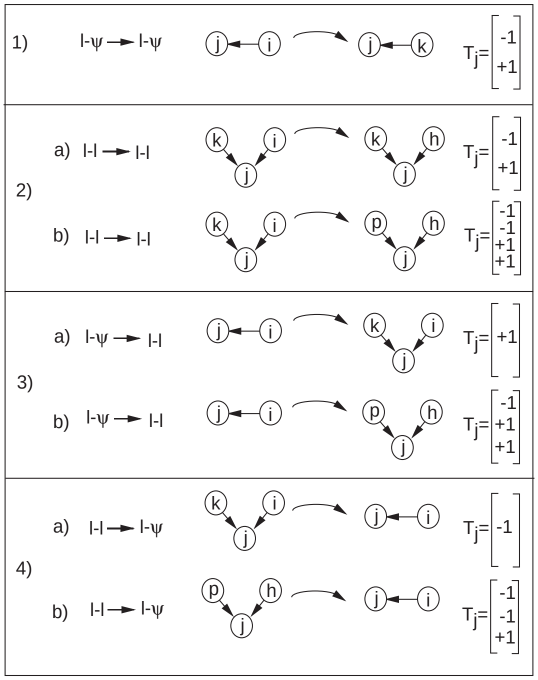

Figure 6.15: Enumeration of transitions between control laws for robot R_j. T_j refers to the j^{th} column of the transition matrix. The row numbers are either h, i, k or p, depending on the example.

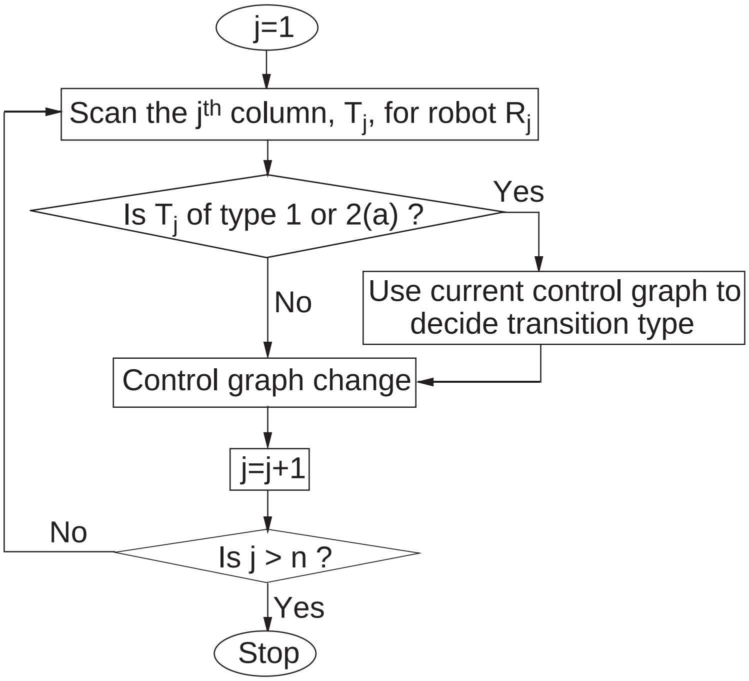
j=1
Scan the jth column, Tj, for robot Rj
Is Tj of type 1 or 2(a) ?
Yes
Use current control graph to decide transition type
No
Control graph change
j=j+1
No
Is j > n ?
Yes
Stop

Figure 6.16: Control graph transition algorithm flowchart.

above have been demonstrated through several examples in each section below. In Sections 6.5.1 and 6.5.2, the changes in formation or motion plans are such that at each instant of time, the formation (or underlying digraph) is isomorphic to the initial configuration. Hence, there is no change in the control graph at any instant of time. In Example 4, we demonstrate that it is possible to have a change in the shape of the formation while preserving the underlying control graph. This can occur since it is not always necessary to change the control graph. Finally, in Section 6.5.3, we discuss three examples of non-isomorphic graph transitions based on sensory constraints that are essentially line of sight constraints. All the simulations were done in Matlab and the constants α_1 and α_2 in Equations (6.4) and (6.9) were chosen to be 10 for all the simulations.

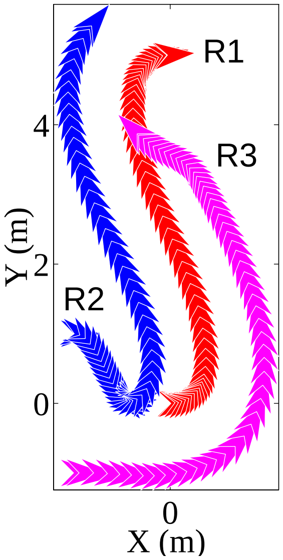

Figure 6.17: Parallel parking.

6.5.1 No obstacles

Example 1: We first consider an example of parallel parking without any obstacles in the environment. The lead robot denoted R_1 plans an optimal path as shown in Figure 6.17. In this example, we show two follower robots and they are both controlled by an $l - \psi$ control law. Using the notation of Figure 6.2, the configuration of the system of three robots is given by $[x_1, y_1, \theta_1, l_{12}, \psi_{12}, \theta_2, l_{13}, \psi_{13}, \theta_3]^T$. In this example, the lead robot's (R_1) optimal path minimizes the distance traveled in the parallel displacement of $5m$ with turning radius constraints. The optimal path is as shown in Figure 6.17. Also shown is the snapshot at various instants of time of the three robots moving in formation. The initial and final values of the shape variables describing the follower robot are $l_{12} = l_{13} = \sqrt{2}m$, $\psi_{12} = 135°$, and $\psi_{13} = -135°$. The offset d, was chosen to be $0.4m$.

Since we cannot guarantee any final orientation (θ_2 or θ_3) of R_2 and R_3, we denote their final orientation by θ_{2f} and θ_{3f}. However, based on the claim in Theorem 6.3.1, if the lead robot after reaching the final position, continues to move in a straight line, the orientation of R_2 and R_3 will exponentially stabilize to that of the lead robot, R_1. We note, however, that the shape variables $[l_{12}, \psi_{12}]^T$ and $[l_{13}, \psi_{13}]^T$ do achieve the desired values at the completion of motion. Note that the distance and relative orientation for follower robots R_2 and R_3, is measured from the center of R_1 to their tip.

6.5.2 Obstacles

In this section, we present examples for the case of 6 robots avoiding an obstacle. An obstacle free optimal path is generated by the lead robot and the path is optimal with respect to a suitable cost function. The $l - l$ control can be modified to avoid obstacles by changing for example, the $l - \psi$ control for R_2 to an $l - l$ control where R_2 maintains a desired separation from R_1 and the obstacle O_1 shown schematically in Figure 6.18. In the first two examples we illustrate how R_2 and R_4 change from an $l - \psi$ control to an $l - l$ control when they are within a certain threshold of the obstacle. The schematic of the various controllers for each robot is as shown in Figure 6.18.

Example 2: In this example, R_1 executes a straight line path from $(0, 0, 0)$ to $(10, 0, 0)$. As proved in Section 6.3, we expect the orientations of the follower robots to exponentially stabilize to that of the lead robot, R_1, after they overcome the obstacle. This is observed in the plot shown in Figure 6.19. Initially, when the lead robot follows a straight line, the follower robots, R_2 and R_4 are controlled by the $l - \psi$ feedback law while the other robots are controlled by the $l - l$ feedback law as shown in Figure 6.18. However, when R_2 and R_4 move within a predefined distance of the obstacle, they switch from the $l - \psi$ controller to an $l - l$ controller, where one of the distances is the distance to the obstacle. This guarantees that the robots are a predefined distance away from the obstacle. After the obstacle is overcome and the distance of R_2 and R_4 exceeds the threshold, it switches back to the $l - \psi$ controller. This is observed in the paths shown in Figure 6.19. The sensor constraints in this example are line-of sight constraints and requires that:

- There is no robot/obstacle between R_1 and R_2.
- There is no robot/obstacle between R_1, R_2 and R_3.

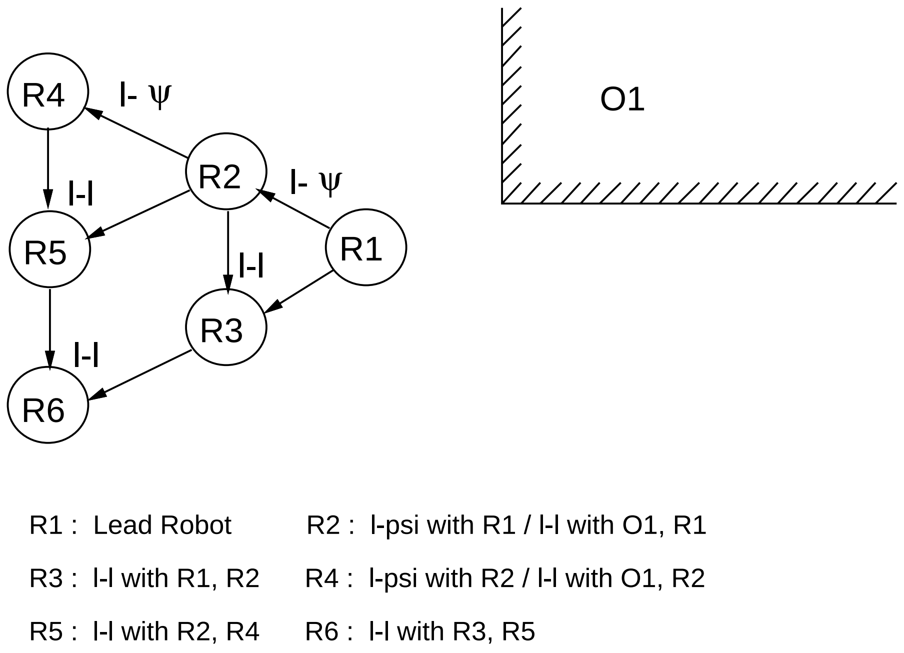

Figure 6.18: Controller schematic.

- There is no robot/obstacle between R_2 and R_4.
- There is no robot/obstacle between R_2, R_4 and R_5.
- There is no robot/obstacle between R_3, R_5 and R_6.

Thus there are 8 sensor constraints corresponding to the eight edges in the digraph. Although it is difficult to see from the figure, all the robots move without contacting each other (or the obstacle) or violating the sensor constraints.

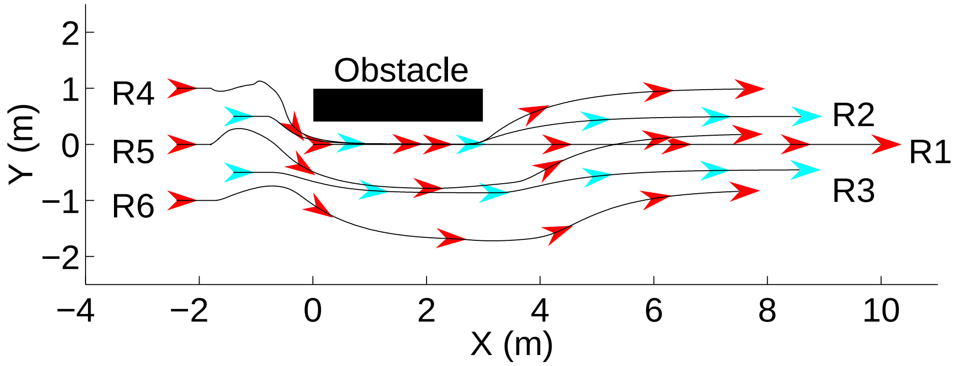

Figure 6.19: Trajectories for 6 robots avoiding an obstacle.

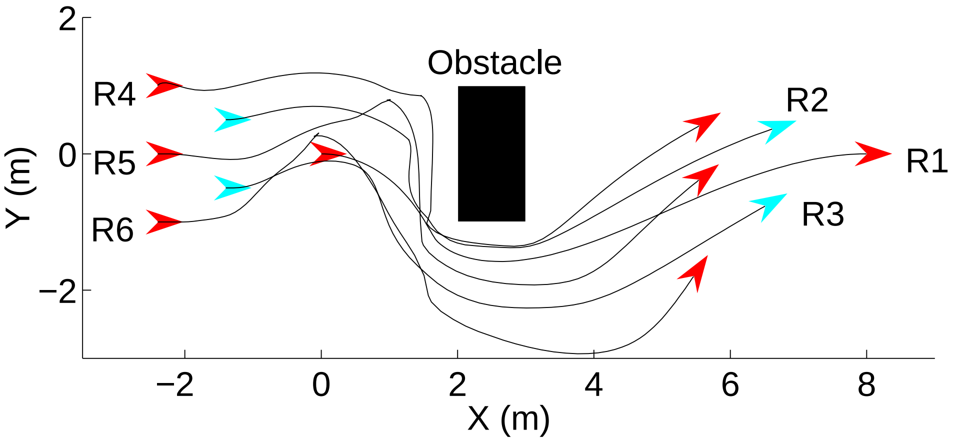

Figure 6.20: Trajectories for 6 robots avoiding an obstacle where the optimal path for the lead robot is not a straight line.

Example 3: The optimal path for R_1 is computed by minimizing a suitable norm which could be the norm of velocities or the actuator forces. In this case, we minimize the norm of the actuator forces, which gives the required trajectory for R_1. The initial and final configuration of R_1 is $(0, 0, 0)$ and $(8, 0, 0)$, respectively. The trajectories for all the robots are as shown in Figure 6.20. Initially, when the lead robot follows the trajectory avoiding the obstacle, the follower robots, R_2 and R_4 are controlled by the $l - \psi$ feedback law while the other robots are controlled by the $l - l$ feedback law as shown in Figure 6.18. However, as in Example 2, when R_2 and R_4 reach within the predefined distance of the obstacle, they switch from the $l - \psi$ controller to the $l - l$ controller to guarantee that the robot moves a safe, predefined distance from the obstacle. After the obstacle is overcome they switch back to the $l - \psi$ controller. This is shown in Figure 6.20. Note that the final orientation of the follower robots is not the same as that of the lead robot, R_1. However, as mentioned in the case of the parallel parking maneuver (Example 1), if the lead robot now continues to move in a straight line, the orientation of the follower robots will exponentially converge to that of R_1 as proved in Theorem 6.3.1.

Example 4: In this example, we consider a change in shape of the formation from a triangular to a rectangular one as shown schematically in Figure 6.21. The initial and final configurations of these robots and their respective paths are shown in Figure 6.22. We note that there is no collision among the robots when they transition from the triangular formation to the rectangular one. This is an example of formation change that is achieved

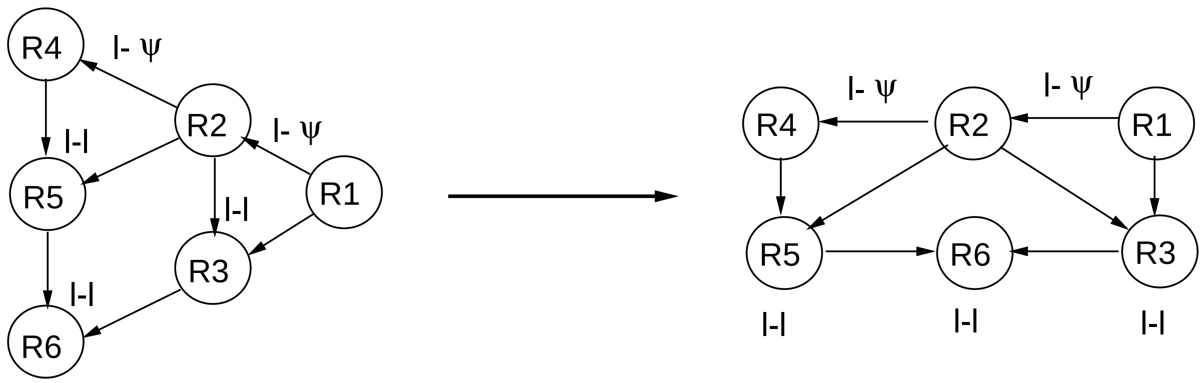

Figure 6.21: Schematic of initial and final control graphs.

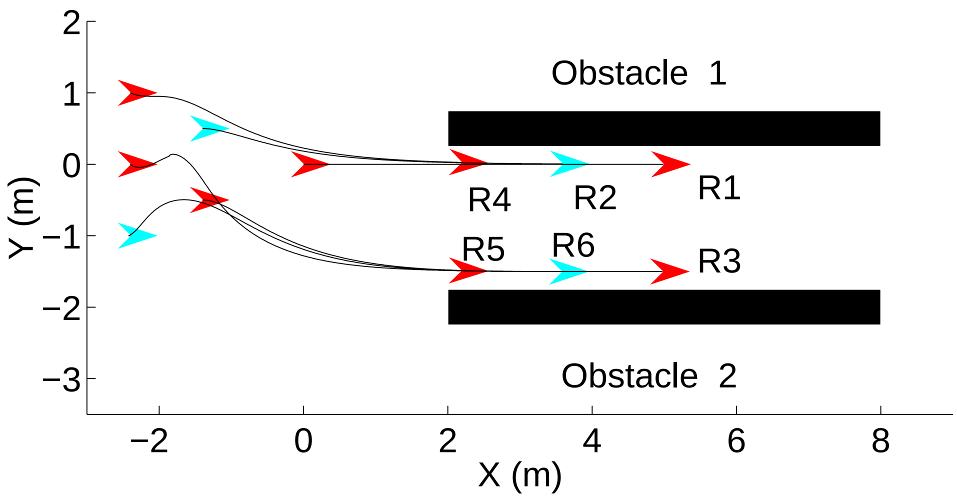

Figure 6.22: Change in formation without change in control graph.

by changing the desired values of the shape variables while maintaining the same underlying control graph at all instants of time. The purpose of this example is to illustrate that it is possible to overcome obstacles without changing the control graph.

6.5.3 Transitions between non-isomorphic control graphs

Finally, we present some results which involve changing from one formation to another. These involve changes in both the shape and the underlying control graph. In each of these examples, we will demonstrate a single transition or a combination of transitions for a particular robot based on the enumeration in Figure 6.15. Clearly, in the case of three robots the sum of members in the two equivalence classes of digraphs corresponds to the total number of allowable formations as shown in Figure 6.6. However, for a larger number of robots in the formation this is not the case and hence we need to choose one representative formation from an equivalence class as is demonstrated by Examples 5, 6 and 7 below.

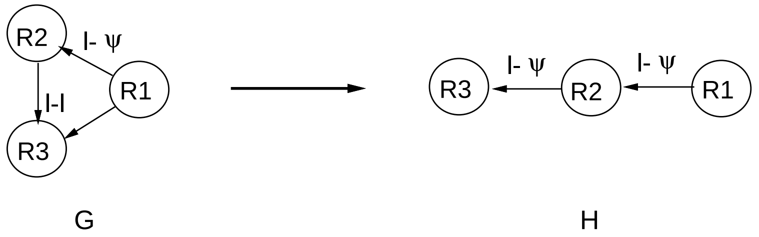

Figure 6.23: Schematic of formation change for three robots.

Example 5: In the first example of changes in control graph, we take the example of three robots moving in a triangular formation, and which need to transition to a straight line formation. The initial (I) and final (F) adjacency matrix along with the transition (T) matrix are given by Equation (6.26). The desired formation is a straight line since the narrow passage is wide enough to only let one robot go through. Clearly, they cannot maintain the original control graph as R_1 would not be in the line of sight of R_3 to obtain sensory information for maintaining its desired shape variables.

$$I = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad F = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad T = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{6.26}$$

Figure 6.23 shows the initial and final control graphs. As seen in the figure, it is necessary for R_3 to change from $l-l$ control following R_1 and R_2, to an $l-\psi$ control following R_2. Thus the transition matrix T, has a (-1) in the $(1,3)$ entry as shown in Equation (6.26). This transition represents case $4(a)$ of Figure 6.15. While the figure shows a smooth change in positions from the triangular formation to the in-line formation, there are several discrete changes that occur. First, robot R_2 senses the obstacles and initiates a change in its shape variables and then for robot R_3, to the desired new internal shape and controller dependency. Thus it guarantees that the overall formation will achieve the desired shape, *i.e.*, a straight line formation, prior to reaching the narrow constriction shown in Figure 6.24. Further, the controller for robot R_3 changes from an $l-l$ controller to an $l-\psi$ controller.

Table 6.2 shows the sequence of events that take place to achieve the desired final configuration shown in Figure 6.24. Note that r_i indicates the initial values for the shape

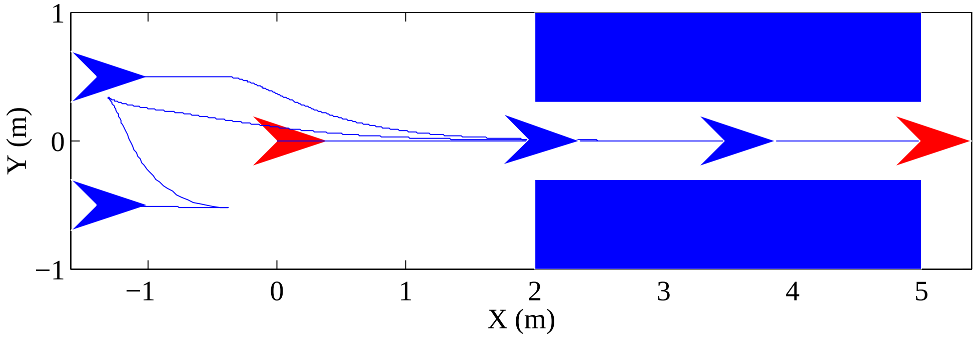

Figure 6.24: Formation change for three robots involving non-isomorphic representations.

Table 6.2: Sequence of events for Example 5.

Event	Control action
R_2 detects obstacle	Shape change: $r_2 \rightarrow \bar{r}_2$
$r_2 \rightarrow \bar{r}_2$ completed	Shape change: $r_3 \rightarrow \bar{r}_3$ Graph transition: $G_3 \rightarrow H_3$

variable and $\bar{r}_i$, the desired values for the shape variables for robot i. G_j and H_j denote the j^{th} column of the matrix. Thus for example, $r_2 = [l_{12}, \psi_{12}]^T$ and $\bar{r}_2 = [\bar{l}^d_{12}, \bar{\psi}^d_{12}]^T$. Similarly, $r_3 = [l_{13}, l_{23}]^T$ and $\bar{r}_3 = [\bar{l}^d_{23}, \bar{\psi}^d_{23}]^T$.

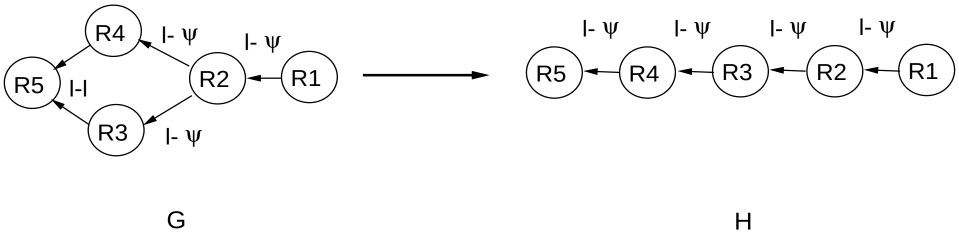

Figure 6.25: Schematic of formation change for five robots.

Example 6: In this example, we illustrate the change in formation which corresponds to transitions 1 and $4(a)$ in Figure 6.15. The initial (I) and final (F) adjacency matrix along

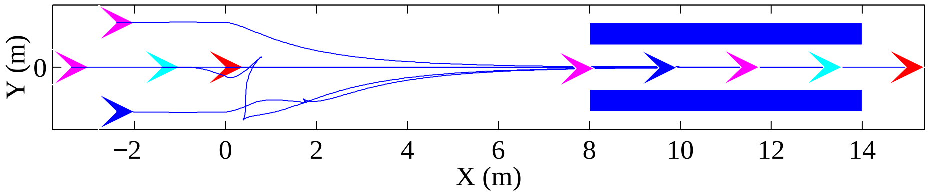

Figure 6.26: Formation change for five robots.

with the transition (T) matrix for the formation is given by:

$$I = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad F = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad T = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad (6.27)$$

Figure 6.26 shows the results of simulation for five robots changing to a straight line formation in the presence of a narrow constriction. A schematic of the initial and final control graphs is as shown in Figure 6.25. Since the only nonzero columns are given by $T_4 = [0\ -1\ 1\ 0\ 0]^T$ and $T_5 = [0\ 0\ -1\ 0\ 0]^T$ (see Figure 6.15 for notation), it implies that only R_4 and R_5 undergo changes in their control laws though the other robots may undergo a change in the shape variables (R_2 and R_3).

The situation in this example for R_5 is similar to the previous example for R_3. However, the changes in the control laws and set points for R_5 are more complex, because it needs to maintain a relative position and orientation with respect to R_4 which itself is undergoing a transition.

Table 6.3: Sequence of events for Example 6.

Event	Control action
R_2 detects obstacle	Shape change: $r_2 \to \bar{r}_2$
$r_2 \to \bar{r}_2$ completed	Shape change: $r_3 \to \bar{r}_3$
$r_3 \to \bar{r}_3$ completed	Shape change: $r_4 \to \bar{r}_4$ Graph transition: $G_4 \to H_4$
$r_4 \to \bar{r}_4$ completed $G_4 \to H_4$ completed	Shape change: $r_5 \to \bar{r}_5$ Graph transition: $G_5 \to H_5$

Table 6.3 shows the sequence of events that take place to achieve the desired final configuration shown in Figure 6.26. Note that r_i indicates the initial values for the shape

variable and $\bar{r}_i$, the desired values for the shape variables for robot i. G_j and H_j denote the j^{th} column of the matrix. Thus for example, $r_2 = [l_{12}, \psi_{12}]^T$ and $\bar{r}_2 = [\bar{l}^d_{12}, \bar{\psi}^d_{12}]^T$. Similarly, $r_3 = [l_{23}, \psi_{23}]^T$ and $\bar{r}_3 = [\bar{l}^d_{23}, \bar{\psi}^d_{23}]^T$, $r_4 = [l_{24}, \psi_{24}]^T$ and $\bar{r}_4 = [\bar{l}^d_{34}, \bar{\psi}^d_{34}]^T$, and $r_5 = [l_{35}, l_{45}]^T$ and $\bar{r}_5 = [\bar{l}^d_{45}, \bar{\psi}^d_{45}]^T$.

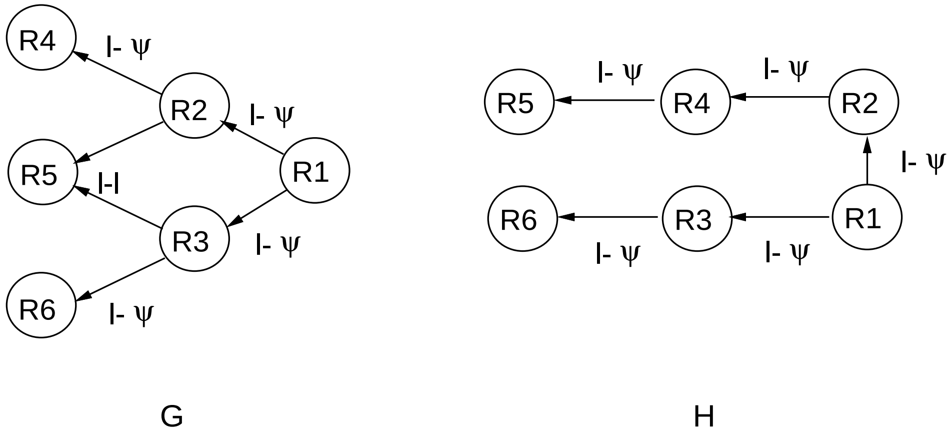

Figure 6.27: Schematic of formation change for six robots.

Example 7: Finally, in this example the task is to transition from a triangular formation to a rectangular formation. The results of the change in leaders and the transition of various robots to the desired internal shape variables is as shown in Figures 6.28 and 6.29. This example illustrates the change in formation which corresponds to transition $4(b)$ in Figure 6.15. This is an example of an intermediate $l - l$ control which robot R_5 has to undergo before stabilizing to the desired $l - \psi$ control with respect to R_4. The initial (I) and final (F) adjacency matrix along with the transition (T) matrix for the formation is given by:

$$I = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad F = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad T = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{6.28}$$

As seen from the transition matrix, the only robot undergoing change in the leader is R_5. The corresponding column using the notation of Figure 6.15 is given by $T_5 = [0\ -1\ -1\ 1\ 0\ 0]^T$. A schematic of the initial and final control graphs is as shown in Figure 6.27.

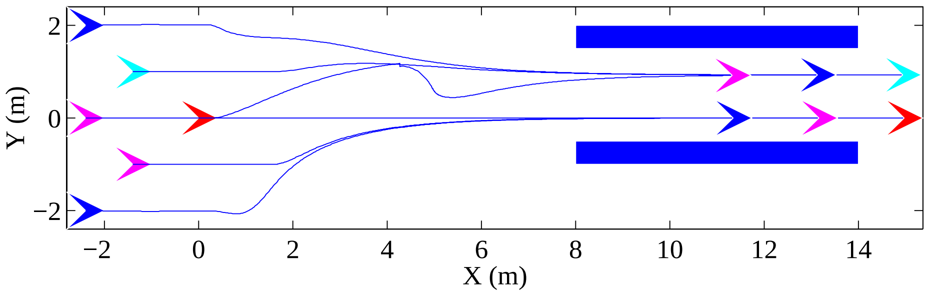

Figure 6.28: Formation change squeezing through a narrow passage.

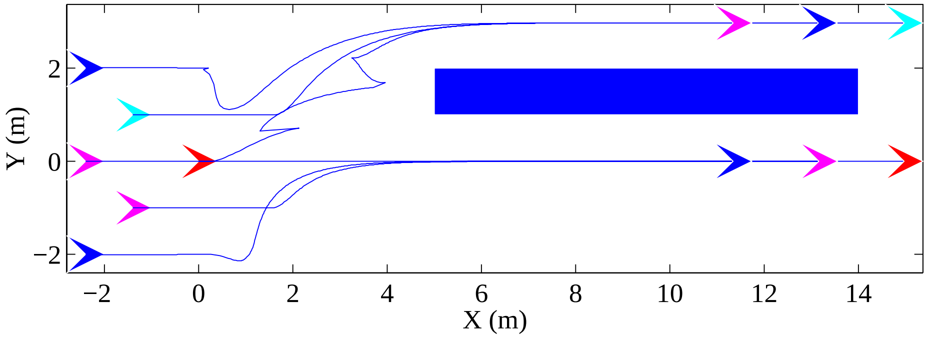

Figure 6.29: Formation change for six robots going around an obstacle.

Since R_5 changes from an $l-l$ control with respect to R_2 and R_3 to an $l-\psi$ control with respect to R_4, it has to break links with two robots and form a new link with a different robot. Once the transition process initiates for R_5, it severs its link with R_2 and forms a new link with R_4. It thus maintains an intermediate $l-l$ control with R_3 and R_4 until it stabilizes the separation to the desired values before initiating the final severing of connection with R_3. Once it has severed connection with R_3, it starts stabilizing its orientation with respect to R_4. The formation finally converges to the desired rectangular formation as shown in Figure 6.28. Figure 6.29 also illustrates similar transitions in shape and control graph for all the robots. However, here since the rectangular formation is the simplest complexity control graph, *i.e.*, a tree, it is possible to go around obstacles with this simpler control graph rather than original one. In this example, we assume that the robots can see each other over obstacles (since R_2 must see R_1).

Table 6.4 shows the sequence of events that take place to achieve the desired final configuration shown in Figures 6.28 and 6.29. Note that r_i indicates the initial values

Table 6.4: Sequence of events for Example 7.

Event	Control action
R_2 detects obstacle	Shape change: $r_2 \rightarrow \bar{r}_2$
$r_2 \rightarrow \bar{r}_2$ completed	Shape change: $r_3 \rightarrow \bar{r}_3$
$r_3 \rightarrow \bar{r}_3$ completed	Shape change: $r_4 \rightarrow \bar{r}_4$
$r_4 \rightarrow \bar{r}_4$ completed	Shape change: $r_5 \rightarrow \bar{r}_5$ Graph transition: $G_5 \rightarrow H_5$
$r_5 \rightarrow \bar{r}_5$ completed $G_5 \rightarrow H_5$ completed	Shape change: $r_6 \rightarrow \bar{r}_6$

for the shape variable and $\bar{r}_i$, the desired values for the shape variables for robot i. G_j and H_j denote the j^{th} column of the matrix. Thus for example, $r_2 = [l_{12}, \psi_{12}]^T$ and $\bar{r}_2 = [\bar{l}^d_{12}, \bar{\psi}^d_{12}]^T$. Similarly, $r_3 = [l_{13}, \psi_{13}]^T$ and $\bar{r}_3 = [\bar{l}^d_{13}, \bar{\psi}^d_{13}]^T$, $r_4 = [l_{24}, \psi_{24}]^T$ and $\bar{r}_4 = [\bar{l}^d_{24}, \bar{\psi}^d_{24}]^T$, $r_5 = [l_{25}, l_{35}]^T$ and $\bar{r}_5 = [\bar{l}^d_{45}, \bar{\psi}^d_{45}]^T$, and $r_6 = [l_{36}, \psi_{36}]^T$ and $\bar{r}_6 = [\bar{l}^d_{36}, \bar{\psi}^d_{36}]^T$.

6.6 Conclusions

In this chapter, we have studied strategies for controlling formations of mobile robots using methods from nonlinear control theory and graph theory. We have focused on decomposing the problem of controlling a formation into: a) controlling a single lead robot and b) controlling other follower robots in the team. We used the terms $l - \psi$ and $l - l$ control to reflect whether the control laws are based on tracking the position and orientation of the robot relative to a leader, or the position relative to two leaders, respectively. It was assumed that there is a single lead robot for the overall formation whose motion plans are generated externally. These results apply to general formations moving through an environment that may include obstacles. The two controllers proposed here are used to achieve this objective. The $l - \psi$ and the $l - l$ controllers guarantee the correct formation, while the modified $l - l$ controller can also be used to change the shape of our formation in the presence of obstacles. This methodology was illustrated by numerous examples both in the presence and absence of obstacles.

We also studied the problem of changes in formation. In practical applications involving obstacles or narrow passageways it may be necessary to transition between non-isomorphic control graphs. Thus, to achieve these transitions, we developed the idea of a transition matrix. We also proved the upper bound on the total number of allowable control graphs for a given number of robots based on two constraining assumptions. We further used the idea of graphical enumeration for digraphs and specialized it to generate equivalence classes of allowable control graphs which satisfy constraint A and B automatically. Such an array of different control graphs can be used in conjunction with the set points for the internal shape variables as a library of possible shapes which can be invoked based on the task requirement. Once a particular control graph is chosen, we can label its vertices based on constraint B. Further, we also enumerated the possible transitions for a robot when it changes leader(s) in the transition process. The enumerated transitions guarantee a unique way to change from one control graph to another modulo the order in which the connections are made or broken with the initial leaders in the formation. These enumerations were used to solve numerous examples involving non-isomorphic control graph transitions. The computation time for optimal trajectory planning based on the methodology of Chapter 5 for two robots avoiding a circular obstacle requires at least 2 min. for a specifically tailored compiled code and the formation may not necessarily be maintained. In contrast, the motion for the formation of six robots around an obstacle of higher complexity such as the one shown in Figure 6.28 has a run time of about 10 s. in Matlab.

Though the transition matrix gives us the information needed to change formations, it is not clear if there is an optimal way for carrying out these changes, rather than the sequential algorithm given here. Also, is it possible to carry out these changes without having an isolated robot at any stage in the transition process? It is possible to have changes in only the group variables, g for the lead robot without changing the internal shape, r or the control graph, $\mathcal{H}$ as in the example of the parallel parking maneuver. Similarly, we can have changes in g and r simultaneously, without changing the control graph H as illustrated in Examples 2 through 4. Finally, we can have changes in all the three variables $(g, r, \mathcal{H})$, as illustrated by Examples 5 through 7. However there are a couple of important issues that need to be addressed in future research in this area: a) how to choose a control graph and the desired shape based on the constraints in the environment? and b) sensor constraints and changes in $(g, r, \mathcal{H})$ have to be planned concurrently to make a general motion planner for a large number of robots so that the robots do not collide with the obstacles and among themselves.

The work presented in this chapter enumerates all possible allowable control graphs, but it does not give an algorithm for selecting the appropriate control graph and the shape of the formation. Finally, experimental verification of these strategies is definitely a very challenging issue. Developing a system of autonomous robots that can maintain a formation while avoiding obstacles will yield many practical insights that can be used to develop better planning and control algorithms.

Chapter 7

Conclusions and future work

This thesis deals with the general problem of motion planning in systems with two or more cooperating arms or robots. We have adopted a framework that allows us to concurrently study and compare biological and robotic systems. We have developed methods that allow us to plan trajectories and actuator forces for cooperating robot systems. The same methods lend themselves to modeling trajectory generation and force distribution in human dual arm manipulation. Further, we have pursued the extension of this framework and the methods to systems with many (tens of) cooperating robots.

In the following section, we will briefly summarize the material presented in the thesis and then highlight the major contributions of this work. Finally, we will present some possible areas of future research that come naturally from this thesis.

7.1 Summary

In Chapter 1, we introduced the subject of motion planning and defined the motion planning problem for a kinematic system as given by Latombe. We then motivated the need for a more general definition of motion planning that is relevant to more complex manipulation tasks that are typical in cooperating robots. In particular, we considered a system of robots lifting and carrying large (but not necessarily heavy), awkwardly sized payloads such as barrels or boxes, in which the dynamics of the manipulation task played a significant role. In such tasks, the robots typically needed to avoid obstacles and plan their trajectories, while being subject to both kinematic and dynamic constraints. Such cooperative manipulation tasks were similar to the cooperative manipulation tasks in which two human arms are used to transport an object from one position and orientation to another.

In Chapter 2, we described extensively the literature in the general area of motion planning for human and robotic systems. Although there is significant literature on the subject of trajectory generation for single arm reaching, there is no model which describes the behavior of the system for complex tasks in which the dynamics of the musculo-skeletal system plays a significant role. One of the reasons for a lack of complete model is the difficulty in understanding complex set of commands and strategies used by the central nervous system(CNS) for planning motion from one position and orientation to another.

Similarly, in the area of cooperative robotic manipulation tasks, there is a need for motion planning algorithms that find optimal trajectories and force distributions for cooperating robots while satisfying equality and inequality constraints. Traditionally, the motion planning problem defined by Latombe, involved finding a feasible path (if there exists one) for a single robot subject to geometric constraints. Since we are interested in deriving continuous motion plans for our experimental system, we need to extend this definition to account for more complicated motion planning tasks. These problems involve constraints ranging from such kinematic constraints as constraints on the turning radius of a mobile robot to inequalities which bound the normal forces, *i.e.*, dynamic constraints. All these requirements make the motion planning problem quite complicated. We reviewed the pertinent existing literature which addresses some of the issues in this area.

In Chapter 3, we discussed the optimal control problem and the solution of Pontryagin for various cases involving constraints on the state and/or the control inputs to the system. An equivalent problem in the variational calculus domain known as the problem of Bolza was also studied and the critical solutions for the problem, namely the Euler-Lagrange equations were derived along with the Weierstrass-Erdmann corner conditions. We stated conditions under which the critical solutions of the optimal control problem are also the critical solutions of the unconstrained variational problem and vice-versa. The motivation for formulating the optimal control problem as a problem in the unconstrained variational calculus framework is two-fold. Firstly, obtaining solutions to the optimal control problem using the theorem of Pontryagin for complex problems by using a numerical method is not an easy task. Secondly, solving constrained optimal control problems is not straightforward as numerical techniques for solving two-point boundary value problems are not very efficient. Our approach overcomes both these difficulties. Further, we derived explicit formulae for the total number of variables in the unconstrained formulation based on new adjoined constraints. We also discussed methods for stabilizing constraints at each mesh point for greater accuracy and increased convergence to the optimal solution. Finally, we also discussed the numerical technique for solving the reformulated problem.

In Chapter 4, we used mathematical modeling techniques of Chapter 3 to model trajectory generation and force distribution in human dual arm manipulation. We studied the bi-manual manipulation task, both in the horizontal and vertical plane and the motion planning problem. Studies of planar human manipulation tasks showed that the trajectories are repeatable across trials and across subjects and that these trajectories and velocity profiles are smooth. Experimental results illustrated an invariance and certain symmetries in trajectories in these manipulation tasks. Our studies suggested that humans may employ a strategy that minimized a suitable integral cost function. This motivated our computational model for modeling trajectory generation and force distribution. We modeled the planning problem as an optimal control problem that involved minimizing the rate of change of actuator torques subject to kinematic and dynamic constraints. The results of this model compared favorably to experimentally observed trajectories, velocities and force profiles for a class of manipulation tasks. The model also predicted an increase in the interaction force as the weight on the grasped object increased, a feature that is also observed in experiments. Though the analytical model did predict the observed trajectories and forces, we are currently unable to account for the dependence of internal forces on the hand velocities. Our results in vertical arm movements were significantly different that those observed in the planar case. The vertical arm trajectories for upward movements exhibited curvature while the downward movements were straight. Though the shape of the internal force profile was repeatable across subjects, their magnitudes varied from one subject to the other. Also the magnitudes of internal forces varied significantly according to the direction of motion in the sagittal plane. The force profiles for these motions showed a higher internal force for the $5 \rightarrow 8$ motion than the $8 \rightarrow 5$ motion. This could be attributed to the fact that in one case work is done against gravity ($5 \rightarrow 8$ motion) while in the other case, it is along the direction of the gravitational force ($8 \rightarrow 5$ motion). Frontal plane movements also exhibited near straight line trajectories and the force profiles are similar to those in the planar case.

In Chapter 5, we explored the problem of cooperative robot motion planning based on findings from human manipulation studies. We considered car-like platforms equipped with robot arms and the task of cooperatively transporting objects. We adopted a cost function similar to the one used in Chapter 4. By minimizing the norm of the actuator forces/torques, we achieved a smooth velocity profile for our platforms which are velocity controlled. Smoother velocity profiles result in smaller dead-reckoning errors caused by slippage on the floor. We demonstrated the generality of the method by considering one other example, the example of the snakeboard. In this system, the nonholonomic constraints are such that the mobility of the system cannot be modeled using just the kinematics. It is necessary to incorporate the dynamics of the system. We studied gait transitions in the snakeboard and showed that smooth motions can be obtained.

When we consider a larger number of robots ($n > 2$), the complexity of the motion planning problem greatly increases. In Chapter 6, we argued that it makes sense to use a hybrid of methodologies, *i.e.*, both centralized and decentralized control strategies to achieve the required objective. We motivated the problem of controlling the formation of a team of robots. Such a task may typically involve changes in both the shape and the underlying control strategy. These strategies are also observed in microscopic organisms like the paramecium or the amoeba and hence serve as basic building blocks for modeling such problems. The motion planning problem for a large formation of robots was broken down into two sub-problems. Firstly, we designed suitable controllers to maintain the shape of a formation and avoid obstacles in the process and secondly, we defined the connectivity constraints between robots and how to achieve transition from one formation to another. We also specialized results from graph theory to our specific problem and developed an algorithm for choosing allowable control graphs. Along with the idea of the transition matrix, we also enumerated possible changes in control graphs and outlined the algorithm for transiting from one control graph to another. In all motion planning tasks in this chapter, we used techniques from optimal control based on Chapter 3 to plan an optimal trajectory for the lead robot based on geometric and kinematic constraints.

7.2 Contributions

This thesis makes significant contributions in the following areas:

- The problem of motion planning for general dynamical systems while optimizing a suitable cost function is a difficult problem. The complexity of the problem increases as different types of constraints are present in the system, like, state and input based equality and inequality constraints. One of the main contributions of the thesis is to handle these varied constraints in a unified framework and illustrate the effectiveness of the procedure by modeling and simulating highly complex systems like the system of cooperating robots or understanding coordinated two-arm manipulation tasks by humans. Having formulated the optimal control problem in the framework of unconstrained variational calculus, we further develop the notion of differentiating constraints and their stabilization at each mesh-point by adjoining the lower order derivatives with suitable constant coefficients. This ensures that the constraints exponentially converged at each mesh-point.

- The second significant contribution is in the area of coordinated human manipulation. So far, researchers have investigated single-arm manipulation tasks in the horizontal or vertical plane and coordination of hands and fingers for small movements without incorporating the dynamics of the human arm. In this thesis, we have investigated dual arm manipulation tasks for large movements and developed a computational model for the first time. Unlike previous studies, we studied tasks that rely on frictional forces for stable prehension of the object as the object is free to slide on a vertical column. The subject needs to hold the object while carrying out the manipulation task. Our computational model predicts many important features of human trajectory generation and load distribution in planar manipulation tasks. We also investigated dual-arm manipulation tasks in the vertical plane and our experimental data gave insight into the trajectories and force distribution between the two arms while performing a typical manipulation task.

- The third contribution is a systematic study of the problem of cooperative robot motion planning for a team of autonomous robots. We developed motion planning algorithms for robot systems consisting of mobile platforms equipped with robot arms. The motion plans were implemented and tested on an experimental testbed which consisted of two TRC-Labmate platforms. In order to illustrate the generality of our methods, we used the same algorithms to generate motion plans for a nonholonomic system called the snakeboard. We obtained numerical solutions for optimal gaits and transitions between the gaits based on certain fixed end-point conditions.

- Finally, we also studied the problem of coordinating a large number of robots and controlling the formation. Since the task of computing optimal motion plans for a large number of robots is computationally expensive, we designed a new method and a set of algorithms for such a system. The basic idea is to combine open loop motion plans with closed loop motion plans or feedback control laws. The open loop plans for one or more leader platforms are generated using optimal control methods, while feedback control laws for follower platforms/robots guarantee that the followers and the leaders remain in a tight formation. In the presence of severe free-space constraints in the environment, it is necessary to transition between non-isomorphic control graphs. Thus to achieve these transitions, we developed the idea of a transition matrix. We also proved the upper bound on the total number for possible formations, given the number of robots based on two constraining assumptions. We further used the idea of graphical enumeration for digraphs and developed an algorithm to choose allowable control graphs. Such an array of different formations can be used as a library of possible formations which can be recalled based on the task requirement. Further, we also enumerated the possible transitions for a robot when it changes its controller dependency. The enumerated transitions guarantee a unique way to change formation modulo the order in which the connections are made or broken with the initial leaders in the formation.

7.3 Future work

In this thesis, we have successfully investigated the motion planning problem for a class of dynamical systems and proposed new methodologies for the motion planning problem. Though we have obtained solutions for the general motion planning problem, there are areas of future research.

1. The proposed methodology for solving optimal control problems in the variational framework uses first order necessary conditions for finding a critical solution. It is meaningful to investigate and prove some results on the global or strong optimality of certain class of problems with specific types of constraints.

2. In the area of human manipulation, there have been no reported studies on general spatial motion planning[1] and computational models for understanding these tasks. Part of the difficulty for these studies lies in design and construction of an appropriate testbed. For planar motion, we can use encoders as used in the experiments described here, but it is not clear how to construct a spherical joint with encoders to capture the orientation of a spatial manipulandum. However, with the availability of motion detectors like the OPTOTRAK system or the POLARIS system, these limitations can be easily overcome. It will be interesting to study the trajectories and velocity profiles for general spatial manipulation tasks using results from single and dual arm movements based on existing observations in horizontal and vertical arm movements. Finally, building a computational model for these observed data in the spatial domain is the next logical step.

[1] We note that there have been studies on vertical arm movements by Atkeson and Hollerbach [6] and others, but these motions are not general.

3. The area of motion planning for general robotic systems has been explored in great detail in this thesis from the point of view of planning optimal motions. The methodology presented is very general to the types and number of constraints. One of the biggest hurdles in extending this methodology to planning motions for a larger number of robots is the increase in computational complexity of the algorithm. We can, however use methods of optimal control to plan the initial trajectory of the lead robot based on the obstacles in the environment. This can be combined with local sensory information acquired on-line by the robots and using the robots own control laws developed in Chapter 6 to plan the motion for the team. Thus using centralized and decentralized control is well suited to this problem. Though, we have developed some groundwork in this direction, there are a host of issues that need to be explored. These are summarized below:

    - Though the transition matrix gives us the algorithm for changing formations by sequential changes in the control graph, it is not understood if there is an optimal way for carrying out these changes rather than a sequential pattern. For example, if there are a lot of changes in the initial control graph, is it possible to find an optimal way to carry out these changes while dynamically relabeling the vertices (except robot 1) in the digraph? Also, is it possible to carry out these changes without having an isolated robot at any stage in the transition process? One possible cost function to study optimality of transitions could be to minimize the number of "make and break" connections.
    - Though it is possible to have changes in some or all of the variables in the triple, $(g, r, \mathcal{H})$, it is not clear, how to choose a control graph and the desired shape based on the constraints in the environment? The work presented here enumerates all possible control graphs that are allowable but it does not give an algorithm for selecting the appropriate control graph and the shape of the formation. Thus, it will be interesting to design a motion planner which will decide "on the fly", the type of formation required to overcome the obstacles in the environment based on the sensory data obtained by the lead/leader robots in the formation.

    Finally, experimental verification of these strategies is definitely a very challenging issue. Developing a system of autonomous robots that can maintain a formation while avoiding obstacles will yield many practical insights that can be used to develop better planning and control algorithms.

Appendix A

Human manipulation

A.1 Data analysis

The recorded encoder readings were analyzed off-line, and the corresponding joint angles of the manipulandum were derived after the data was processed with a third order Butterworth low pass filter using a cutoff frequency of 7.5 Hertz. The corresponding angular velocities were obtained by numerical differentiation using a five-point Lagrangian difference method [56]. The Cartesian trajectories and velocity profiles were then derived by employing the forward kinematics transformations for the manipulandum.

For each measured motion, only those components for which the prescribed amplitudes were non-zero were considered. Thus for example, in the case of pure translatory motion (no rotational component), the rotational components of motions were discarded. Therefore, only components having high signal/noise ratio were used. These components are referred to as the "significant components" of motion. Considering each significant component separately, the start and end times were taken as the point in time when the velocity of that component has reached 5% of its peak velocity. For a given motion, the minimum of the start times of its significant components was taken as the motion start time, and the maximum of their end times was taken as the motions end time. Since in general, the duration of motion may be different for the repetition of that particular motion, we normalized the data so that we can compare the data across trials for different motions. Thus the durations of motion were normalized based on the methods of Atkeson and Morasso [6, 91]. For each motion, the normalized time, (τ) was:

$$\tau = \frac{t - t_0}{t_f - t_0} \tag{A.1}$$

Similarly, for each motion component (e.g., x), the normalized amplitude $(\hat{x})$ was taken as

$$\hat{x} = \frac{x(t) - x(t_0)}{x(t_f) - x(t_0)} \tag{A.2}$$

In order to normalize the velocity profiles, Equation (A.2) was differentiated with respect to normalized time. Since,

$$\frac{d\hat{x}}{d\tau} = (\frac{d\hat{x}}{dt}) \ \ (\frac{dt}{d\tau}) \tag{A.3}$$

and

$$\frac{dt}{d\tau} = t_f - t_0 \ , \tag{A.4}$$

it follows that:

$$\frac{d\hat{x}}{d\tau} = \dot{x}\frac{t_f - t_0}{x(t_f) - x(t_0)} \tag{A.5}$$

A.2 Human data

In this section, we will present some sample experimental data and the physical parameters required from the human subject in order to model the manipulation task. Figure A.1 shows the schematic of the human subject and the critical parameters like the length of upper-arm, fore-arm, and hands. The data for six subjects is as shown in Table A.1. Based on the table and the correlations in Winter [130], the mass of the upper-arm, fore-arm and

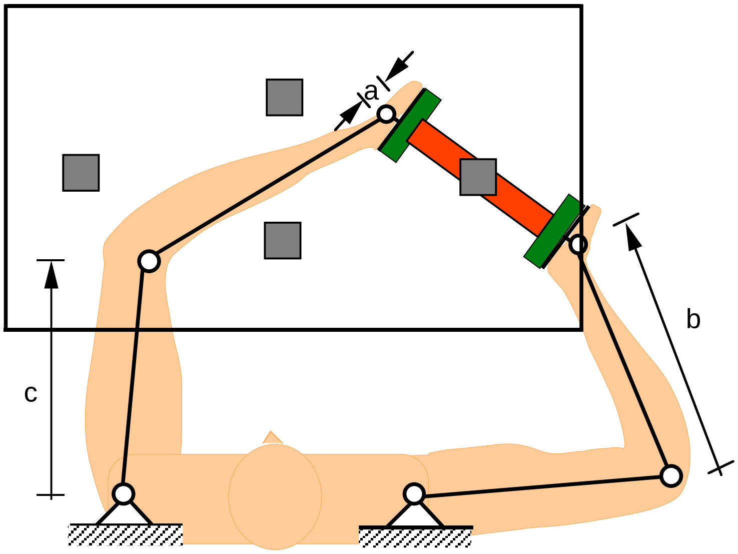

Figure A.1: Schematic of essential parameters required for modeling human motion.

Table A.1: Physical parameters for six subjects.

Subjects	a	b	c	Weight (W)
S1	$2\frac{6}{8}$ in	$11\frac{7}{8}$ in	$12\frac{1}{8}$ in	165 lbs.
S2	$2\frac{4}{8}$ in	$10\frac{4}{8}$ in	$11\frac{4}{8}$ in	160 lbs.
S3	$3\frac{1}{8}$ in	$10\frac{5}{8}$ in	$12\frac{4}{8}$ in	190 lbs.
S4	$2\frac{1}{8}$ in	10 in	$11\frac{4}{8}$ in	170 lbs.
S5	2 in	11 in	13 in	200 lbs.
S6	$1\frac{6}{8}$ in	11 in	$11\frac{4}{8}$ in	145 lbs.

the hand grasping the object can be computed by the following relations:

$$m_{upper} = 0.028W$$

$$m_{forearm} = 0.016W$$

$$m_{hand} = 0.006W$$

where, W is the weight of the subject. The radius of gyration of each arm is obtained through similar correlations from Winter and is used to compute the moment of inertia of the upper arm, forearm and hand. These moments of inertia are given by:

$$I_{upper} = m_{upper}(0.322c)^2$$

$$I_{forearm} = m_{forearm}(0.303b)^2$$

$$I_{hand} = m_{hand}(0.297a)^2$$

The physical parameters, a, b and c along with the mass of each arm segment is used to compute the inertia matrix for each arm. The inertia of the force sensors along three principal directions is given by: $I_{xx} = I_{zz} = 7.18 \times 10^{-5}\ kg.m^2$ and $I_{yy} = 1.19 \times 10^{-4}\ kg.m^2$

Sample raw data collected from the OPTOTRAK is shown in Table A.2 and the raw force data is shown in Table A.3.

A.3 Dynamics of a three link serial manipulator

We use Lagrange's equations to derive the dynamics of a three link serial chain manipulator as that is the model for the human arm. Let v_i represent the translational velocity of the center of mass of the i^{th} link and ω_i $(= \dot{\theta}_i)$ its angular velocity in the z direction. Let I_{zi} be the moment of inertia in the z direction of the i^{th} link relative to the frame attached

Table A.2: Sample raw data from the OPTOTRAK.

x_1 (mm)	y_1 (mm)	z_1 (mm)	x_2 (mm)	y_2 (mm)	z_2 (mm)
103.69	-153.03	-1713.29	100.90	-215.67	-1723.19
103.31	-152.88	-1713.48	100.43	-215.50	-1723.47
102.98	-152.72	-1713.68	100.01	-215.32	-1723.75
102.66	-152.52	-1713.87	99.64	-215.11	-1724.03
102.36	-152.27	-1714.07	99.32	-214.85	-1724.30

Table A.3: Sample raw data from the force sensors.

F_x	F_y	F_z	τ_x	τ_y	τ_z
55	-77	-321	49	52	-287
56	-78	-320	48	52	-286
56	-78	-320	49	53	-286
56	-78	-319	48	54	-285
56	-77	-318	48	55	-283

to the center of each link and oriented along the principal axes. Since the motion of the mechanism is in the plane, the Lagrangian is the kinetic energy of the system and is given by:

$$L(\Theta,\dot{\Theta}) = \frac{1}{2}\sum_{i=1}^{3} m_i v_i^2 + I_{zi}\dot{\theta}_i^2 \tag{A.6}$$

where $\Theta = [\theta_1,\ \theta_2,\ \theta_3]^T$ is the 3×1 vector of generalized coordinates representing the relative joint angles as shown in Figure A.2.

Let r_i represent the distance from the joint to the center of mass of the i^{th} link and l_i the length of each link as shown in Figure A.2. Hence, the coordinates of the center of mass for each link is given by:

$$\begin{aligned} \hat{x}_1 &= r_1 c_1 & \hat{y}_1 &= r_1 s_1 \\ \hat{x}_2 &= l_1 c_1 + r_2 c_{12} & \hat{y}_2 &= l_1 s_1 + r_2 s_{12} \\ \hat{x}_3 &= l_1 c_1 + l_2 c_{12} + r_3 c_{123} & \hat{y}_3 &= l_1 s_1 + l_2 s_{12} + r_3 s_{123} \end{aligned} \tag{A.7}$$

where $c_1 = \cos\theta_1$, $c_{12} = \cos(\theta_1 + \theta_2)$, $c_{123} = \cos(\theta_1 + \theta_2 + \theta_3)$ and similarly, $s_1 = \sin\theta_1$, $s_{12} = \sin(\theta_1 + \theta_2)$, $s_{123} = \sin(\theta_1 + \theta_2 + \theta_3)$. The translational velocity of the center of mass of each link is thus: $v_i^2 = \dot{\hat{x}}_i^2 + \dot{\hat{y}}_i^2$ $(i = 1, 2, 3)$ where $\dot{\hat{x}}_i$ and $\dot{\hat{y}}_i$ are given by:

$$\dot{\hat{x}}_1 = -r_1 s_1 \dot{\theta}_1$$

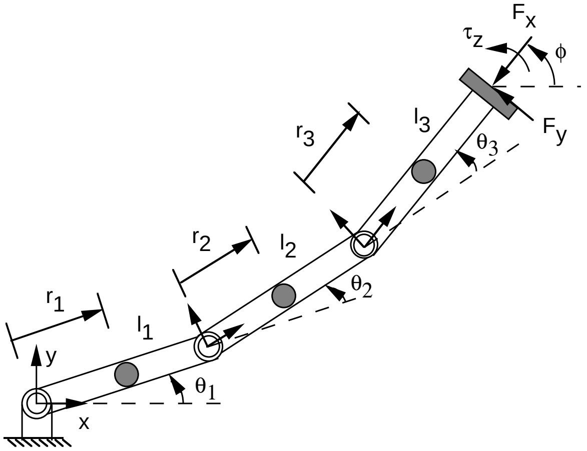

Figure A.2: A three link serial manipulator.

$$\begin{aligned}
\dot{y}_1 &= r_1 c_1 \dot{\theta}_1 \\
\dot{x}_2 &= -(l_1 s_1 + r_2 s_{12})\dot{\theta}_1 - r_2 s_{12}\dot{\theta}_2 \\
\dot{y}_2 &= (l_1 c_1 + r_2 c_{12})\dot{\theta}_1 + r_2 c_{12}\dot{\theta}_2 \\
\dot{x}_3 &= -(l_1 s_1 + l_2 s_{12} + r_3 s_{123})\dot{\theta}_1 - (l_2 s_{12} + r_3 s_{123})\dot{\theta}_2 - r_3 s_{123}\dot{\theta}_3 \\
\dot{y}_3 &= (l_1 c_1 + l_2 c_{12} + r_3 c_{123})\dot{\theta}_1 + (l_2 c_{12} + r_3 c_{123})\dot{\theta}_2 + r_3 c_{123}\dot{\theta}_3
\end{aligned} \tag{A.8}$$

Lagrange's equations are given by:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_i}\right) - \frac{\partial L}{\partial \theta_i} = \Gamma_i \tag{A.9}$$

where Γ_i is the sum of all the external forces acting on the body along the generalized coordinate, θ_i. The vector, Γ is a 3×1 vector and is given by:

$$\Gamma = \tau - J^T F$$

where J is the manipulator Jacobian and $F = [F_x,\ F_y,\ \tau_z]^T$ is the vector of external forces acting on the end effector as shown in Figure A.2 and $\tau = [\tau_1,\ \tau_2,\ \tau_3]^T$ is the vector of joint torques at each link joint. The end-effector coordinates are given by, $p = [x,\ y,\ \phi]^T$ where:

$$\begin{aligned}
x &= l_1 c_1 + l_2 c_{12} + l_3 c_{123} \\
y &= l_1 s_1 + l_2 s_{12} + l_3 s_{123} \\
\phi &= \theta_1 + \theta_2 + \theta_3
\end{aligned} \tag{A.10}$$

Thus, the manipulator Jacobian is given by:

$$J = \begin{bmatrix} \frac{\partial x}{\partial \theta_1} & \frac{\partial x}{\partial \theta_2} & \frac{\partial x}{\partial \theta_3} \\ \frac{\partial y}{\partial \theta_1} & \frac{\partial y}{\partial \theta_2} & \frac{\partial y}{\partial \theta_3} \\ \frac{\partial \phi}{\partial \theta_1} & \frac{\partial \phi}{\partial \theta_2} & \frac{\partial \phi}{\partial \theta_3} \end{bmatrix} \tag{A.11}$$

Using Equation (A.10), we can easily calculate the manipulator Jacobian J, by evaluating the various entries of the matrix in Equation (A.11). Once we have calculated J, we can use the Lagrangian given by Equation (A.6) and Lagrange's equations given by Equation (A.9) to derive the equations of motion for the three link manipulator. After some simplification, the equations for each arm can be compactly written as:

$$I(\Theta)\ddot{\Theta} + C(\Theta, \dot{\Theta}) = \tau - J^T F$$

where $\Theta = [\theta_1, \theta_2, \theta_3]^T$ is the 3×1 vector of the joint coordinates, $I(\Theta)$ is the 3×3 inertia matrix, $C(\Theta, \dot{\Theta})$ is the 3×1 vector of nonlinear terms (Coriolis and centrifugal forces), τ is the 3×1 vector of joint torques, J is the 3×3 Jacobian matrix relating the velocity of the center of mass of the object to the joint velocities and F is the 3×1 generalized force vector, representing the force exerted by the manipulator on the object and the moment about the center of mass.

Bibliography

[1] J. Adams, R. Bajcsy, J. Kosecka, V. Kumar, R. Mandelbaum, M. Mintz, R. Paul, C.C. Wang, Y. Yamamoto, and X. Yun. Cooperative material handling by human and robotic agents: Module development and system synthesis. In *Proceedings of the International Conference on Intelligent Robots and Systems*, pp. 200–205, Pittsburgh, PA, August 1995.

[2] M. Ahmadabadi and N. Eiji. Constrain and move: A new concept to develop distributed object transferring protocols. In *Proceedings of 1997 IEEE International Conference on Robotics and Automation*, Albuquerque, New Mexico, April 1997.

[3] U. S. Army. *FM100-5: Operations*. Headquarters of the Army, 1993. http://155.217.58.58/cgi-bin/atdl.dll/fm/100-5/100-5toc.htm.

[4] H. Asada and J.-J. E. Slotine. *Robot Analysis and Control*. John Wiley & Sons, New York, 1986.

[5] H. Asama, M. Sato, N. Goto, H. Kaetsu, A. Matsumoto, and I. Endo. Mutual transportation of cooperative mobile robots using forklift mechanisms. In *Proceedings of 1996 IEEE International Conference on Robotics and Automation*, Minnesota, Minneapolis, April 1996.

[6] C. G. Atkeson and J. M. Hollerbach. Kinematic features of unrestrained vertical arm movements. *The Journal of Neuroscience*, 5(9), pp. 2318–2330, 1985.

[7] J. Baillieul. Geometric methods for nonlinear optimal control problems. *Journal of optimization theory and applications*, 25(4), pp. 519–549, August 1978.

[8] R. Bajcsy, V. Kumar, M. Mintz, R. Paul, and X. Yun. Cooperative agents: Machines and humans. In *Proceedings of the International Conference on Automation and Robotics*, Tokyo, Japan, November 1993.

[9] T. Balch and R. C. Arkin. Behavior-based formation control for multi-robot teams. *IEEE Trans. on Robotics and Automation*, 1999. To be published in 1999.

[10] T. Balch and R.C. Arkin. Communication in reactive multiagent robotic systems. *Autonomous Robots*, 1(1), pp. 27–52, 1994.

[11] J. Barraquand, B. Langlois, and J. C. Latombe. Robot motion planning with many degrees of freedom and dynamic constraints. In *Fifth International symposium on Robotics research*, Tokyo, Japan, 1989.

[12] J. Barraquand and J. C. Latombe. On nonholonomic mobile robots and optimal maneuvering. *Revue d'Intelligence Artificielle*, 3, pp. 1746–1756, 1989.

[13] J. Baumgarte. Stabilization of constraints and integrals of motion in dynamic systems. *Computer Methods in Applied Mechanics and Engineering*, pp. 1–16, 1972.

[14] D. P. Bertsekas. *Constrained Optimization and Lagrange Multiplier Methods*. Academic Press, 1982.

[15] A. M. Bloch, P. S. Krishnaprasad, J. E. Marsden, and R. M. Murray. Nonholonomic mechanical systems and symmetry. *Archive for Rational Mechanics and Symmetry*, 1996.

[16] A. M. Bloch, M. Reyhanoglu, and N. H. McClamroch. Control and stabilization of nonholonomic dynamic systems. *IEEE Transactions on Automatic Control*, 37(11), pp. 1746–1756, 1992.

[17] R. W. Brockett. Control theory and singular Riemannian geometry. In *New Directions in Applied Mathematics*, pp. 11–27, Springer-Verlag, New York, 1981.

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

[19] R. A. Brooks and T. Lozano-Perez. A subdivision algorithm in configuration space for find-path with rotation. *Eighth International Joint Conference on Artificial Intelligence,* pp. 799–806, Karlsruhe, Germany, 1983.

[20] A. E. Bryson, W. F. Denham, and S. E. Dreyfus. Optimal programming problems with inequality constraints I: Necessary conditions for extremal solutions. *AIAA Journal*, 1(11), pp. 2544–2550, 1963.

[21] A. E. Bryson and Y. Ho. *Applied Optimal Control*. Hempshire Publishing Co., New York, second edition, 1975.

[22] C. E. Buckley. *The Application of Continuum Methods to Path Planning*. PhD thesis, Stanford University, Palo Alto, CA, August 1985.

[23] L. G. Bushnell, D. M. Tilbury, and S. S. Sastry. Steering three-input nonholonomic systems: The fire truck example. *The International Journal of Robotics Research*, 14(4), pp. 366–381, 1995.

[24] J. F. Canny. *The complexity of robot motion planning*. PhD thesis, MIT, 1988. MIT Press, Cambridge.

[25] H. Choset and J. W. Burdick. Sensor based motion planning: The hierarchial generalized voronoi graph. In J. P. Laumond and M. Overmars, editors, *Workshop on Algorithmic Foundations of Robotics*, pp. 47–62, A. K. Peters, 1996.

[26] C. C. de Wit and O. J. Sordalen. Exponential stabilization of mobile robots with nonholonomic constraints. *IEEE Transactions on Automatic Control*, 37(11), pp. 1791–1797, 1992.

[27] A. P. del Pobil and M. A. Serna. *Spatial representation and motion planning*. Lecture notes in Computer Science, Springer-Verlag, New York, 1995.

[28] N. Deo. *Graph theory with applications to engineering and computer science*. Prentice-Hall Inc., 1974.

[29] J. P. Desai and V. Kumar. Optimal motion plans for non-holonomic mobile robots amidst obstacles. Technical report, University of Pennsylvania, GRASP Laboratory, Philadelphia, February 1996.

[30] J. P. Desai and V. Kumar. Experimental investigation of planar two-arm friction assisted grasping tasks. In *Sixth Annual Symposium on Haptics Interfaces for Virtual Environment and Teleoperator Systems*, Dallas, Texas, USA, November 1997.

[31] J. P. Desai and V. Kumar. Nonholonomic motion planning for multiple mobile manipulators. In *International Conference on Robotics and Automation*, Albuquerque, NM, April 1997.

[32] J. P. Desai, J. P. Ostrowski, and V. Kumar. Controlling formations of multiple mobile robots. In *Proceedings of 1998 IEEE International Conference on Robotics and Automation*, Leuven, Belgium, May 1998.

[33] J. P. Desai, C. C. Wang, M. Žefran, and V. Kumar. Motion planning for multiple mobile manipulators. In *Proceedings of 1996 IEEE International Conference on Robotics and Automation*, Minneapolis, MN, 1996.

[34] J. P. Desai, M. Žefran, and V. Kumar. Two-arm manipulation tasks with friction assisted grasping. In *International Conference on Intelligent Robots and Systems*, Grenoble, France, September 1997.

[35] J. P. Desai, M. Žefran, and V. Kumar. Two arm manipulation tasks with friction assisted grasping. *The Advanced Robotics Journal*, 13(4), 1998.

[36] B. R. Donald. A search algorithm for motion planning with six degrees of freedom. *Artificial Intelligence*, 31, pp. 295–353, 1987.

[37] L. E. Dubins. On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79, pp. 497–516, 1957.

[38] O. Egeland, E. Berglund, and O. J. Sordalen. Exponential stabilization of a nonholonomic underwater vehicle with constant desired configuration. In *Proceedings of 1994 IEEE International Conference on Robotics and Automation*, pp. 20–25, San Diego, CA, May 1994.

[39] M. Erdmann. An exploration of nonprehensile two-palm manipulation: Planning and execution. In *Seventh International Symposium on Robotics Research*, October 1995.

[40] B. Faverjon and P. Tournassoud. A local based approach for path planning of manipulators with a high number of degrees of freedom. In *Proceedings of 1987 International Conference on Robotics and Automation*, pp. 1152–1159, 1987.

[41] C. Fernandes, L. Gurvits, and Z. X. Li. Foundations of nonholonomic motion planning. Technical Report 577, Courant Institute of Mathematical Sciences, New York, August 1991.

[42] J. R. Flanagan, J. Tresilian, and A. M. Wing. Coupling of grip force and load force during arm movements with grasped objects. *Neuroscience letters*, 152, pp. 53–56, 1993.

[43] T. Flash and E. A. Henis. Arm trajectory modifications during reaching towards visual targets. *Journal of Cognitive Neuroscience*, 3(3), pp. 220–230, 1991.

[44] T. Flash and N. Hogan. The coordination of arm movements: An experimentally confirmed mathematical model. *The Journal of Neuroscience*, 5(7), pp. 1688–1703, 1985.

[45] G. J. Garvin, M. Žefran, E. A. Henis, and V. Kumar. Two-arm trajectory planning in a manipulation task. *Biological Cybernetics*, 76, pp. 53–62, 1997.

[46] E. G. Gilbert and D. W. Johnson. Distance functions and their application to robot path planning in the presence of obstacles. *IEEE Journal of Robotics and Automation*, RA-1(1), pp. 21–30, March 1985.

[47] E. G. Gilbert, D. W. Johnson, and S. S. Keerthi. A fast procedure for computing the distance between complex objects in three-dimensional space. *IEEE Journal of Robotics and Automation*, 4(2), pp. 193–203, 1988.

[48] A. M. Gordon, H. Forssberg, R. S. Johansson, and G. Westling. The integration of haptically acquired size information in the programming of precision grip. *Experimental Brain Research*, 83, pp. 483–488, 1991.

[49] L. Gouzenes. Strategies for solving collision-free trajectories problems for mobile and manipulator robots. *International Journal of Robotics Research*, 3(4), pp. 51–65, 1984.

[50] J. Gregory and C. Lin. *Constrained Optimization in the Calculus of Variations and Optimal Control Theory*. Van Nostrand Reinhold, New York, 1992.

[51] Y. Guiard. Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. *Journal of Motor Behavior*, 19(4), pp. 486–517, 1987.

[52] F. Harary and E. M. Palmer. *Graphical enumeration*. Academic press, 1973.

[53] E. A. Henis, T. Stanford, and D. Sparks. A model for the generation of modified eye trajectories. *American Society for Neuroscience Abstracts*, April 1992.

[54] I. N. Herstein. *Topics in Algebra*. John Wiley and Sons, 1973.

[55] M. R. Hestenes. *Calculus of variations and optimal control theory*. John, Wiley and Sons., 1966.

[56] F. B. Hildebrand. *Advanced Calculus for Applications*. Prentice-Hall, Inc., Englewood Cliffs, NJ, second edition, 1976.

[57] B. Hoff and M. A. Arbib. Models of trajectory formation and temporal interaction of reach and grasp. *Journal of Motor Behavior*, 25(3), pp. 175–192, 1993.

[58] N. Hogan. Control and coordination of voluntary arm movements. In *Proceedings of the 1982 American Control Conference*, pp. 522–528, 1982.

[59] N. Hogan. An organizing principle for a class of voluntary movements. *The Journal of Neuroscience*, 4(11), pp. 2745–2754, November 1984.

[60] N. Hogan. Impedance control: An approach to manipulation: Part I. *Transactions of ASME, Journal of Dyamic Systems, Measurement, and Control*, 107(1), pp. 1–7, March 1985.

[61] P. Jacobs and J. Canny. Planning smooth paths for mobile robots. In *Proceedings of 1989 International Conference on Robotics and Automation*, pp. 2–7, 1989.

[62] D. H. Jacobsen and M. M. Lele. A transformation technique for optimal control problems with a state variable inequality constraint. *IEEE Transactions on Automatic Control*, 14(5), pp. 457–464, October 1969.

[63] D. H. Jacobsen, M. M. Lele, and J. L. Speyer. New necessary conditions of optimality for control problems with state-variable inequality constraints. *Journal of Mathematical Analysis and Applications*, 35, pp. 255–284, 1971.

[64] R. S. Johansson, R. Riso, C. Hager, and L. Backstrom. Somatosensory control of precision grip during unpredictable pulling loads. *Experimental Brain Research*, 89, pp. 181–191, 1992.

[65] R. S. Johansson and G. Westling. Roles of glabrous skin receptors and sensorimotor memory in automatic control of precision grip when lifting rougher or more slippery objects. *Experimental Brain Research*, 56, pp. 550–564, 1984.

[66] R. S. Johansson and G. Westling. Coordinated isometric muscle commands adequately and erroneously programmed for the weight during lifting tasks with precision grip. *Experimental Brain Research*, 71, pp. 59–71, 1988.

[67] L. Kavraki, P. Švestka, J. C. Latombe, and M. H. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration space. *IEEE Transactions on Robotics and Automation*, 12(4), pp. 566–580, 1996.

[68] S. D. Kelly and R. M. Murray. Geometric phases and locomotion. *Journal of Robotic Systems*, 12(6), pp. 417–431, June 1995.

[69] O. Khatib. *Commande Dynamique dans l'Espace Opérationnel des Robots Manipulateurs en Présence d'Obstacles*. PhD thesis, Ecole Nationale Supérieure de l'Aéronautique et de l'Espace, Toulouse, France, 1980.

[70] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research*, 5(1), pp. 90–99, 1986.

[71] D. E. Koditschek. Exact robot navigation by means of potential functions: Some topological considerations. In *Proceedings of 1987 International Conference on Robotics and Automation*, pp. 1–6, Raleigh, NC, 1987.

[72] W. Koon and J. E. Marsden. Optimal control for holonomic and nonholonomic mechanical systems with symmetry and Lagrangian reduction. Technical Report CIT/CDS 95-022, California Institute of Technology, 1995.

[73] K. Kosugue, T. Oosumi, and K. Chiba. Load sharing of decentralized controlled multiple mobile robots handling a single object. In *Proceedings of 1997 IEEE International Conference on Robotics and Automation,* Albuquerque, New Mexico, April 1997.

[74] K. Kreutz and A. Lokshin. Load balance and closed chain multiple arm control. In *Proceedings 1988 American Control Conference*, Atlanta, GA, 1988.

[75] P. S. Krishnaprasad and D. P. Tsakiris. G-snakes: Nonholonomic kinematic chains on Lie groups. Technical Report ISR - T.R. 94-27, Institute for Systems Research, University of Maryland, 1994.

[76] J.C. Latombe. *Robot motion planning*. Kluwer Academic Publishers, Boston, Massachussetts, 1991.

[77] J.-P. Laumond. Finding collision free smooth trajectories for a nonholonomic mobile robot. In *Proceedings of the 10th International Joint Conference on Artificial Intelligence,* pp. 1120–1123, Milan, Italy, 1987.

[78] J.-P. Laumond, P. E. Jacobs, M. Taix, and R. M. Murray. A motion planner for nonholonomic mobile robots. *IEEE Transactions on Robotics and Automation*, 10(5), pp. 577–593, 1994.

[79] A. Lewis, J. P. Ostrowski, R. Murray, and J. W. Burdick. Nonholonomic mechanics and locomotion: The snakeboard example. In *1994 IEEE International Conference on robotics and Automation*, 1994.

[80] Z. Li and J. Cannny(editors). *Nonholonomic motion planning*. Kluwer Academic Publishers, 1993.

[81] M. C. Lin and J. F. Canny. A fast algorithm for incremental distance calculation. In *Proceedings of 1991 International Conference on Robotics and Automation*, pp. 1008–1014, Sacramento, CA, 1991.

[82] T. Lozano-Perez. Spatial planning: A configuration space approach. *IEEE Transactions on Computers*, C-32(2), pp. 108–120, 1983.

[83] T. Lozano-Perez. A simple motion planning algorithm for general robot manipulators. *IEEE Journal of Robotics and Automation*, 3(3), pp. 224–238, June 1987.

[84] P. S. Lum, D. J. Reinkensmeyer, S. L. Lehman, P. Y. Li, and L. W. Stark. Feedforward stabilization in a bimanual unloading task. *Experimental Brain Research*, 89, pp. 172–180, 1992.

[85] V. Lumelsky, S. Mukhopadhyay, and K. Sun. Dynamic path planning in sensor-based terrain acquisition. *IEEE Transactions on Robotics and Automation*, 6(4), pp. 462–472, August 1990.

[86] J. Lygeros, D. N. Godbole, and S. S. Sastry. A verified hybrid controller for automated vehicles. *IEEE Transactions on Automatic Control*, 43(4), pp. 522–539, 1998.

[87] K. Lynch and M. Mason. Stable pushing: Mechanics, controllability, and planning. In *Workshop on Algorithmic Foundations of Robotics*, pp. 239–262. A.K. Peters, Boston, MA, 1995.

[88] V. Manikonda, J. Hendler, and P. S. Krishnaprasad. Formalizing behavior-based planning for nonholonomic robots. In *International Joint Conference on Artificial Intelligence*, 2, Montreal, Canada, 1995.

[89] M. Mataric. Reinforcement learning in the multi-robot domain. *Autonomous Robots*, 4(1), January 1997.

[90] J. McIntyre, F. A. Mussa-Ivaldi, and E. Bizzi. Modeling of multi-joint motor systems. In *Proceedings of the IEEE Engineering in Medicine & Biology Society 11th Annual International Conference*, 1989.

[91] P. Morasso. Spatial control of arm movements. *Experimental Brain Research*, 42, pp. 223–227, 1981.

[92] R. M. Murray, Z. Li, and S. S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.

[93] R. M. Murray and S. S. Sastry. Nonholonomic motion planning: Steering using sinusoids. *IEEE Transactions on Automatic Control*, 38(5), pp. 700–716, 1993.

[94] L. Noakes, G. Heinzinger, and B. Paden. Cubic splines on curved spaces. *IMA Journal of Mathematical Control & Information*, 6, pp. 465–473, 1989.

[95] J. P. Ostrowski. *The Mechanics and Control of Undulatory Robotic Locomotion*. PhD thesis, California Institute of Technology, 1995.

[96] J. P. Ostrowski and J. W. Burdick. Geometric perspectives on the mechanics and control of robotic locomotion. In *Preprints of the 7th International Symposium on Robotics Research*, Herrsching, Germany, 1995.

[97] J. P. Ostrowski and J. W. Burdick. The geometric mechanics of undulatory robotic locomotion. *International Journal of Robotics Research*, July 1997.

[98] J. P. Ostrowski, J. P. Desai, and V. Kumar. Optimal gait selection for nonholonomic locomotion systems. In *International Conference on Robotics and Automation*, Albuquerque, NM, April 1997.

[99] E. Paljug, X. Yun, and V. Kumar. Control of rolling contacts in multi-arm manipulation. *IEEE Transactions on Robotics and Automation*, 10(4), 1994.

[100] F. C. Park and R. W. Brockett. Kinematic dexterity of robotic mechanisms. *International Journal of Robotics Research*, 13(1), pp. 1–15, 1994.

[101] L. E. Parker. The effect of action recognition and robot awareness in cooperative robotic teams. In *Proceedings of the 1995 International Conference on Intelligent Robots and Systems*, 1995.

[102] D. Pawluk and R. D. Howe. Dynamic contact mechanics of the human fingerpad, Part II: Distributed response. Technical Report 96-004, Harvard University, December 1996.

[103] F. G. Pin, K. A. Morgansen, F. A. Tulloch, C. J. Hacker, and K. B. Gower. Motion planning for mobile manipulators with a nonholonomic constraint using the FSP (Full space parametrization) method. *Journal of Robotic Systems.*, 13(11), pp. 723–737, November 1996.

[104] J. A. Reeds and L. A. Shepp. Optimal paths for a car that goes both forwards and backwards. *Pacific Journal of Mathematics*, 145(2), pp. 367–393, 1990.

[105] D. J. Reinkensmeyer, P. S. Lum, and S. L. Lehman. Human control of a simple two-hand grasp. *Biological Cybernetics*, 67, pp. 553–564, 1992.

[106] E. Rimon and D. E. Koditschek. Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation*, 8(5), pp. 501–518, 1992.

[107] H. Sagan. *Introduction to the calculus of variations*. McGraw-Hill, New York, 1969.

[108] C. Samson. Time-varying feedback stabilization of car-like wheeled mobile robots. *The International Journal of Robotics Research*, 12(1), pp. 55–64, 1993.

[109] C. Samson and K. Ait-Abderrahim. Feedback control of a nonholonomic wheeled cart in cartesian space. In *Proceedings of 1991 International Conference on Robotics and Automation*, pp. 1136–1141, Sacramento, CA, April 1991.

[110] J. T. Schwartz and M. Sharir. On the piano movers problem I: The case of two-dimensional rigid polygonal body moving amidst polygonal barriers. *Communications on pure and applied mathematics*, 36, pp. 345–398, 1983.

[111] J.T. Schwartz and M. Sharir. On the piano movers problem problem II: General techniques for computing topological properties of real algebraic manifolds. *Advances in Applied Mathematics*, 4, pp. 298–351, 1983.

[112] Z. Shiller. Time-energy optimal control of articulated systems with geometric path constraints. In *Proceedings of 1994 IEEE International Conference on Robotics and Automation*, pp. 2680–2685, San Diego, CA, 1994.

[113] J. F. Soechting and F. Lacquaniti. Invariant characteristics of a pointing movement in man. *The Journal of Neuroscience*, 1(7), pp. 710–720, 1981.

[114] J. F. Soechting and F. Lacquaniti. Modification of trajectory of a pointing movement in response to a change in target location. *The Journal of Neurophysiology*, 49, pp. 548–564, 1983.

[115] J. L. Speyer and A. E. Bryson. Optimal programming problems with a bounded state space. *AIAA Journal*, 6(8), pp. 1488–1491, 1968.

[116] M. A. Srinivasan and J. Chen. Human performance in controlling normal forces of contact with rigid objects. *Advances in Robotics, Mechatronics and Haptic Interfaces*, 49, pp. 119–125, 1993.

[117] T. Sugar and V. Kumar. Decentralized control of cooperating manipulators. In *Proceedings of 1998 IEEE International Conference on Robotics and Automation*, Leuven, Belgium, May 1998.

[118] H. J. Sussmann and W. Liu. Lie bracket extensions and averaging: The single-bracket case. In Z. Li and J. F. Canny, editors, *Nonholonomic motion planning*, pp. 109–148. Kluwer Academic Publishers, Boston, 1993.

[119] D. Tilbury, R. M. Murray, and S. S. Sastry. Trajectory generation for the n-trailer problem using Goursat normal form. *IEEE Transactions on Automatic Control*, 40(5), pp. 802–819, 1995.

[120] S. Udapa. Collision detection and avoidance in cmputer controlled manipulators. In *Proceedings of Fifth International Joint Conference on Artificial Intelligence*, 1977. MIT.

[121] Y. Uno, M. Kawato, and R. Suzuki. Formation and control of optimal trajectory in human multijoint arm movement. *Biological Cybernetics*, 61, pp. 89–101, 1989.

[122] F. A. Valentine. The problem of lagrange with differential inequalities as added side conditions. In *Contributions to the Calculus of Variations, 1933-1937*, pp. 407–448, University of Chicago Press, 1937.

[123] P. Varaiya. Smart cars on smart roads: Problems of control. *IEEE Transactions on Automatic Control*, 38, pp. 195–207, February 1993.

[124] P. Viviani and C. Terzuolo. Space-time invariance in motor skills. In G. E. Stelmach and J. Requin, editors, *Tutorials in Motor Behavior*, pp. 525–533, Elsevier-North-Holland Publishing Co., Amsterdam, 1980.

[125] P. Viviani and C. Terzuolo. Trajectory determines movement dynamics. *Neuroscience*, 7, pp. 431–437, 1982.

[126] G. Walsh, D. Tilbury, S. Sastry, R. Murray, and J. P. Laumond. Stabilization of trajectories for systems with nonholonomic constraints. *IEEE Transactions on Automatic Control*, 39(1), pp. 216–222, 1994.

[127] G. C. Walsh and S. S. Sastry. On reorienting linked rigid bodies using internal motions. *IEEE Transactions on Robotics and Automation*, 11(1), pp. 139–146, 1995.

[128] C. Wang and V. Kumar. Rate kinematics for mobile manipulators. In *1993 IEEE International Conference on Robotics and Automation*, pp. 713–718, Atlanta, GA, May 1993.

[129] D. Williams and O. Khatib. The virtual linkage: A model for internal forces in multi-grasp manipulation. In *Proceedings of 1993 IEEE International Conference on Robotics and Automation*, Atlanta, GA, 1993.

[130] D. A. Winter. *Biomechanics and Motor Control of Human Movement*. John Wiley and Sons, Inc., New York, 1990.

[131] Y. Yamamoto. *Control and coordination of locomotion and manipulation of a wheeled mobile manipulator*. PhD thesis, University of Pennsylvania, GRASP Laboratory, May 1994.

[132] Y. Yamamoto and X. Yun. Coordinating locomotion and manipulation of a mobile manipulator. In *Proceedings of the 31th IEEE Conference on Decision and Control*, Tucson, Arizona, December 1992.

[133] X. Yun. Object handling using two arms without grasping. *The International Journal of Robotics Research*, 12(1), pp. 99–106, February 1993.

[134] X. Yun and V. Kumar. An approach to simultaneous control of trajectory and interaction forces in dual-arm configurations. *IEEE Transactions on Robotics and Automation*, 7(5), pp. 618–625, 1991.

[135] M. Žefran. *Continuous Methods for Motion Planning*. PhD thesis, University of Pennsylvania, Philadelphia, PA, October 1996.

[136] M. Žefran, J. P. Desai, and V. Kumar. Continuous motion plans for robotic systems with changing dynamic behavior. In *Workshop on Algorithmic Foundations of Robotics (WAFR) '96*, Toulouse, France, July 1996.

[137] M. Žefran and V. Kumar. Optimal control of systems with unilateral constraints. In *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, Nagoya, Japan, 1995.

[138] M. Žefran, V. Kumar, J. P. Desai, and E. Henis. Two-arm manipulation: What can we learn by studying humans? In *1995 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Pittsburgh, PA, August 1995.

[139] M. Žefran, V. Kumar, G. Garvin, and J. P. Desai. Experimental investigation of human two-arm manipulation. In *International conference on Mechanics in Medicine and Biology*, 1996.

[140] M. Žefran, V. Kumar, and X. Yun. Optimal trajectories and force distribution for cooperating arms. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, San Diego, 1994.

[141] H. Zhang, V. Kumar, and J. Ostrowski. Motion planning with uncertainty. In *Proceedings of 1998 IEEE International Conference on Robotics and Automation*, Leuven, Belgium, May 1998.

[142] Y. Zheng and J.Y.S. Luh. Optimal load distribution for two industrial robots handling a single object. In *IEEE International Conference on Robotics and Automation*, Philadelphia, PA, April 1988.