GEODESIC PROBLEMS FOR MOBILE ROBOTS

BY

HAMID REZA CHITSAZ

M.S., University of Illinois at Urbana-Champaign, 2006
B.S., Sharif University of Technology, 2001

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2008

Urbana, Illinois

Doctoral Committee:

Associate Professor Steven M. LaValle, Chair
Associate Professor Jeff Erickson
Assistant Professor Anil N. Hirani
Assistant Professor Devin J. Balkcom, Dartmouth

Abstract

As mobile robots operate with limited resources which they carry onboard in large obstructed environments, their success is dependent on how efficiently they move while they avoid collision with obstacles and other robots. Moving optimally is the ultimate efficiency a mobile robot can achieve. Therefore, planning optimal motions and devising optimal coordination strategies are two important and challenging fundamental problems in mobile robotics, which have received significant attention in the last couple of decades. Both of those problems can be reduced to shortest path, or equivalently *geodesic*, problems in appropriate geometric settings. Geodesic problems have been studied in two disciplines: 1) optimal control theory, and 2) computational geometry. Optimal control theory has focused on the differential constraints of robotic systems, while computational geometry has focused on shortest path problems in an environment with obstacles. Optimal control theory has historically disregarded obstacles in the environment, and computational geometry does not consider dynamics of the robotic system, various optimality criteria, or multi-objective optimality. While each discipline has its own powerful tools to address some geodesic problems, there is a large class of problems that cannot be solved using existing algorithms and methods. We introduce a unified approach that is inspired by main results in both disciplines. In this dissertation, we demonstrate our technique, which combines the celebrated Pontryagin Maximum Principle from optimal control theory with visibility graph methods from computational geometry, by solving three geodesic problems for mobile robots: 1) geodesics for the differential drive among obstacles, 2) geodesics for a kinematic airplane, and 3) optimal coordination of two polygonal robots moving on a predetermined network of paths.

We consider the differential drive because it is ubiquitous in mobile robotics. To obtain a well-defined notion of shortest, the total amount of wheel rotation is optimized. We analytically characterize minimum wheel-rotation trajectories in the absence of obstacles, and identify 52 different minimum wheel-rotation trajectories. In the presence of obstacles, every minimum wheel-rotation trajectory is composed of two kinds of subtrajectories: on the boundary of obstacles and in the interior of collision-free space. We prove those subtrajectories that lie in the interior of collision-free space are tangent to the obstacles at both ends. The bitangency condition yields a *nonholonomic bitangency graph* which is a network of collision-free trajectories in which the solution is sought. In general, our nonholonomic bitangency graph is a 2-dimensional subset of the 3-dimensional configuration space of the robot. Therefore, further optimization or a continuous search may be required to answer queries. However if obstacles are circular and far enough from one another, the graph is a 1-dimensional subset of the configuration space and any graph search algorithm, such as Dijkstra's algorithm, extracts the solution. In the second problem, we introduce a new kinematic airplane model. Our airplane is a natural extention of the Dubins car [56], and extends it with an additional configuration variable for the altitude. We use the Pontryagin Maximum Principle to analytically characterize geodesics for it. To obtain a notion of shortest, time is optimized. Finally, we present an algorithm for computing optimal coordination of two polygonal robots without differential constraints. Each robot has a reference point that must lie on a network of paths called a roadmap. Each robot wants to move from its given initial location to its goal location without colliding with the other one. Rather than impose an a priori cost scalarization for choosing the best combined motion, we consider finding motions whose cost vectors are Pareto-optimal. Pareto-optimal coordination strategies are the ones for which there exists no strategy that would be better for both robots. The problem is equivalent to computing geodesics in the coordination space which is the Cartesian product of the roadmap with itself. We extend visibility graphs to solve the problem. If the roadmap is acyclic, then our algorithm has $O(mn^2 \log n)$ time complexity, in which m is the number of paths in the roadmap, and n is the number of coordination space vertices. For cyclic roadmaps, our algorithm computes solutions in time $O(2^{5\alpha} m^{1+5\alpha} n^2 \log(m^{2\alpha} n))$, in which $\alpha = 1 + \lceil (5\ell + r)/b \rceil$ where ℓ is total length of the

roadmap, r is total length of coordination space obstacle boundary, and b is the length of the shortest edge in the roadmap.

To my father and my mother, who were my first teachers,

and sources of unconditional love.

To my brothers and sisters whom I adore and cherish.

Last but not least, to my teachers and professors since elementary school.

Acknowledgments

This dissertation would not have been possible without the support of many people. Many thanks to my adviser, Steven M. LaValle, who provided me with unfaltering encouragement and help along the way. I also gratefully acknowledge valuable guidance offered by my committee members, Devin J. Balkcom, Jeff Erickson, and Anil N. Hirani. I thank Matthew T. Mason and Jason M. O'Kane for collaborating with me. I would like to thank those Motion Strategy Laboratory colleagues with whom I had the honor to work: Peng Cheng, Stephen Lindemann, Kamilah Taylor, Benjamin Tovar, Anna Yershova, and Jingjin Yu. Financial support for this work was provided in part by NSF under award 0528086 (MSPA-MCS).

More personally, I am grateful for the support I have received from my father Alireza Chitsaz, my mother Mahvash Chitsaz, and my brothers and sisters Mahsa, Ehsan, Mohsen, Maedeh, and Mehrnaz. Thanks to all my other friends whose friendship is very valuable for me, to name a few: Abbas Aminmansour, Amir Behgooy, Ehsan Chiniforooshan, Mohsen Dadfarnia, Salem Derisavi, Reza Etebari, Ali Farhadi, Zahra Golshani, MohammadTaghi Hajiaghayi, Moslem Kazemi, Faezeh Koohestani, Arash Mahdian, Vahab S. Mirrokni, Mahdi Mirzazadeh, Mahdi Rastad, Bashir Sadjad, Ali Toossi, and Saman Aliari Zonouz.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Mobile robot technology is expected to have deep near-term impact in our society after having seen a long period of research. iRobot's Roomba, an autonomous home vacuum cleaner, and RoboMower, an automatic lawn mower, are instances of autonomous mobile robots which have already entered home zone. Honda's Asimo, a sophisticated humanoid robot, is expected to be robust, fast, and autonomous enough to be deployed as a home assistant in near future[1]. We may soon see humanoid robots provide assistance and companionship to the elderly. In the industry zone, Kiva Mobile Fulfillment System (Kiva MFS) uses a breakthrough new approach to order fulfillment in a commercial distribution center. With the Kiva MFS, operators stand still while the products stored on inventory pods are picked up and brought to them by a fleet of mobile robotic drive units[2]. In addition, prebuilt research mobile robotic platforms such as the Pioneer[3] and Khepera[4] are increasingly deployed in experimental robotics research.

Mobile robots usually operate with limited resources which they can carry onboard in a large obstructed environment. Most likely, there are other agents such as humans and other robots in the environment as well. A perfect example is the Kiva MFS in which several mobile robotic drive units carry pods around a large inventory without colliding with stationary pods or each other. For the system to be applicable, robots have to avoid obstacles and coordinate their motion with one another. Time sensitivity of their mission

[1] http://asimo.honda.com

[2] http://www.kivasystems.com/

[3] http://www.mobilerobots.com/

[4] http://www.k-team.com/

and also limited power they can carry onboard makes it essential for mobile robots to move around optimally. As a result, planning optimal collision-free motion for mobile robots has been one of the most important and challenging problems in robotics.

This problem has been studied in two different disciplines: optimal control theory and computational geometry. Each discipline deals with a specific category of optimal motion planning problems for mobile robots. Optimal control theory, a generalization of the calculus of variations [65], is a mathematical optimization method [118] for deriving control policies [21, 33]. The method is largely due to the work of Lev S. Pontryagin and his collaborators, summarized in English in [141]. Optimal control theory deals with the problem of finding a control law for a given system such that a certain optimality criterion is achieved. The optimal control can be derived using the Pontryagin Maximum Principle which is a necessary condition [141], or by solving the Hamilton-Jacobi-Bellman equation which is a sufficient condition [18]. It is worth mentioning that optimal control is closely related to sub-Riemannian-Finslerian geometry depending on the properties of the system [23, 92, 115, 127]. In geometric settings, an optimal path is called a shortest path or a *geodesic*. In this dissertation, we broadly adopt that usage for 'geodesic'. Computational geometry is the study of algorithms to solve problems stated in terms of geometry [52]. For instance, the following is a problem studied in computational geometry: given a set of polyhedral obstacles in a Euclidean space, having a total of n vertices, design algorithms for efficient (exact or approximate) calculation of a shortest, obstacle-avoiding path connecting any two query points.

Besides analytical tools, numerical methods have been developed to compute optimal paths. For instance, level set methods have proven to be successful in numerically solving Hamilton-Jacobi equation in various applications [132, 156]. As it was coined by Bellman, those methods suffer from the curse of dimensionality [18]. In a philosophically different approach, Discrete Mechanics and Optimal Control offers a unified framework for designing numerical symplectic integrators with desired precision [84, 120]. Symplectic integration methods can be applied to high-dimensional systems, but they give only local optimal solutions. Therefore, analytical characterization of optimal paths is a challenging but extremely rewarding approach for solving optimal path problems.

Historically, optimal control for linear systems has been profoundly studied and almost thoroughly resolved, because it is very common in control theory to model reality as a linear system. Mobile robots are highly nonlinear systems, in which case linear optimal control results are irrelevant. On the other hand, nonlinear optimal control tools have focused on the differential constraints of the system rather than work space obstacles. Computational geometry has mainly focused on shortest paths for a point robot, often without differential constraints, in a cluttered work space. As a result, a large class of optimal motion planning problems for mobile robots cannot be solved by existing methods. In this dissertation, we present a unifying novel approach, based on nonlinear optimal control tools and some computational geometry techniques, to planning optimal motions for mobile robots. We demonstrate our approach by considering three mobile robot mechanisms: the differential drive which is ubiquitous in mobile robotics, an aircraft model, and polygonal robots without differential constraints. Our complete mathematical characterization of shortest paths for those mechanisms is also helpful in mechanism design, computing a nonholonomic metric for motion planning algorithms, and building a local motion planner. We expect our work to have significant practical impact in broad areas such as mobile robotics, autonomous vehicles control, airtraffic control, autonomous inventory management, and computer animation. Our results can be used to optimize a class of task specifications for mobile platforms. Our characterization of shortest paths will have direct practical impact in aerial vehicles, home robots such as iRobot, manufacturing mobile robots, and automated inventory systems such as Kiva MFS.

Another motivation for this work emanates from the fact that one can simplify the control and planning problem, usually in the presence of obstacles, by piecing together a set of elementary trajectories chosen from a library. Such pieces of trajectories that can be combined sequentially to produce more complicated trajectories are called *motion primitives* [19, 64, 73]. They may even be computed and stored offline, particularly when there are symmetries, to yield speedup in online motion planning applications such as computer games. Many motion planning approaches have relied on good motion primitives. Finding suitable motion primitives for a robot is an area of recent, active research. One approach is using the optimal trajectories as motion primitives. For instance, Latombe successfully used shortest paths in a fast path planner for an indoor mobile robot among obstacles [98]. There are many other successful examples of using optimal paths as motion primitives [6, 64, 122]. In this dissertation, our characterization of geodesics for mobile robots yields sets of motion primitives for the demonstrated mechanisms.

1.2 Related Work

1.2.1 General Motion Planning

Motion planning as a field was started by the introduction of configuration spaces by Lozano-Pérez [116, 117]. A robot usually works in a 2D or 3D environment with obstacles, which is called the work space. Lozano-Pérez suggested that one can add a layer of abstraction by associating any motion of the robot with a path in the set of feasible distinct robot configurations. There is a natural correspondence between work space obstacles and obstacle regions in the configuration space. Reif showed that motion planning is PSPACE-hard [144]. Schwartz and Sharir gave the first complete motion planning algorithm for a rigid body in two and three dimensions [153, 154, 155]. Their algorithm is based on geometric methods, specifically Collins decomposition [50]. The running time of Schwartz-Sharir algorithm is doubly-exponential in the dimension of the configuration space. There have been sporadic motion planning algorithms given for specific systems as well [31, 55]. Canny proposed the roadmap algorithm, which is a singly-exponential general motion planning algorithm based on Morse theory and resultants in commutative algebra [37]. Since the problem was shown to be PSPACE-hard, Canny's algorithm was regarded as a theoretical bottom-line at that time. Emiris later gave an improved algorithm for computing resultants that is interesting in its own right [59]. Recently, Canny's algorithm was polished and improved by Basu et al. [16, 17]. Complete general motion planning algorithms, namely Schwartz-Sharir and Canny's, are based on real algebraic geometry algorithms which are extremely difficult to implement. Moreover, they completely ignore differential constraints. Therefore, potential field and sampling-based methods emerged in the 1990s to address practical motion planning problems.

Discretization and grid search were among the first attempts along those lines [40, 91, 107]. The first sampling-based motion planning algorithm to gain significant popularity was the Randomized Path Planner of Barraquand and Latombe [14]. Probabilistic roadmaps (PRMs) [87] and Rapidly-exploring Random Trees (RRTs) [102, 105] are continuations of that trend. There are other variations of sampling-based motion planning algorithms [80, 97, 112, 121]. Probabilistic roadmaps sample the configuration space and attempt to connect nearby free samples with a collision-free path computed by a local planner. Rapidly-exploring Random Trees explore the configuration space by building a search tree. PRMs have been modified to give a significant number of extensions and applications [8, 9, 24, 28, 34, 35, 90, 96, 108, 139, 140, 157, 167, 175, 177]. RRTs have also generated a diverse set of applications and variants [22, 29, 32, 41, 48, 61, 63, 83, 86, 89, 109, 110, 111, 148, 169, 177, 179]. Path planning algorithms based on potential fields follow the gradient of a potential or navigation function, which usually combines a term attractive to the goal state with terms repulsive to the obstacles [74, 88, 94, 113, 114, 147, 166, 172]. Potential fields and its variants usually suffer from local minima. For a complete exposition to such methods, see [99, 103].

1.2.2 Motion Planning for Mobile Robots

Mobile robots are typical examples of nonholonomic systems, which are characterized by constraint equations containing the time derivatives of the system configuration variables. The equations are non-integrable and typically arise when the system has less controls than configuration variables. For example, a differential drive robot has two controls and three configuration variables [13]. As a result, any path in the configuration space does not necessarily correspond to a feasible motion for the robot. Even in the absence of obstacles, planning nonholonomic motion is not an easy task. As of writing this dissertation, there is no general algorithm to plan motions for any nonholonomic system so that the system is guaranteed to exactly reach a given goal. Existing results are general approximate methods, and exact methods for a class of nonholonomic systems that includes car-like robots [101].

1.2.3 Motion Primitives

Many general motion planning methods have relied on good motion primitives, which are pieces of trajectories that can be concatenated in an appropriate way to yield a solution. Latombe successfully used Reeds-Shepp curves in a fast path planner for an indoor mobile robot among obstacles [98]. Conner et al. used a set of continuous local feedback control policies and a discrete automaton to plan verifiably correct motions for a mobile robot in a changing environment [51]. Mehta and Egerstedt used optimal control for constructing control programs from a given collection of motion primitives, and also for augmenting the motion primitive set [122]. Frazzoli et al. proposed a set of motion primitives, for a six-dimensional aircraft, which contains pieces of optimal trajectories called trim trajectories [64]. Shortest paths are interesting because they provide optimal motions and also they are used in many applications as sets of motion primitives. We now focus on optimal-motion planning methods for mobile robots.

1.2.4 Nonlinear Optimal Control

Nonholonomic shortest path problems, in the absence of obstacles, have been studied for many mobile robots [12, 13, 49, 56, 142, 160, 161, 163, 164, 174]. The first work on shortest paths for car-like vehicles was done by Dubins [56], who gave a characterization of time-optimal trajectories for a car with a bounded turn radius that always moves forward with constant speed. Dubins used a purely geometrical method to characterize shortest paths in this setting. Later, Reeds and Shepp solved a similar problem in which the car is able to move backward as well [142]. They identified 48 different classes of paths such that between any pair of configurations there is a shortest path in one of those classes. Shortly after Reeds and Shepp, their problem was solved and also refined by Sussmann and Tang [164] and by Boissonnat, Cérézo, and Leblond [25] with the help of optimal control techniques. Sussmann and Tang showed that only 46 different shortest path classes are necessary for the Reeds-Shepp car. Souères and Laumond classified the shortest paths for the Reeds-Shepp car into symmetric classes, and also gave the optimal control synthesis, i.e. the path class and its parameters for any pair of configurations [161]. Balkcom and Mason gave a complete characterization of time-optimal trajectories for the differential drive [13]. They also characterized the time-optimal trajectories for an omni-directional mobile robot [12]. Souères and Boissonnat studied the time optimality of the Dubins car with angular acceleration control [160]. They presented an incomplete characterization of time-optimal trajectories for their system. A full characterization of such time-optimal trajectories seems to be difficult because Sussmann proved that some time-optimal trajectories for that system require infinitely many input switchings in a finite time interval (chattering or Fuller phenomenon) [163]. Sussmann used Zelikin and Borisov's theory of chattering control to prove his result [181]. Reister and Pin took a numerical approach to time optimality for differential-drive robots [145]. Renaud and Fourquet studied numerical time-optimal paths for acceleration-driven mobile robots [146]. Chyba and Sekhavat studied time optimality for a mobile robot with one trailer [49]. A car with multiple trailers is an example of a chained system and Goursat structure [134, 149, 159]. Geodesics for Goursat structures were studied by Pasillas-Lépine and Respondek [135], and for chained systems by Sarychev and Nijmeijer [151]. However, a complete characterization of the time-optimal trajectories for a bounded velocity car with n trailers is still an open problem.

1.2.5 Shortest Path Algorithms

Here we divert our attention from nonholonomic shortest path results, and consider shortest paths among obstacles without nonholonomic constraints [70, 76, 79, 117, 123, 129]. Canny and Reif showed that the problem, among general polyhedral obstacles, is NP-hard in the three dimensional Euclidean space [36]. On the contrary, there are polynomial time approximation algorithms for that problem [47, 133]. There are several polynomial algorithms for the two dimensional case. The 2-dimensional shortest path problem has been formulated mainly in two different settings: semialgebraic obstacles [70, 75] and polygonal obstacles [78, 79, 117]. In this dissertation, we focus on the second setting; a method that inspires some of our work is the visibility graph method [52]. The visibility graph was introduced for motion planning by Nilsson [129]. A visibility graph comprises a network of collision-free shortest path segments between obstacle vertices. The graph is usually built offline to answer multiple shortest path queries. To compute the visibility graph, Lee gave a radial sweep algorithm with $O(n^2 \log n)$ time complexity [106]. Later in Section 2.4, we will give a detailed overview of Lee's algorithm. A more efficient algorithm, which runs in $O(n^2)$ time, has been proposed [58]. Ghosh and Mount gave an optimal output-sensitive algorithm for computing the visibility graph [66]. Any algorithm that computes the shortest path by first constructing the entire visibility graph has at least quadratic running time. Mitchell showed that the shortest path can be computed in $O(n^{3/2+\epsilon})$ time [124]. Hershberger and Suri developed an optimal algorithm for the shortest path problem [79]. Their algorithm runs in $O(n \log n)$ time. A shortest polygonal path with specified endpoints can be determined efficiently in a simple polygon [72, 77] and with a specified homotopy [70, 78]. Furthermore, using the idea of retraction motion planning [131], a path of maximum clearance, within a specified homotopy class, can be determined efficiently from the generalized Voronoi diagram of the domain. For a survey on current shortest path methods see [125].

1.2.6 Combined Approaches

Differential constraints add a major level of difficulty to shortest path problems. Desaulniers proved that the shortest path for the Reeds-Shepp car need not even exist among obstacles [53]. However, the shortest path always exists for the Dubins car and the convexified Reeds-Shepp car. This fact can be proved using Filippov's theorem [38]. Reif and Wang showed that computing a shortest path for the Dubins car amongst general polygonal obstacles is NP-hard [143]. In contrast, efficient approximation algorithms are known [2, 3, 82]. Laumond studied shortest paths that are composed of straight line, arc, and obstacle boundary for the Dubins car [100]. Fortune and Wilfong gave an algorithm to compute a collision-free path for the Dubins car among polygonal obstacles [62]. Their algorithm is exponential in time and space. Boissonnat et al. presented a linear time algorithm for finding a convex unit-curvature path (not necessarily shortest), if one exists, in a simple polygon [26]. Ahn et al. characterized reachable regions, by the Dubins car from a given start configuration, inside a convex polygon with n vertices [4]. They show that the number of such regions is $O(n)$. Agarwal et al. presented an $O(n^2 \log n)$ time algorithm for determining whether a collision-free path for the Dubins car exists between two configurations inside a convex polygon [1]. Bereg and Kirkpatrick studied traversals of narrow corridors by unit-curvature paths [20]. Lutterkort and Peters studied the construction of smooth paths, using spline functions, in channels defined by two polygonal chains [119]. Boissonnat and Lazard gave an $O(n^2 \log n)$ algorithm for computing a shortest path for the Dubins car amongst disjoint moderate obstacles [27]. An obstacle is said to be *moderate* if it is convex and its boundary is a differentiable curve whose curvature is everywhere not more than 1. Agarwal et al. obtained an $O(n^2 \log n)$ algorithm for finding shortest paths for the Dubins car in a convex polygonal region. They also gave an approximation algorithm for the Reeds-Shepp car [2]. Note that their algorithm computes a collision-free path for the Reeds-Shepp car, that is at most a constant amount longer than a shortest path. Moutarlier et al. studied the problem of finding the shortest distance for the Reeds-Shepp car to a manifold in the configuration space [128]. Desaulniers et al. gave an algorithm to compute the shortest path for the Reeds-Shepp car among polygonal obstacles by decomposing the space into polygonal regions and discretizing boundaries of the regions [54]. Venditelli et al. presented a method to compute the shortest distance for a car-like robot from a given configuration to the obstacle region [170, 171]. They used optimal control tools, namely the transversality condition of the Pontryagin Maximum Principle, to solve the problem.

1.3 Our Contribution

In this dissertation, we integrate visibility graph methods and nonlinear optimal control tools into a unified novel technique for computing geodesics for mobile robots among obstacles. We demonstrate our technique by considering geodesics for three mechanisms: 1) the differential drive among obstacles, 2) a kinematic airplane, which we call *Dubins airplane*, and 3) multiple polygonal robots moving on a predetermined network of paths.

1.3.1 Geodesics for the Differential Drive Among Obstacles

We consider the differential drive because it is a common mobile robot platform. To obtain a well-defined notion of *shortest*, the total amount of wheel rotation is optimized. We prove the existence of minimum wheel-rotation trajectories using the Filippov theorem. We analytically characterize minimum wheel-rotation trajectories in the absence of obstacles. Using that characterization and further analysis, we present a *nonholonomic bitangency graph* which is used to find minimum wheel-rotation trajectories in the presence of obstacles. Vertices of the graph are on the obstacle boundary, and there is an edge between two vertices if they can see each other by a minimum wheel-rotation segment, which is conceptually like a straight line segment in the Euclidean space. We give a general bitangency condition which restricts possible edges. In general, our nonholonomic bitangency graph is a 2-dimensional subset of the 3-dimensional configuration space of the robot. Therefore, further optimization or a continuous search may be required to answer queries. When obstacles are circular and far enough from one another, the graph is 1-dimensional and can be computed. In that case, any graph search algorithm, such as Dijkstra's algorithm, is employed to extract the solution.

1.3.2 Geodesics for the Dubins Airplane

We analytically characterize geodesics for a new airplane model. Our airplane is a natural extention of the Dubins car [56], and extends it with an additional configuration variable for the altitude. In this way, our airplane is able to move in a 3D work space. To obtain a notion of shortest, time is optimized. Besides a set of time-minimizing maneuvers, the time-optimal trajectories comprise a useful set of motion primitives as it was discussed in Section 1.1. The time-optimal trajectories also play a crucial role in air traffic management systems [95, 168, 174, 182], e.g. in detecting the safety regions.

1.3.3 Combined Geodesics for Multiple Polygonal Robots

We present an algorithm for computing optimal coordination of two polygonal robots without differential constraints. Each robot has a reference point that must lie on a given graph, called a roadmap, which is embedded in the plane. Each robot wants to move from its given initial location to its goal location without colliding with the other one. Rather than impose an a priori cost scalarization for choosing the best combined motion, we consider finding motions whose cost vectors are Pareto-optimal. Pareto-optimal coordination strategies are the ones for which there exists no strategy that would be better for both robots. The problem is equivalent to computing $\mathcal{L}^\infty$-geodesics in the coordination space which is the Cartesian product of the roadmap with itself. We extend visibility graphs to solve the problem.

1.4 Outline

We conclude this introductory chapter with preview of the remaining chapters.

In Chapter 2, we review required background material. We include only those subjects that are needed throughout multiple chapters. We define geodesic problems for the differential drive and our airplane by presenting their respective differential constraints and cost functions. We prove the existence of those geodesics using the Filippov theorem. Eventually, the Pontryagin Maximum Principle and visibility graph method are reviewed.

In Chapter 3, we derive the family of 52 minimum wheel-rotation trajectories for a differential-drive mobile robot in the plane without obstacles. Up to symmetry, we identify 6 different classes of minimum wheel-rotation trajectories that are maximal with respect to subpath partial order. Minimum wheel-rotation trajectories are composed of three motion primitives: rotation in place, straight line, and swing segments (one wheel stationary and the other rolling). Six classes of minimum wheel-rotation trajectories are shown in Figure 1.1 and the robot model in Figure 2.1. We prove that minimum time for the convexified Reeds-Shepp car [164] is equal to minimum wheel-rotation for the differential drive, and the two families of optimal curves are identical. As of writing this dissertation, it is unknown whether there is a proof for this fact that does not require optimal control tools. That chapter was a joint work with Steven M. LaValle, Devin J. Balkcom, and Matthew T. Mason [44, 45].

Our results in Chapter 3 form the first step of our approach to finding geodesics among

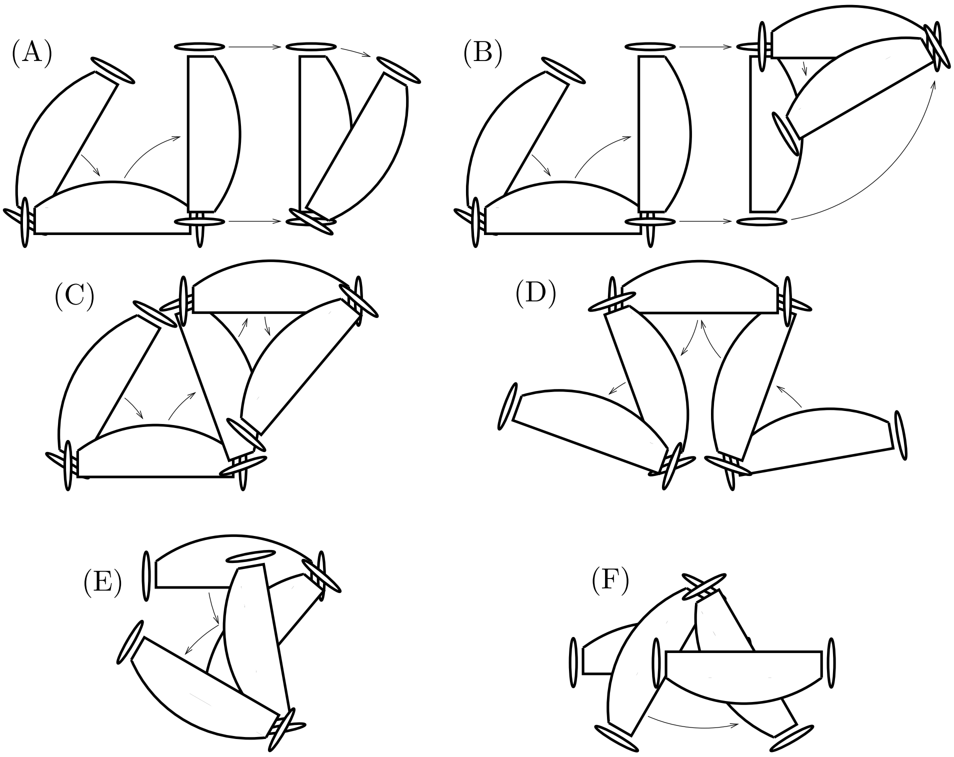


Figure 1.1: Six classes of minimum wheel-rotation trajectories up to symmetry in Chapter 3.

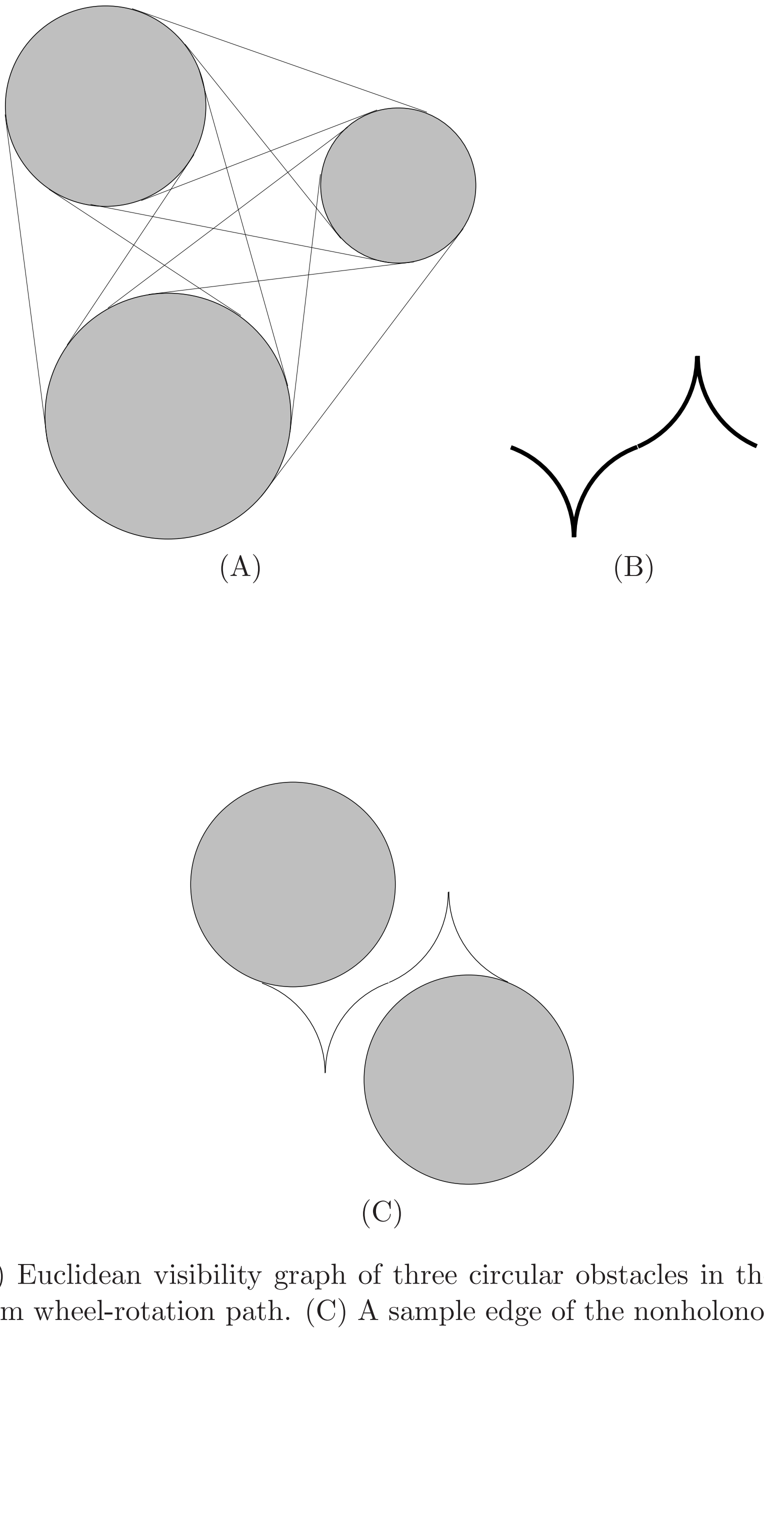


Figure 1.2: (A) Euclidean visibility graph of three circular obstacles in the plane. (B) A sample minimum wheel-rotation path. (C) A sample edge of the nonholonomic bitangency graph.

obstacles. We present the remaining three steps in Chapter 4:

1. Characterization of minimum wheel-rotation trajectories on the boundary of obstacle region using the Pontryagin Maximum Principle,

2. Characterization of intersection points between free and boundary minimum wheel-rotation trajectories using the Pontryagin Jump Condition [141], and

3. Definition of a nonholonomic bitangency graph in which the solution is sought.

Vertices of the nonholonomic bitangency graph are points on the obstacle boundary, and its edges are composed of minimum wheel-rotation segments, each of which is either thoroughly on the obstacle boundary or in the interior of the free portion of the configuration space. The edges are generally bitangent, i.e. tangent to the obstacle boundary at both ends. Figure 1.2 shows a sample Euclidean visibility graph, a minimum wheel-rotation path, and the minimum wheel-rotation path as an edge of the nonholonomic bitangency graph. For each shortest path query, initial and goal configurations are appropriately appended to the graph, and a search on parts or all of the graph gives the solution.

In Chapter 5, we characterize time-optimal trajectories for our airplane through the use of the Pontryagin Maximum Principle. We assume that the system has independent bounded control over the altitude velocity as well as the turning rate in the plane. The time-optimal trajectories are composed of three motion primitives: turns with minimum radius, straight line segments, and pieces of planar elastica [85]. Some examples of time-optimal paths are depicted in Figure 1.3. In the figure, those subpaths that are in between ℓ_+ and ℓ_- are planar elastica, and others are arcs of circle. We distinguish three cases: low, medium, and high goal altitudes of the airplane. Intuitively, if the goal altitude is low, the airplane has to follow the shortest path for the Dubins car with an unsaturated altitude velocity. If the goal altitude is high, the altitude velocity gets saturated and the system has to maneuver until it reaches the goal altitude. For medium altitudes in between low and high, the time-optimal path is either a locally longest curve for the Dubins car or a path composed of turns and pieces of planar elastica with saturated altitude velocity. Locally longest curves for the Dubins car, which cannot be infinitesimally elongated, play an

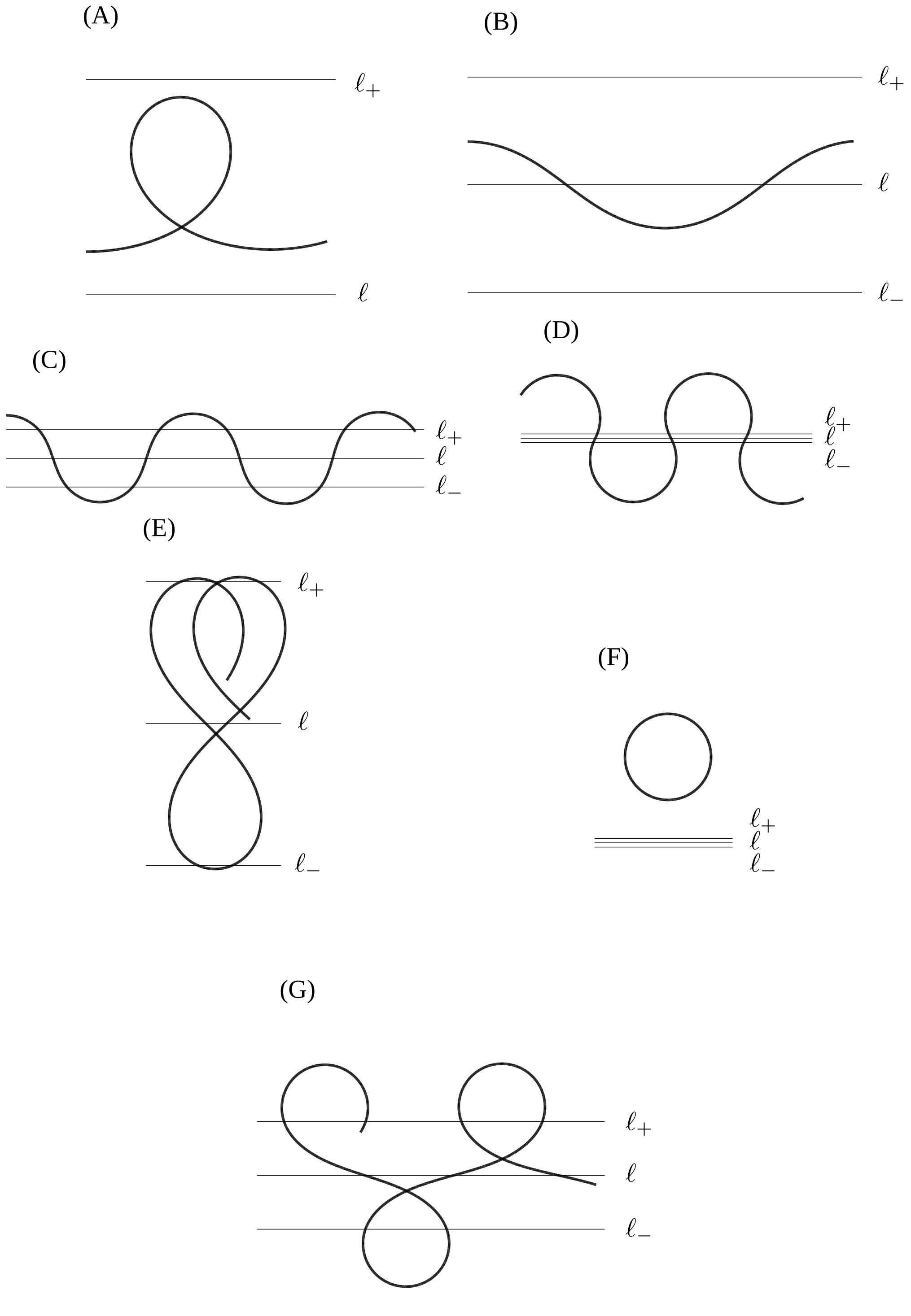


Figure 1.3: Some examples of time-optimal paths for our airplane. Depicted paths are projection of time-optimal trajectories onto the plane. Those subpaths between ℓ_+ and ℓ_- are planar elastica; otherwise they are arcs of circle.

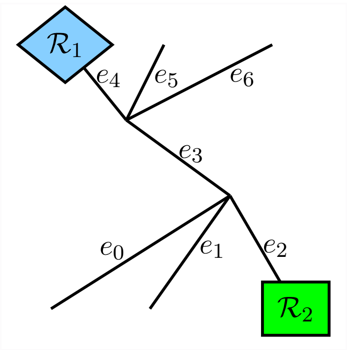


Figure 1.4: A Pareto-optimal coordination problem on a roadmap with 7 edges. The two robots want to exchange place.

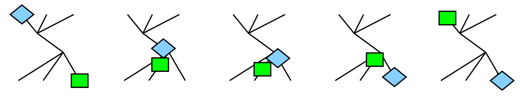
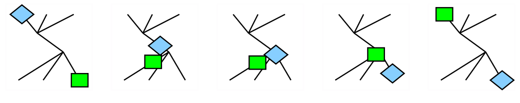
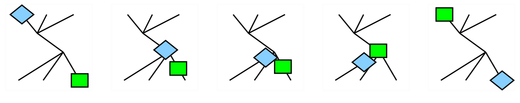
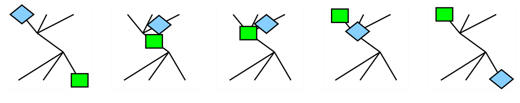
	($\mathcal{R}_1$ cost, $\mathcal{R}_2$ cost)
	(8.9,14.8)
	(9.3,14.3)
	(14.4,13.7)
	(15.1,8.7)

Figure 1.5: The four solutions for the problem in Figure 1.4.

important role in the airplane time-optimal trajectories for medium altitude. An example of such locally longest curves is a short arc of circle. As a by-product, we characterize locally longest curves.

In Chapter 6, we present an algorithm to compute Pareto-optimal coordinations of two polygonal robots moving on a roadmap. We first consider the case where the underlying roadmaps are trees. We present an algorithm that computes the complete set of Pareto-optimal coordination strategies in time $O(mn^2 \log n)$, in which m is the number of paths in the roadmap, and n is the number of coordination space vertices. Our algorithm computes the visibility graph in the coordination space, augments it with extra edges, and computes the shortest path. Figure 1.4 shows a coordination problem on a roadmap with 7 edges and

Figure 1.5 presents the four Pareto-optimal solutions for the problem. Second, we present an algorithm that solves the general case. That algorithm computes an upper bound on the cost of each motion in any Pareto-optimal coordination. Thus, only a finite number of homotopy classes of paths in the coordination space need to be considered. In effect, the new algorithm applies the first algorithm to a finite portion of the universal cover of the roadmap. The algorithm computes solutions in time $O(2^{5\alpha} m^{1+5\alpha} n^2 \log(m^{2\alpha} n))$, in which m is the number of edges in the roadmap, n is the number of coordination space obstacle vertices, and $\alpha = 1 + \lceil (5\ell + r)/b \rceil$ where ℓ is total length of the roadmap and r is total length of coordination space obstacle boundary and b is the length of the shortest edge in the roadmap. That chapter was a joint work with Steven M. LaValle and Jason M. O'Kane [46].

Chapter 2

Background

2.1 Geodesics for Mobile Robots

In this section, we define geodesic problems that we consider in this dissertation for the differential drive and the Dubins airplane. According to some metric, a geodesic is a shortest path among a class of admissible paths. In an equivalent formulation, a geodesic is an admissible path that minimizes the integral of a cost function. We define admissible paths, which satisfy the differential constraints and avoid obstacles, for the differential drive and the Dubins airplane. We complete presentation of the problems by giving the cost function for each problem.

2.1.1 Differential Drive

A differential-drive robot [13, 45] is a three-dimensional system with its configuration variable denoted by $q = (x, y, \theta) \in \mathcal{C} = \mathbb{R}^2 \times \mathbb{S}^1$ in which x and y are the coordinates of the point on the axle, equidistant from the wheels, in a fixed frame in the plane, and $\theta \in [0, 2\pi)$ is the angle between x-axis of the frame and the robot local longitudinal axis; see Figure 2.1.

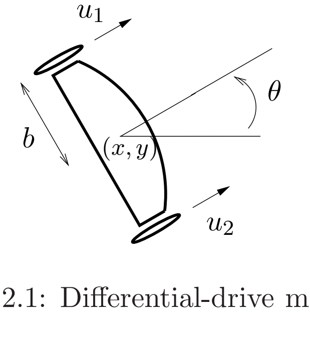


Figure 2.1: Differential-drive model.

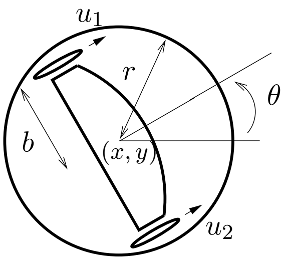


Figure 2.2: A differential-drive disc of radius r.

The robot has independent velocity control of each wheel. Assume that the wheels have equal bounds on their velocity. More precisely, $u_1, u_2 \in [-1, 1]$, in which the inputs u_1 and u_2 are respectively the left and the right wheel velocities, and the input space is $U = [-1, 1] \times [-1, 1] \subset \mathbb{R}^2$. The system is

$$\dot{q} = f(q, u) = u_1 f_1(q) + u_2 f_2(q), \tag{2.1}$$

in which f_1 and f_2 are vector fields as follows. Let the distance between the robot wheels be $2b$. In that case,

$$f_1 = \frac{1}{2}\begin{pmatrix} \cos\theta \\ \sin\theta \\ -\frac{1}{b} \end{pmatrix} \text{ and } f_2 = \frac{1}{2}\begin{pmatrix} \cos\theta \\ \sin\theta \\ \frac{1}{b} \end{pmatrix}. \tag{2.2}$$

The cost functional J to be minimized is

$$J(u) = \int_0^T L(u(t))dt, \tag{2.3}$$

$$L(u) = \frac{1}{2}(|u_1| + |u_2|). \tag{2.4}$$

The factor $\frac{1}{2}$ above helps to simplify further formulas, and does not alter the optimal trajectories.

The robot is a closed disc of radius $r > b$; see Figure 2.2. We assume that there are n obstacles, $O_1, O_2, \ldots, O_n$, in the workspace of the robot. Each O_i is a bounded, open, and convex subset of $\mathbb{R}^2$. Recall that the robot is a disc of radius r. Let

$$P_i = \{p \in \mathbb{R}^2 \ \mid \ d(p, O_i) < r\} = O_i + B_r, \tag{2.5}$$

in which d is the Euclidean distance from a set, and B_r is a disc of radius r. The obstacle region in the configuration space of the robot is

$$\mathcal{C}_{obs} = (P_1 \cup P_2 \cup \cdots \cup P_n) \times \mathbb{S}^1. \tag{2.6}$$

We also assume that $\overline{P}_i$ are disjoint. Hence

$$\partial\mathcal{C}_{obs} = (\partial P_1 \cup \partial P_2 \cup \cdots \cup \partial P_n) \times \mathbb{S}^1. \tag{2.7}$$

Note that P_i are open subsets of $\mathbb{R}^2$, and hence, $\mathcal{C}_{obs}$ is open. Let $\mathcal{C}_{free} = \mathcal{C} \backslash \mathcal{C}_{obs}$ be the free part of the configuration space. Note that $\mathcal{C}_{free}$ is closed and $\partial\mathcal{C}_{free} = \partial\mathcal{C}_{obs}$. It is obvious that ∂P_i are simple, piecewise-smooth curves.

Proposition 2.1. *The curvature of ∂P_i is not more than $\dfrac{1}{r}$ everywhere, for $i = 1, 2, \ldots, n$.*

Sketch of proof. For every point $p \in \partial P_i$, a circle of radius r tangent to ∂P_i at p is contained in $P_i \cup \partial P_i$. This implies that the curvature of ∂P_i is not more than $\frac{1}{r}$ everywhere. ■

For every pair of free initial and goal configurations, not on the boundary of $\mathcal{C}_{free}$, we seek an admissible control, i.e. a measurable function $u : [0, T] \to U$, that minimizes J while transferring the initial configuration to the goal configuration in $\mathcal{C}_{free}$. Since the cost J is invariant by scaling the input within U, we can assume without loss of generality that the controls are either constantly zero ($u \equiv (0, 0)$) or saturated at least in one input, i.e. $\max(|u_1(t)|, |u_2(t)|) = 1$ for all $t \in [0, T]$. Since $u \equiv (0, 0)$ gives trivial motionless trajectory, we assume throughout this dissertation that $u \not\equiv (0, 0)$. Throughout this dissertation, a trajectory for which $u \equiv (0, 0)$ over its time interval is called *motionless*.

2.1.2 Dubins Airplane

Our model, called *Dubins airplane*, extends the Dubins car with an additional configuration variable for the altitude. The Dubins airplane is a four-dimensional system with its configuration variable denoted by $p = (x, y, z, \theta) \in \mathcal{C}' = \mathbb{R}^3 \times \mathbb{S}^1$ in which x, y, and z are the coordinates of the airplane in the three-dimensional Euclidean space, and $\theta \in [0, 2\pi)$ is the angle between x-axis of the frame and the airplane local longitudinal axis in $x - y$ plane; see Figure 2.3. Equivalently, the Dubins airplane is the Dubins car, $(x, y, \theta) \in \mathbb{R}^2 \times \mathbb{S}^1$, with an additional configuration variable for altitude, z. This model is a simplified model of a real airplane.

The system has independent bounded control of $\dot{\theta}$ and $\dot{z}$. In other words, the system is

$$\dot{p} = k(p, u) = k_0(p) + u_z k_z(p) + u_\theta k_\theta(p) \tag{2.8}$$

in which k_0, k_z, and k_θ are vector fields in the tangent bundle of the configuration space. We assume the minimum turning radius and the maximum altitude velocity of the airplane are 1. In this case, k_0, k_z, and k_θ are

$$k_0 = \begin{pmatrix} \cos\theta \\ \sin\theta \\ 0 \\ 0 \end{pmatrix}, \ k_z = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \text{ and } k_\theta = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}. \tag{2.9}$$

We assume that $|u_z|, |u_\theta| \leq 1$. Thus, the control region is $U = [-1, 1]^2$ and $(u_z, u_\theta) \in U$. The cost functional J' to be minimized is time, i.e. $J'(u) = \int_0^T dt$. For every pair of initial and goal configurations, we seek an admissible control, i.e. a measurable function $u : [0, T] \to U$, that minimizes J' while transferring the initial configuration to the goal configuration. Without loss of generality we may assume, throughout this dissertation, that the initial configuration of the system is $(0, 0, 0, 0) \in \mathcal{C}'$. We also denote the goal configuration by $(x_g, y_g, z_g, \theta_g)$. Throughout the dissertation, sgn is the sign function.

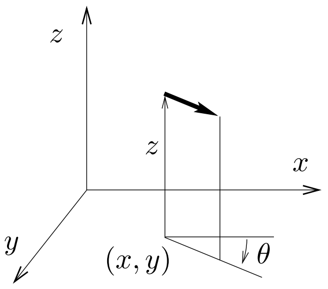


Figure 2.3: The Dubins airplane model.

2.2 Existence of Optimal Trajectories

In this section, we show that our geodesic problems are well-behaved, so that it is viable to use the Pontryagin Maximum Principle. More precisely, we prove the existence of a minimum wheel-rotation trajectory for every pair of initial and goal configurations. We also prove that time-optimal trajectories for the Dubins airplane exist.

2.2.1 Existence of Minimum Wheel-Rotation Trajectories

The differential drive is clearly controllable [13]. Moreover, it can be shown that it is small-time locally controllable. Hence, there exists at least one trajectory between any pair of initial and goal configurations, and it is meaningful to discuss the existence of optimal trajectories. In the following, we will use a version of the Filippov Existence Theorem to prove the existence of minimum wheel-rotation trajectories.

Theorem 2.2 (Filippov Existence Theorem [38])**.** *Let $A \subset \mathcal{C}$ be compact, $G \subset \mathcal{C} \times \mathcal{C}$ closed, $L(u)$ continuous on U, and f continuous on $A \times U$. Define $Q(q) \subset \mathbb{R} \times T_q\mathcal{C} \cong \mathbb{R}^4$ as*

$$Q(q) = \{(z_0, z) | \exists u \in U : z_0 \geq L(u) \text{ and } z = f(q, u)\}. \tag{2.10}$$

Let Ω_A be the set of all admissible trajectory-control pairs $(q(t), u(t))$ defined on $[0, T]$ that for some $(q_0, q_1) \in G$ transfer q_0 to q_1 while staying in A, i.e. $(q(0), q(T)) \in G$, and $q([0, T]) \subset A$. Assume that $Q(q)$ are convex for all $q \in A$, and Ω_A is nonempty. The functional J has an absolute minimum in the nonempty class Ω_A.

From this we derive the following corollary which establishes the existence of minimum wheel-rotation trajectories for the system described in (2.1).

Corollary 2.3. *Minimum wheel-rotation trajectories for the differential-drive exist.*

Proof. Fix the initial configuration $q_0 = (x_0, y_0, \theta_0)$ and the goal configuration $q_1 = (x_1, y_1, \theta_1)$. Let A in Theorem 2.2 be $A = \mathcal{C}_{free} \cap (B_T(x_0, y_0) \times \mathbb{S}^1)$, in which $B_T(x_0, y_0)$ is the closed ball of radius T around (x_0, y_0) in the plane. Note that T here is both maximum time and the radius of $B_T(x_0, y_0)$. Assume T is large enough so that $(x_1, y_1) \in B_T(x_0, y_0)$. The projection of robot configuration onto the x-y plane cannot leave $B_T(x_0, y_0)$ in time T because $\sqrt{\dot{x}^2 + \dot{y}^2} \leq 1$. Thus, any trajectory starting at q_0 stays in A over the time interval $[0, T]$. Choose T such that $\Omega_A \neq \emptyset$ in Theorem 2.2. Let $G = \{(q_0, q_1)\} \subset \mathcal{C} \times \mathcal{C}$ be the pair of initial and goal configurations.

It is obvious that A is compact, G closed, $L(u)$ continuous on U, and f continuous on $A \times U$ in this case. Since U is convex and $f(q, \cdot)$ is a linear transformation, $f(q, U)$ is also convex. The fact that $L(\cdot)$ is a convex function helps to show $Q(q)$ is convex for all q. Thus, Theorem 2.2 guarantees the existence of a minimum wheel-rotation trajectory-control pair $(q_T(t), u_T(t))$ in Ω_A. Let $J_T = J(u_T)$, and let τ be the time of q_T. In that case, $\tau \leq T$ because $(q_T(t), u_T(t))$ is in Ω_A. Since $L \leq 1$ along any trajectory, $J_T \leq \tau \leq T$.

Now let the time duration be $2T$ and $A' = \mathcal{C}_{free} \cap (B_{2T}(x_0, y_0) \times \mathbb{S}^1)$. Using Theorem 2.2 again, $\Omega_{A'}$ contains a minimum wheel-rotation trajectory-control pair $(q_{2T}(t), u_{2T}(t))$. Let $J_{2T} = J(u_{2T})$. Note that $J_{2T} \leq J_T$ because all elements of Ω_A are contained in $\Omega_{A'}$. Any trajectory-control pair that is not in $\Omega_{A'}$ takes at least $2T$ time. Observe that $1/2 \leq L$ along any trajectory because at least one input is saturated. Hence, the cost of any trajectory-control pair that is not in $\Omega_{A'}$ is at least $2T/2 = T$. Note that $J_{2T} \leq J_T \leq T$. Thus, $q_{2T}(t)$ is an absolute minimum wheel-rotation trajectory over all trajectories. ■

2.2.2 Existence of Time-optimal Trajectories for the Dubins Airplane

The Dubins airplane is an extension of the Dubins car by adding a configuration variable for altitude. Since the altitude variable and input are decoupled from the Dubins car, the controllability of the Dubins airplane follows from that of the Dubins car. Hence, there exists at least one trajectory between any pair of initial and goal configurations, and it is meaningful to discuss the existence of optimal trajectories. We again use the Filippov Theorem to prove the existence of time-optimal trajectories.

Theorem 2.4 (Filippov Existence Theorem [38]). *Let $A \subset \mathcal{C}'$ be compact, $G \subset \mathcal{C}' \times \mathcal{C}'$ closed, and k continuous on $A \times U$. Define $Q(p) \subset T_p\mathcal{C}' \cong \mathbb{R}^4$ as*

$$Q(p) = \{z | \exists u \in U : z = k(p, u)\}. \tag{2.11}$$

Let Ω_A be the set of all admissible trajectories $p(t)$ defined on $[0, T]$ that for some $(p_0, p_1) \in G$ transfer p_0 to p_1 while staying in A, i.e. $(p(0), p(T)) \in G$, and $p([0, T]) \subset A$. Assume that $Q(p)$ are convex for all $p \in A$, and Ω_A is nonempty. The functional J' has an absolute minimum in the nonempty class Ω_A.

The existence of time-optimal trajectories follows from the theorem above in a similar way as for the differential drive. Fix the initial configuration $p_0 = (x_0, y_0, z_0, \theta_0)$ and the goal configuration $p_1 = (x_1, y_1, z_1, \theta_1)$. Let $A = B_T(x_0, y_0) \times [z_0 - T, z_0 + T] \times \mathbb{S}^1$, in which $B_T(x_0, y_0)$ is the closed ball of radius T around (x_0, y_0) in the plane. Note that T here is both the duration of the trajectory and the radius of $B_T(x_0, y_0)$. Let $G = \{(p_0, p_1)\}$, and choose T such that $\Omega_A \neq \emptyset$. It is obvious that A is compact, G closed, and k continuous on $A \times U$ in this case. Since U is convex and $k(p, \cdot)$ is a linear transformation, $Q(p) = k(p, U)$ is also convex. Since Ω_A contains any trajectory that transfers p_0 to p_1 in time T, the theorem above guarantees the existence of a time-optimal trajectory.

2.3 Pontryagin Maximum Principle and Jump Condition

This section barely affords a compact summary of celebrated Pontryagin's results in optimal control. Since that much is enough for the rest of this dissertation, curious reader is advised to read the original monograph written by Pontryagin, Boltyanskii, Gamkrelidze, and Mishchenko [141]. We formulate a general optimal control problem, and we state the Pontryagin Maximum Principle for it. We also consider the case in which the state space of the system is constrained. We state the Pontryagin Maximum Principle and Jump Condition for the constrained problem as well. Please note that all of the notations are limited to this section, and not used in the rest of this dissertation. Throughout this section, we consider the following system:

$$\frac{dx}{dt} = f(x,u) = (f^1(x,u), f^2(x,u), \ldots, f^n(x,u)), \tag{2.12}$$

in which $x = (x^1, x^2, \ldots, x^n) \in X$, where the state space $X \subset \mathbb{R}^n$ is an open set, and $u \in U \subset \mathbb{R}^m$.

2.3.1 Statement of the Problem

We assume in (2.12) f is continuous in x and u, and continuously differentiable with respect to x. Equivalently, $f^i(x,u), \partial f^i(x,u)/\partial x^j$ are continuous on $X \times U$ for $i, j = 1, 2, \ldots, n$. A cost function $f^0(x,u)$ is given. We assume f^0 is continuous in x and u, and continuously differentiable in x. Two points x_0 and x_1 are given in the state space X. We seek an admissible control, i.e. a measurable function $u : [0,T] \to U$, that minimizes

$$J = \int_0^T f^0(x(t), u(t))dt \tag{2.13}$$

while transferring x_0 to x_1. Here, $x(t)$ is the solution of (2.12) with initial condition $x(0) = x_0$ corresponding to the control $u(t)$, and T is such that $x(T) = x_1$. An *optimal control corresponding to a transition from x_0 to x_1* is the control $u(t)$ at which the minimum of J is achieved. Note that optimal control need not always exist.

2.3.2 Maximum Principle

To state the maximum principle, we consider an auxiliary system of equations

$$\frac{d\lambda_i}{dt} = -\sum_{k=0}^{n} \frac{\partial f^k(x(t), u(t))}{\partial x^i} \lambda_k(t), \quad i = 1, 2, \ldots, n, \tag{2.14}$$

in which $\lambda_0 \in \mathbb{R}$ is a constant and $\lambda = (\lambda_0, \lambda_1, \ldots, \lambda_n) \in \mathbb{R}^{n+1}$ is called *adjoint*. Note that by (2.14), $\lambda(t)$ is uniquely determined if a trajectory-control pair $(x(t), u(t))$ and the initial condition $\lambda(0)$ are given. Define a function $H\colon \mathbb{R}^{n+1} \times X \times U \to \mathbb{R}$ as

$$H(\lambda, x, u) = \langle \lambda, (f^0, f) \rangle = \sum_{k=0}^{n} \lambda_k f^k(x, u). \tag{2.15}$$

It is clear that (2.14) and (2.12) can be rewritten as

$$\frac{d\lambda_i}{dt} = -\frac{\partial H}{\partial x^i}(\lambda(t), x(t), u(t)), \quad i = 1, 2, \ldots, n, \tag{2.16}$$

and

$$\frac{dx^i}{dt} = f^i(x, u) = \frac{\partial H}{\partial \lambda_i}(\lambda(t), x(t), u(t)), \quad i = 1, 2, \ldots, n, \tag{2.17}$$

in which case H plays the role of a Hamiltonian function. For constant values of λ and x, the Hamiltonian H becomes a function of the parameter u. We denote the least upper bound of the values of this function by

$$M(\lambda, x) = \sup_{u \in U} H(\lambda, x, u). \tag{2.18}$$

Theorem 2.5 (Pontryagin Maximum Principle [141])**.** *Let $x(t)$ be a trajectory for the system (2.12) defined on $[0, T]$ associated with control $u(t)$. For $u(t)$ to be optimal, it is necessary that there exist a nonzero continuous vector-valued adjoint function $\lambda(t)$ corresponding to $(x(t), u(t))$ through (2.16), such that $\lambda_0 \leq 0$,*

$$H(\lambda(t), x(t), u(t)) = M(\lambda(t), x(t)), \tag{2.19}$$

and

$$M(\lambda(t), x(t)) = 0, \text{ for } t \in [0, T]. \tag{2.20}$$

2.3.3 Constrained Maximum Principle

In Section 2.3.2, we assumed that the state space X is an open subet of $\mathbb{R}^n$. New difficulties arise when we confine the state space to a closed set. Here, we confine x to $F \subset X$, a closed region whose boundary is a piecewise smooth hypersurface. We consider only those optimal trajectories that can be split into a finite number of sections each of which lies either entirely on a smooth piece of the boundary of F or entirely in the interior of F. Those sections of the optimal trajectory that lie in the interior of F satisfy the unconstrained maximum principle in Section 2.3.2. Those sections that lie entirely on the boundary of F satisfy a constrained maximum principle stated in this section. Finally, every pair of adjacent sections satisfy a certain condition (characterizing jumps in the adjoint λ) which is called the jump condition. We present the jump condition in Section 2.3.4.

First, we give the required definitions and notions. Let $g : X \to \mathbb{R}$ be a smooth function such that the inequality $g(x) \leq 0$ locally defines F near the boundary. We assume g has continuous second partial derivatives near the boundary $g(x) = 0$, and the vector $\partial g/\partial x$ does now vanish anywhere on the boundary. Let

$$p(x,u) = \langle \frac{\partial g}{\partial x}, f(x,u) \rangle = \sum_{k=1}^{n} \frac{\partial g}{\partial x^k} f^k(x,u), \tag{2.21}$$

and H and M be defined as in Section 2.3.2.

Theorem 2.6 (Constrained Maximum Principle [141]). *Let $x(t)$ be a trajectory for the system (2.12) defined on $[0,T]$ associated with control $u(t)$. Assume $x(t)$ lies entirely on the boundary of F. For $u(t)$ to be optimal, it is necessary that there exist a nonzero continuous vector-valued adjoint function $\lambda(t)$ and a piecewise continuous function $\eta(t)$ such that $\lambda_0 =$ const. ≤ 0,*

$$\frac{d\lambda_i}{dt} = -\frac{\partial H(\lambda(t), x(t), u(t))}{\partial x^i} + \eta(t)\frac{\partial p(x(t), u(t))}{\partial x^i}, \quad i = 1, 2, \ldots, n, \tag{2.22}$$

$$H(\lambda(t), x(t), u(t)) = M(\lambda(t), x(t)) = 0, \text{ for } t \in [0,T], \tag{2.23}$$

and

$$\frac{d\eta}{dt} \leq 0 \tag{2.24}$$

wherever $d\eta/dt$ exists.

2.3.4 Jump Condition

An optimal trajectory in the closed region F lies partly in the interior of F and partly on the boundary. Those sections of the optimal trajectory that lie in the interior of F must satisfy Theorem 2.5. Those sections of the optimal trajectory that lie on the boundary of F must satisfy Theorem 2.6. Those theorems guarantee the existence of an adjoint for each section. In the following theorem, adjoints of two adjacent sections are related to one another.

Theorem 2.7 (Pontryagin Jump Condition [141])**.** *Let $(x(t), u(t))$ be an optimal trajectory-control pair defined on $[0, T]$ for the constrained problem, i.e. $x(t) \in F$. Suppose $\tau \in [0, T]$ is such that $x(\tau)$ is on the boundary of F, and for $\sigma > 0$ each of $x(]\tau-\sigma, \tau[)$ and $x(]\tau, \tau+\sigma[)$ lies either entirely in the interior of F or entirely on the boundary of F, locally defined by $g(x) = 0$. Let $\lambda(\tau^-)$ and $\lambda(\tau^+)$ be the left and the right limits of the adjoint λ respectively. In that case, the adjoints can be chosen so that either*

$$\lambda_0(\tau^+) = \lambda_0(\tau^-), \tag{2.25}$$

$$\lambda_i(\tau^+) = \lambda_i(\tau^-) + \mu\frac{\partial g}{\partial x^i}(x(\tau)) \tag{2.26}$$

or

$$\lambda_0(\tau^-) = 0, \tag{2.27}$$

$$\lambda_i(\tau^-) + \mu\frac{\partial g}{\partial x^i}(x(\tau)) = 0, \quad \mu \neq 0, \tag{2.28}$$

for $i = 1, 2, \ldots, n$ and some constant μ. Moreover, (2.26) is equivalent to $\lambda(\tau^+) = \lambda(\tau^-)$ if $x([\tau, \tau+\sigma[)$ lies on the boundary.

2.4 Visibility Graph

This section defines a visibility graph among polygonal obstacles and presents Lee's algorithm to compute it [106]. For a detailed presentation of the algorithm see [52]. Lee's algorithm has $O(n^2 \log n)$ time complexity, in which n is the number of obstacle vertices. An optimal algorithm was given by Ghosh and Mount which runs in $O(n \log n + k)$ time, in which k is the number of edges of the visibility graph [66]. We choose Lee's algorithm here because it is more intuitive. We will refer to this section in Chapters 4 and 6.

To define visibility graph, suppose we are given two points p_{init} and p_{goal} and a set S of disjoint polygonal obstacles in the Euclidean plane. Lemma 15.1 of [52] which follows, characterizes shortest collision-free paths from p_{init} to p_{goal}.

Lemma 2.8 ([52])**.** *Any shortest path between p_{init} and p_{goal} among obstacles S is a piecewise-linear (polygonal) path whose inner vertices are vertices of S.*

To find a shortest path, we construct a network of paths which is called the *visibility graph* of S. Vertices of the visibility graph are vertices of S. We say two vertices v and w can *see* each other if the straight line segment between v and w does not intersect the interior of any obstacle in S. There is an edge (v, w) in the visibility graph if v and w can see each other. In that case, the line segment that connects v and w is called a visibility edge. Note that obstacle edges in S are visibility edges. We add p_{init} and p_{goal} as vertices to the visibility graph, and add visibility edges between them and obstacle vertices. Lemma 2.8 shows that any shortest path between p_{init} and p_{goal} must lie on the visibility graph. Once the visibility graph is computed, a graph shortest path algorithm such as Dijkstra's algorithm gives the solution. Figure 2.4 shows a visibility graph of two obstacles and the shortest path between p_{init} and p_{goal}.

2.4.1 Computing the Visibility Graph

To compute the visibility graph $G = (V, E)$, we need to find pairs of vertices that can see each other. For every pair we have to test whether the straight line segment connecting them is collision-free. A naïve test takes $O(n)$ time, in which n denotes the number of obstacle vertices. Therefore, a naïve algorithm has $O(n^3)$ time complexity. Lee's algorithm

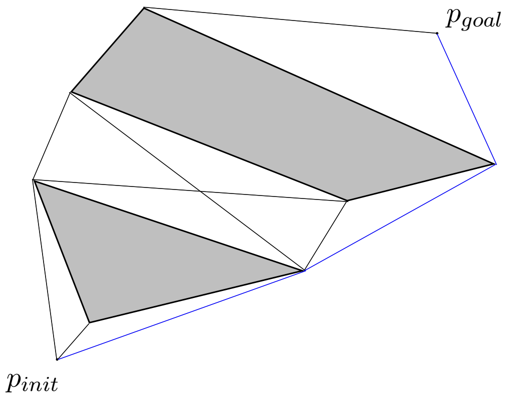


Figure 2.4: A visibility graph.

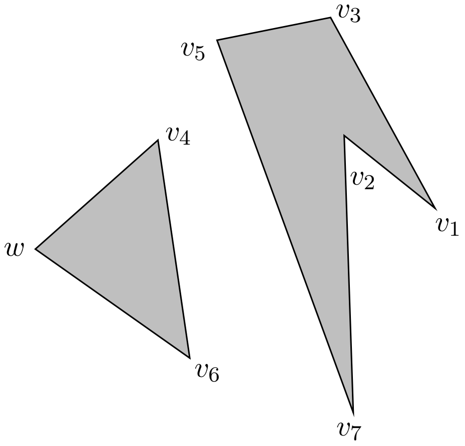


Figure 2.5: The sequence of vertices in a rotational counterclockwise plane sweep centered at w.

concentrates on one vertex at a time and performs a rotational counterclockwise plane sweep to compute its visibility edges. In the following, let w be a vertex of the visibility graph for which the algorithm computes visibility edges.

For each $v \in V$ different from w, let $\alpha_w(v)$ be the counterclockwise angle that the line from w to v makes with the positive x-axis. Sort V ascendingly according to α_w. In the case of equal entries, sort according to the distance from w. Let $v_1, \ldots, v_{n-1}$ be the sorted list. Figure 2.5 illustrates an example. Determine whether v_1 is visible from w and find those obstacle edges that intersect the half-line that emanates from w and passes through v_1. Sort them according to their distance from w. Consider only those edges incident to v_1

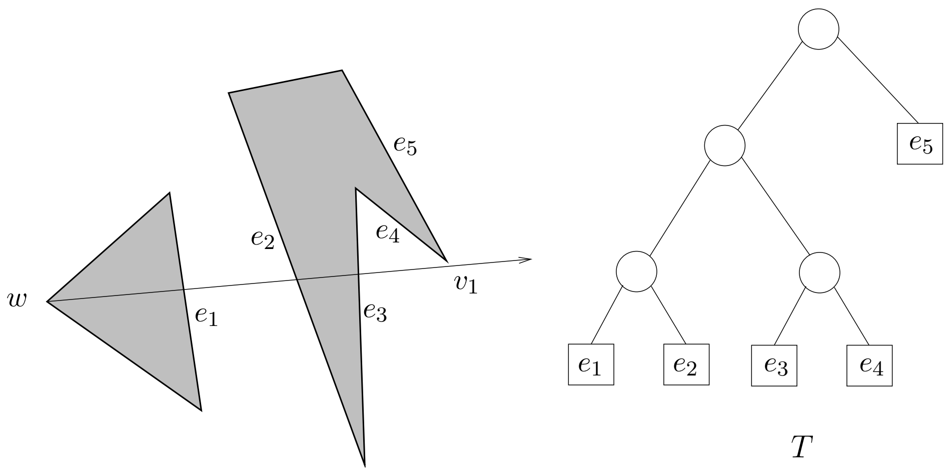


Figure 2.6: Those obstacle edges that intersect the segment w to v_1 and the search tree T.

that are on the counterclockwise side of the half-line from w to v_1. Let the sorted list be $e_1, \ldots, e_m$. Store e_i in order from left to right in the leaves of a balanced binary search tree T. An inner node of T is the rightmost leaf of its left subtree. Figure 2.6 demonstrates an example. In the algorithm, T maintains a sorted list of candidate edges that may obstruct visibility of w. For $i = 2$ to $n - 1$, do the following two steps:

1. Test if v_i is visible from w. This test can be done by searching T. We will describe below how to do that.

2. Insert into T those obstacle edges incident to v_i that lie on the counterclockwise side of the half-line from w to v_i. Delete from T those obstacle edges incident to v_i that lie on the clockwise side of the half-line.

To determine whether v_i and w can see each other, first verify that the line segment from w to v_i does not intersect the interior of the obstacle of which v_i is a vertex. Suppose v_{i-1} is not on the segment w to v_i. In that case, search T to find its leftmost leaf e. If e does not intersect the segment from w to v_i, then w and v_i can see each other. Now suppose v_{i-1} is on the segment w to v_i. If v_{i-1} cannot see w, then v_i evidently cannot see w. If v_{i-1} can see w, determine whether in T there is an edge that intersects the segment from v_{i-1} to v_i.

It is obvious that the two steps above take $O(\log n)$ time. Therefore, computing visibility edges for each w takes $O(n \log n)$ time, and the whole algorithm runs in $O(n^2 \log n)$ time.

2.5 Summary

In this chapter, we first defined minimum wheel-rotation for the differential drive. We then introduced the Dubins airplane and proved the existence of optimal paths for the two problems. We gave a compact summary of the Pontryagin Maximum Principle and Jump Condition. Finally, we presented Lee's visibility graph algorithm. The Pontryagin Maximum Principle and the visibility graph are the main tools that we use in this dissertation.

Chapter 3

Minimum Wheel-Rotation Paths for the Differential Drive

This chapter derives the family of 52 minimum wheel-rotation trajectories for a differential-drive mobile robot in the plane without obstacles. The shortest paths are composed of rotation in place, straight line, and swing segments (one wheel stationary and the other rolling). Twenty eight different minimum wheel-rotation trajectories are identified that are maximal with respect to subpath partial order. Up to symmetry, they are in 6 distinct classes. Although there are some numerical optimal control algorithms that can be utilized to solve the problem, a complete mathematical characterization of shortest paths, in the sense of Dubins and Reeds-Shepp curves, is helpful in comparing different mechanisms, computing a nonholonomic metric for motion planning algorithms, and building a local motion planner. In addition, we will use our analysis here to characterize minimum wheel-rotation paths among obstacles in Chapter 4.

The problem was precisely defined in Section 2.1.1. We proved existence of minimum wheel-rotation trajectories in Section 2.2.1. It is then plausible to apply the Pontryagin Maximum Principle as a necessary condition, which was reviewed in Section 2.3. The analysis is completed by our geometric arguments which rule out non-optimal trajectories. We prove that minimum time for the convexified Reeds-Shepp car [164] is equal to minimum wheel-rotation for the differential drive, and the two families of optimal curves are identical. As of writing this dissertation, it is unknown whether there is a proof for this fact that does not require optimal control tools. This chapter was a joint work with Steven M. LaValle, Devin J. Balkcom, and Matthew T. Mason [44, 45].

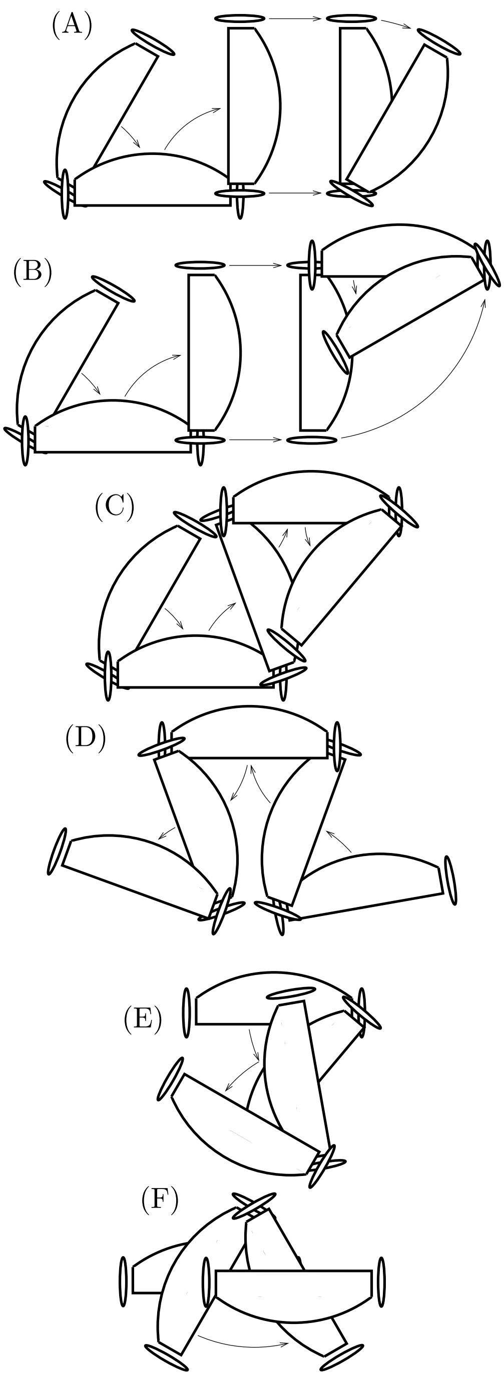


Figure 3.1: Minimum wheel-rotation trajectories up to symmetry: (A) and (B) are composed of two swings, straight, and one or two swings respectively. (C) and (D) are composed of four alternating swings. (E) is composed of swing, rotation in place, and swing. (F) is composed of rotation in place, swing, and rotation in place.

3.1 Related Work

Sussmann and Tang proposed a framework, based on the Pontryagin Maximum Principle [141], for solving shortest path problems for mobile robots [164]. Their approach brought the power of the Pontryagin Maximum Principle to help researchers and proved to be successful. In their template, one first proves that optimal paths exist. It is then viable to apply the necessary condition of the Pontryagin Maximum Principle. The geometric interpretation of the Pontryagin Maximum Principle leads to geometric arguments that rule out some non-optimal trajectories. The remaining trajectories are then filtered through a problem-specific argument. Usually, the last step is the most difficult one.

The approach that we use to derive optimal trajectories is similar to the one used by Sussmann and Tang [164], Souères, Boissonnat and Laumond [160, 161], Chyba and Sekhavat [49], and Balkcom and Mason [13]. However, we give specific geometric arguments to rule out non-optimal trajectories. In Section 2.2.1, we proved existence of minimum wheel-rotation trajectories by using Filippov's theorem [38]. We follow the template by applying the Pontryagin Maximum Principle and deriving its geometric interpretations. This step filters out some non-optimal trajectories. The remaining finite set of candidates are compared with each other to find the optimal ones.

3.2 Necessary Conditions

Since we proved the existence of optimal trajectories, it is viable now to apply the Pontryagin Maximum Principle which is a necessary condition for optimality.

3.2.1 Pontryagin Maximum Principle

Let the Hamiltonian $H : \mathbb{R}^3 \times \mathcal{C} \times U \to \mathbb{R}$ be

$$H(\lambda, q, u) = \langle \lambda, \dot{q} \rangle + \lambda_0 L(u) \tag{3.1}$$

in which λ_0 is a constant. According to the Pontryagin Maximum Principle [141], which was summarized in Section 2.3, for every optimal trajectory $q(t)$ defined on $[0, T]$ and associated with control $u(t)$, there exists a constant $\lambda_0 \leq 0$ and an absolutely continuous vector-valued adjoint function $\lambda(t)$, that is nonzero if $\lambda_0 = 0$, with the following properties along the optimal trajectory:

$$\begin{aligned} \dot{\lambda} &= -\frac{\partial}{\partial q} H, && (3.2) \\ H(\lambda(t), q(t), u(t)) &= \max_{z \in U} H(\lambda(t), q(t), z), && (3.3) \\ H(\lambda(t), q(t), u(t)) &\equiv 0. && (3.4) \end{aligned}$$

Definition 3.1. An *extremal* is a trajectory $q(t)$ that satisfies the conditions of the Pontryagin Maximum Principle. Also, an extremal for which $\lambda_0 = 0$ is called *abnormal.*

Let the switching functions be

$$\varphi_1 = \langle \lambda, f_1 \rangle \text{ and } \varphi_2 = \langle \lambda, f_2 \rangle , \tag{3.5}$$

in which f_1 and f_2 are given by (2.2). We rewrite (3.1) as $H = u_1\varphi_1 + u_2\varphi_2 + \lambda_0 L$. The Pontryagin Maximum Principle implies that an optimal trajectory is also an *extremal*; however, the converse is not necessarily true. Throughout the current section, we characterize *all* extremals because the optimal trajectories are among them. In the following sections, we will provide more restrictive conditions for optimality and we will rule out all non-optimal ones.

3.2.2 Switching Structure Equations

Lemma 3.1 (Sussmann and Tang [164])**.** *Let f_k be a smooth vector field in the tangent bundle of the configuration space $T\mathcal{C}$, and let $q(t)$ be an extremal associated with control $u(t)$ and adjoint vector $\lambda(t)$. Let φ_k be defined as $\varphi_k(t) = \langle \lambda(t), f_k(q(t)) \rangle$. It follows that*

$$\dot{\varphi}_k = u_1 \langle \lambda, [f_1, f_k] \rangle + u_2 \langle \lambda, [f_2, f_k] \rangle . \tag{3.6}$$

Lemma 3.1 reveals valuable information by relating the structure of the Lie algebra to the structure of φ_i functions. To complete the Lie closure of $\{f_1, f_2\}$, we introduce f_3 as the

Lie bracket of f_1 and f_2:

$$f_3 = [f_1, f_2] = \frac{1}{2b} \begin{pmatrix} \sin\theta \\ -\cos\theta \\ 0 \end{pmatrix}. \tag{3.7}$$

Let $\varphi_3(t) = \langle \lambda(t), f_3(q(t)) \rangle$ be the switching function associated with f_3. Lemma 3.1 implies the structure of switching functions as follows [13]:

$$\dot{\varphi}_1 = -u_2\varphi_3, \dot{\varphi}_2 = u_1\varphi_3, \dot{\varphi}_3 = \frac{1}{4b^2}(-u_1 + u_2)(\varphi_1 + \varphi_2). \tag{3.8}$$

The vectors f_i are linearly independent. Consequently, $\{f_1(q), f_2(q), f_3(q)\}$ forms a basis for $T_q\mathcal{C}$. As an immediate consequence of the Pontryagin Maximum Principle and Lemma 3.1, the following proposition holds.

Proposition 3.2. *An abnormal extremal is motionless.*

Proof. If $\lambda_0 = 0$, then (3.4) implies $u_1\varphi_1 + u_2\varphi_2 \equiv 0$. This means $|\varphi_1| \equiv |\varphi_2| \equiv 0$ because by maximization of the Hamiltonian, we must have $u_i\varphi_i = |\varphi_i|$ for $i = 1, 2$. For a detailed argument, see [13]. Consequently, φ_1 and φ_2 are constantly zero, and $\dot{\varphi}_1 \equiv \dot{\varphi}_2 \equiv 0$. In this case, $|\varphi_1| + |\varphi_2| + |\varphi_3| \neq 0$ because $\{f_1, f_2, f_3\}$ forms a basis for tangent space of the configuration space, and φ_i's are the coordinates of a nonzero vector $\lambda(t)$ in this basis. Thus, $\varphi_3 \neq 0$ and (3.8) imply $u_1 \equiv u_2 \equiv 0$. ■

3.2.3 Extremals

Having dealt with abnormal extremals in Proposition 3.2, we may now, without loss of generality, scale the Hamiltonian (3.1) so that $\lambda_0 = -2$. More precisely, the Pontryagin Maximum Principle conditions are valid if we replace $\lambda(t)$ by $-\frac{2\lambda(t)}{\lambda_0}$ and λ_0 by -2 in (3.1). We will assume that $\lambda_0 = -2$ for the rest of the chapter. In that case, the Hamiltonian has the simple form

$$H = u_1\varphi_1 + u_2\varphi_2 - (|u_1| + |u_2|). \tag{3.9}$$

Equation (3.2) can be solved for λ to obtain

$$\lambda(t) = \begin{pmatrix} c_1 \\ c_2 \\ c_1 y - c_2 x + c_3 \end{pmatrix}, \tag{3.10}$$

in which c_1, c_2, and c_3 are constants. Let $i, j \in \{1, 2\}$ throughout the rest of the chapter.

Definition 3.2. An extremal for which $|\varphi_i(t)| = 1$ over some positive-length interval of time is called *singular*, for some $i = 1, 2$.

In Lemma 3.3, we will show that a non-singular extremal is motionless. We will also show that there are two categories of singular extremals depending on whether or not $c_1^2 + c_2^2 = 0$. The first category corresponds to $c_1^2 + c_2^2 \neq 0$, and consists of all singular extremals that are composed of a number of swing ($u_i = 0$) and straight ($u_1 = u_2$) intervals. Such extremals will be called *tight*. The second category corresponds to $c_1^2 + c_2^2 = 0$. Such extremals will be called *loose*.

Lemma 3.3. *Let $q(t)$ be an extremal associated with the control $u(t) = (u_1(t), u_2(t))$, adjoint vector function $\lambda(t)$, and switching functions $\varphi_i(t)$. Moreover, assume $q(t)$ is not motionless. In that case, the following hold:*

(i) $|\varphi_i(t)| \leq 1$.

(ii)

$$u_i(t) \in \begin{cases} [0, 1] & \text{if} \quad \varphi_i(t) = 1 \\ \{0\} & \text{if} \quad |\varphi_i(t)| < 1 \\ [-1, 0] & \text{if} \quad \varphi_i(t) = -1 \end{cases} . \tag{3.11}$$

(iii) If $c_1^2 + c_2^2 \neq 0$ and $|\varphi_1| = |\varphi_2| = 1$ over some interval $[t_1, t_2]$, then $u_1 = u_2$, and $\varphi_1 = \varphi_2$.

(iv) If $c_1^2 + c_2^2 \neq 0$ and $|\varphi_j| < |\varphi_i| = 1$ over a time interval $[t_1, t_2]$, then $u_j = 0$ and $|u_i| = 1$, in which $j \neq i$.

(v) If $c_1 = c_2 = 0$, then $\varphi_1 \equiv -\varphi_2$, and $u_1 u_2 \leq 0$. In other words, the wheels move in opposite directions.

Proof. *(i)* By inspection of (3.9), if $|\varphi_i| > 1$, there exist feasible controls yielding $H > 0$. This contradicts the maximum principle (3.3) and (3.4), which states that the maximum of H is zero.

(ii) If $|\varphi_i| < 1$, then (3.3) and (3.9) implies $u_i = 0$. In a similar way, if $\varphi_i = 1$, then $u_i \in [0, 1]$, and if $\varphi_i = -1$, then $u_i \in [-1, 0]$.

(iii) Assume $\varphi_1 = -\varphi_2$. From (2.2), (3.5), and (3.10) it follows that $c_1 \cos\theta + c_2 \sin\theta \equiv 0$. Differentiate this equation to obtain $\dot{\theta} \equiv 0$ because $-c_1 \sin\theta + c_2 \cos\theta \neq 0$. Thus, $2b\dot{\theta} = u_1 - u_2 = 0$, and (3.11) implies $u_1 = u_2 = 0$, which is not possible because $q(t)$ is not motionless.

(iv) This follows from (3.11).

(v) In that case, $\varphi_1 \equiv -\varphi_2$ by (2.2), (3.5), and (3.10). It follows from (3.11) that $u_1 u_2 \leq 0$. ■

Geometric interpretation of tight extremals in Section 3.2.4 will help to show that the number of switchings along a tight extremal is finite. Along a tight extremal we can assume $u_1 = 0, u_2 \in \{1, -1\}$ or $u_1 \in \{1, -1\}, u_2 = 0$ on swing segments, and $u_1 = u_2 \in \{1, -1\}$ on straight segments because at least one of the inputs is saturated. Thus, inputs are always either zero or bang $u_i \in \{1, 0, -1\}$ along tight extremals. In Section 3.3.3, we will show that there may exist many wheel-rotation equivalent loose extremals, and for an appropriate choice of representative loose extremals, the inputs are always either zero or bang. In this section, we finished an elementary characterization of extremals. We have identified three main types of extremals:

1. *non-singular*: $u_1 \equiv u_2 \equiv 0$ (i.e. motionless)
2. *tight singular*: composed of a finite number of swing and straight segments
3. *loose singular*: $u_1 u_2 \leq 0$, $\varphi_1 \equiv -\varphi_2$, and $|\varphi_1| \equiv |\varphi_2| \equiv 1$.

3.2.4 Geometric Interpretation of Tight Extremals

Let (x_1, y_1) and (x_2, y_2) be the coordinates of the left and the right wheel respectively. In that case,

$$\begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} x - b\sin\theta \\ y + b\cos\theta \end{pmatrix} \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} x + b\sin\theta \\ y - b\cos\theta \end{pmatrix}. \tag{3.12}$$

Define functions $\gamma_1(x, y)$ and $\gamma_2(x, y)$ as

$$\gamma_1(x, y) = c_1 y - c_2 x + c_3 - 2b, \tag{3.13}$$

$$\gamma_2(x, y) = c_1 y - c_2 x + c_3 + 2b. \tag{3.14}$$

Taking (2.2), (3.5), (3.10), (3.12), (3.13), and (3.14) into account, we obtain

$$\varphi_1 = -\frac{1}{2b}\gamma_2(x_2, y_2) + 1 = -\frac{1}{2b}\gamma_1(x_2, y_2) - 1, \tag{3.15}$$

$$\varphi_2 = \frac{1}{2b}\gamma_1(x_1, y_1) + 1 = \frac{1}{2b}\gamma_2(x_1, y_1) - 1. \tag{3.16}$$

Note that $c_1^2 + c_2^2 > 0$, and consider the parallel lines $\ell_1 : \gamma_1(x, y) = 0$ and $\ell_2 : \gamma_2(x, y) = 0$ in the robot x-y plane. The value of γ_i at each point $P \in \mathbb{R}^2$ determines $d(P, \ell_i)$ scaled by $\sqrt{c_1^2 + c_2^2}$ for $i = 1, 2$, in which $d(P, \ell)$ is the signed distance of point P from a line $\ell \subset \mathbb{R}^2$. Since the base distance b of the robot is positive, $\gamma_2 > \gamma_1$ everywhere in the plane. Thus, ℓ_1 and ℓ_2 cut the plane into five disjoint subsets (Figure 3.2): S_+, ℓ_1, $S_\pm$, ℓ_2, and S_- in which

$$S_+ = \{(x, y) \in \mathbb{R}^2 |\ \ \gamma_2(x, y) > \gamma_1(x, y) > 0\} \tag{3.17}$$

$$S_\pm = \{(x, y) \in \mathbb{R}^2 |\ \ \gamma_2(x, y) > 0 > \gamma_1(x, y)\} \tag{3.18}$$

$$S_- = \{(x, y) \in \mathbb{R}^2 |\ \ 0 > \gamma_2(x, y) > \gamma_1(x, y)\}. \tag{3.19}$$

Using Lemma 3.3 and (3.15) and (3.16), along a tight extremal $\gamma_1(x_i, y_i) \leq 0 \leq \gamma_2(x_i, y_i)$ for $i = 1, 2$. Thus, the robot always stays in the band $\ell_1 \cup S_\pm \cup \ell_2$; see Figure 3.2. By

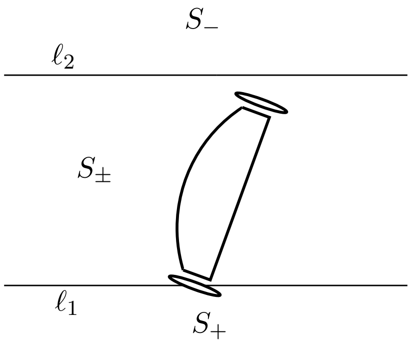


Figure 3.2: The robot stays between two lines ℓ_1 and ℓ_2 along a tight extremal.

appropriately substituting in (3.11), we obtain

$$u_1 \in \begin{cases} [-1,0] & \text{if} \quad \text{wheel } 2 \quad \in \quad \ell_1 \\ \{0\} & \text{if} \quad \text{wheel } 2 \quad \in \quad S_\pm \\ [0,1] & \text{if} \quad \text{wheel } 2 \quad \in \quad \ell_2 \end{cases} \tag{3.20}$$

$$u_2 \in \begin{cases} [0,1] & \text{if} \quad \text{wheel } 1 \quad \in \quad \ell_1 \\ \{0\} & \text{if} \quad \text{wheel } 1 \quad \in \quad S_\pm \, . \\ [-1,0] & \text{if} \quad \text{wheel } 1 \quad \in \quad \ell_2 \end{cases} \tag{3.21}$$

3.3 Characterization of Extremals

3.3.1 Symmetries

Assume $(q(t), u(t))$ is a minimum wheel-rotation trajectory-control pair that is defined on $[0, T]$. Let $\tilde{q}(t)$ be the trajectory associated with control $u(T-t)$, $\bar{q}(t)$ the trajectory associated with control $-u(t)$, and $\hat{q}(t)$ the trajectory associated with control $\hat{u}(t) = (u_2(t), u_1(t))$. Define the operators $\mathcal{O}_1$, $\mathcal{O}_2$, and $\mathcal{O}_3$ acting on trajectory-control pairs by

$$\mathcal{O}_1 : (q(t), u(t)) \quad \mapsto \quad (\tilde{q}(t), u(T-t)) \tag{3.22}$$

$$\mathcal{O}_2 : (q(t), u(t)) \quad \mapsto \quad (\bar{q}(t), -u(t)) \tag{3.23}$$

$$\mathcal{O}_3 : (q(t), u(t)) \quad \mapsto \quad (\hat{q}(t), \hat{u}(t)). \tag{3.24}$$

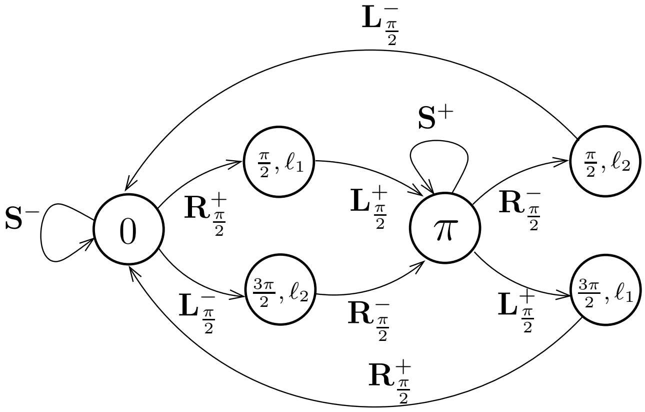

$\mathbf{L}^-_{\frac{\pi}{2}}$
$\mathbf{S}^+$
$\mathbf{S}^-$
0
$\frac{\pi}{2}, \ell_1$
$\mathbf{R}^+_{\frac{\pi}{2}}$
$\mathbf{L}^+_{\frac{\pi}{2}}$
π
$\mathbf{R}^-_{\frac{\pi}{2}}$
$\frac{\pi}{2}, \ell_2$
$\frac{3\pi}{2}, \ell_2$
$\mathbf{L}^-_{\frac{\pi}{2}}$
$\mathbf{R}^-_{\frac{\pi}{2}}$
$\mathbf{L}^+_{\frac{\pi}{2}}$
$\frac{3\pi}{2}, \ell_1$
$\mathbf{R}^+_{\frac{\pi}{2}}$

Figure 3.3: $\mathcal{F}_1$ is a finite state machine whose language is the tight extremals for which the distance between ℓ_1 and ℓ_2 is $2b$ (Case 1).

Due to symmetries, $\mathcal{O}_1(q(t), u(t))$, $\mathcal{O}_2(q(t), u(t))$, and $\mathcal{O}_3(q(t), u(t))$ are also minimum wheel-rotation trajectories. $\mathcal{O}_1$ corresponds to reversing the extremal in time, $\mathcal{O}_2$ corresponds to reversing the inputs, and $\mathcal{O}_3$ corresponds to exchanging the left and the right wheels.

3.3.2 Characterization of Tight Extremals

In the following we give only the representatives of symmetric families of tight extremals. We will use L, R, and S to denote swing around the left wheel, the right wheel, and straight line motions, respectively. In cases where the directions must be specified, we use a superscript: $-$ is clockwise, $+$ is counter-clockwise, $+$ is forward, and $-$ is backward. Otherwise, the direction of swing is constant throughout the extremal. The symbol * means zero or more copies of the base expression. Subscripts are non-negative angles.

Depending on the distance between ℓ_1 and ℓ_2 we identify three different types of tight extremals. For each type, we define a finite state machine to present extremals more precisely.

Case 1: Let $d(\ell_1, \ell_2) = 2b$. Besides swing, the robot can move straight forward and backward by keeping the wheels on ℓ_i's. In this case, the extremals are composed of a sequence of swing and straight segments. In general, there can be an arbitrary number of swing and straight segments. Since the straight segments can be translated and merged together, a

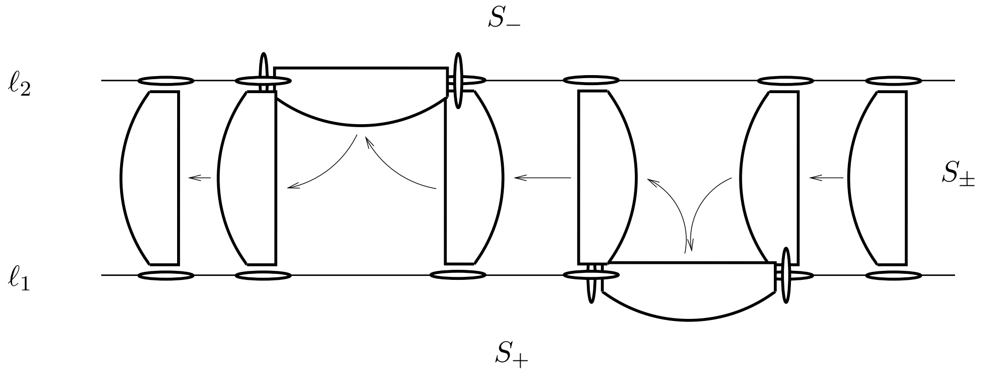


Figure 3.4: An $\mathcal{F}_1$ trajectory.

representative subclass with only one straight segment is described by the following forms:

- $(\mathbf{R}_\pi^- \mathbf{L}_\pi^-)^* \mathbf{R}_{\frac{\pi}{2}}^- \mathbf{S}^+ \mathbf{R}_{\frac{\pi}{2}}^- (\mathbf{L}_\pi^- \mathbf{R}_\pi^-)^*$
- $(\mathbf{R}_\pi^- \mathbf{L}_\pi^-)^* \mathbf{R}_{\frac{\pi}{2}}^- \mathbf{S}^+ \mathbf{L}_{\frac{\pi}{2}}^+ (\mathbf{R}_\pi^+ \mathbf{L}_\pi^+)^*$.

We define a finite state machine $\mathcal{F}_1$ to present such extremals more precisely. Let

$$Q_1 = \{0, (\frac{\pi}{2}, \ell_1), (\frac{\pi}{2}, \ell_2), \pi, (\frac{3\pi}{2}, \ell_1), (\frac{3\pi}{2}, \ell_2)\} \tag{3.25}$$

be the set of states. States are the robot orientations together with its position, i.e. whether it lies on the line ℓ_1 or ℓ_2. Let the input alphabet be $\Sigma_1 = \{\mathbf{S}^+, \mathbf{S}^-, \mathbf{L}_{\frac{\pi}{2}}^+, \mathbf{L}_{\frac{\pi}{2}}^-, \mathbf{R}_{\frac{\pi}{2}}^+, \mathbf{R}_{\frac{\pi}{2}}^-\}$. Define $\mathcal{F}_1$ by the transition function that is depicted in Figure 3.3. If robot starts in one of the states in Q_1, it has to move according to $\mathcal{F}_1$. If the initial configuration of robot is none of the states, the robot performs a compliant $\mathbf{L}_\alpha$ or $\mathbf{R}_\alpha$ motion, in which $0 \leq \alpha < \frac{\pi}{2}$, to reach one of the states and continues according to $\mathcal{F}_1$. In general, there can be an arbitrary number of swing and straight segments. Since the straight segments can be translated and merged together, a representative subclass with only one straight segment suffices for giving all such minimum wheel-rotation trajectories. For optimal representatives of this class see (A) and (B) in Figure 3.1. We call such tight extremals *type I.*

Case 2: Let $d(\ell_1, \ell_2) > 2b$. The robot cannot move straight because it cannot keep the wheels on the lines ℓ_i over some interval of time. Thus, such extremals are of the form

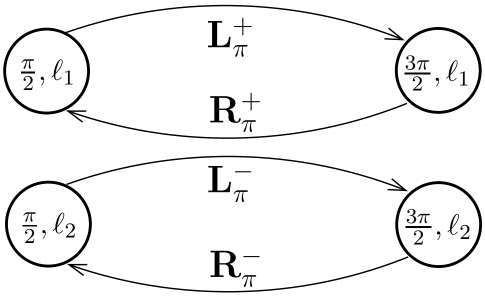


Figure 3.5: $\mathcal{F}_2$ is a finite state machine whose language is the tight extremals for which the distance between ℓ_1 and ℓ_2 is greater than $2b$ (Case 2).

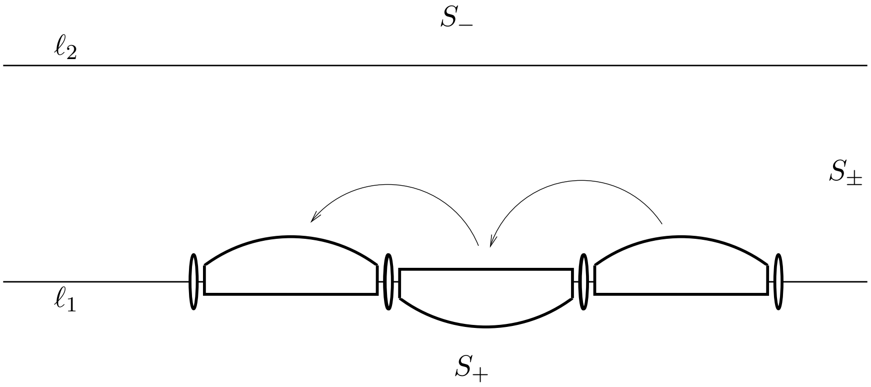


Figure 3.6: An $\mathcal{F}_2$ trajectory.

$(\mathbf{R}_\pi \mathbf{L}_\pi)^*$. Note that these extremals are subpaths of type I extremals. Again, we define a finite state machine $\mathcal{F}_2$ to present such extremals more precisely. Let

$$Q_2 = \{(\frac{\pi}{2}, \ell_1), (\frac{\pi}{2}, \ell_2), (\frac{3\pi}{2}, \ell_1), (\frac{3\pi}{2}, \ell_2)\} \tag{3.26}$$

be the set of states. States are the robot orientations together with its position, i.e. whether it lies on the line ℓ_1 or ℓ_2. Let the input alphabet be $\Sigma_2 = \{\mathbf{L}_\pi^+, \mathbf{L}_\pi^-, \mathbf{R}_\pi^+, \mathbf{R}_\pi^-\}$. Define $\mathcal{F}_2$ by the transition function that is depicted in Figure 3.5.

Case 3: Let $d(\ell_1, \ell_2) < 2b$. In this case, the extremals are of the form $(\mathbf{L}_\gamma^- \mathbf{R}_\gamma^- \mathbf{L}_\gamma^+ \mathbf{R}_\gamma^+)^*$ in which $\gamma \leq \frac{\pi}{2}$. Like the two previous cases, we define a finite state machine $\mathcal{F}_3$ to present

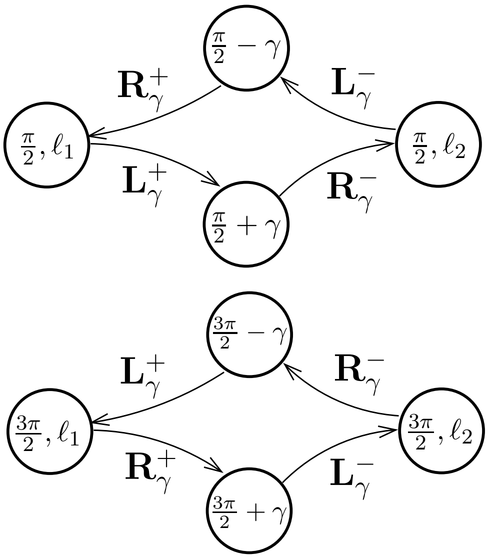


Figure 3.7: $\mathcal{F}_3$ is a finite state machine whose language is the tight extremals for which the distance between ℓ_1 and ℓ_2 is less than $2b$ (Case 3).

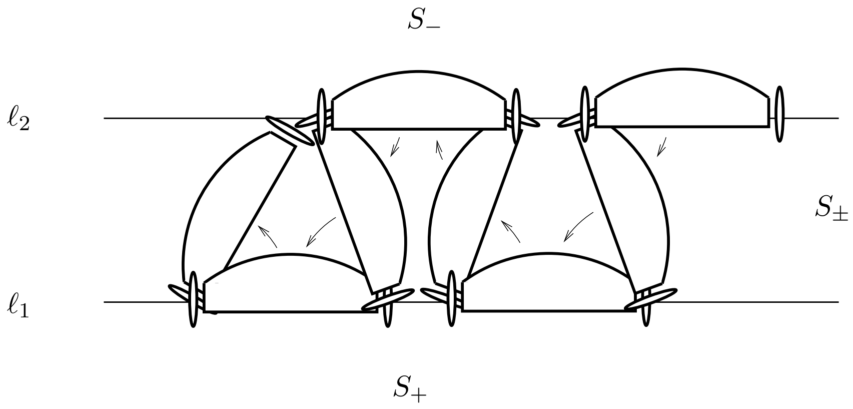


Figure 3.8: An $\mathcal{F}_3$ trajectory.

such extremals more precisely. Let

$$Q_3 = \{\frac{\pi}{2} - \gamma, (\frac{\pi}{2}, \ell_1), (\frac{\pi}{2}, \ell_2), \frac{\pi}{2} + \gamma, \frac{3\pi}{2} - \gamma, (\frac{3\pi}{2}, \ell_1), (\frac{3\pi}{2}, \ell_2), \frac{3\pi}{2} + \gamma\} \tag{3.27}$$

be the set of states. States are the robot orientations together with its position, i.e. whether it lies on the line ℓ_1 or ℓ_2. Let the input alphabet be $\Sigma_3 = \{\mathbf{L}_\gamma^+, \mathbf{L}_\gamma^-, \mathbf{R}_\gamma^+, \mathbf{R}_\gamma^-\}$. Define $\mathcal{F}_3$ by the transition function that is depicted in Figure 3.7. For optimal representatives of this class see (C) and (D) in Figure 3.1. We call such tight extremals *type II.*

Lemma 3.4. *Let $q(t)$ be a tight extremal associated with the control $u(t)$ that transfers (x_0, y_0, θ_0) to (x_1, y_1, θ_1). In this case*

$$J(u) = l = \int_0^T (\sqrt{\dot{x}^2 + \dot{y}^2})dt, \tag{3.28}$$

i.e. the cost $J(u)$ is the length of the projection of $q(t)$ onto the x-y plane.

Proof. Since $2\sqrt{\dot{x}^2 + \dot{y}^2} = \sqrt{(u_1 + u_2)^2} = |u_1 + u_2|$, it is enough to show $|u_1 + u_2| = |u_1| + |u_2|$ along a tight extremal. Tight extremals are composed of swing and straight segments. Over a swing segment one of the inputs is zero; for instance $u_1 = 0$ in which case $|u_1 + u_2| = |u_2| = |u_1| + |u_2|$. Over a straight segment $u_1 = u_2$ and $|u_1 + u_2| = 2|u_1| = |u_1| + |u_2|$. ∎

3.3.3 Characterization of Loose Extremals

The Pontryagin Maximum Principle does not give a restrictive enough extremal control law for loose extremals. In fact, the only constraint on loose extremals is that $u_1, -u_2 \in [-1, 0]$ or $u_1, -u_2 \in [0, 1]$. Thus, a variety of non-bang-bang controls generate various loose extremals. For instance, it can be verified that rotation round any point on the axle is a minimum wheel-rotation trajectory. In this section, we will first show that loose optimal trajectories can only cover a bounded region of the configuration space around the initial configuration. There may be different loose extremals that transfer the initial configuration to the goal configuration. In particular, there may exist different such loose extremals which have equal wheel rotation. Equivalence of wheel rotation defines equivalence classes of loose extremals. We will show in Lemma 3.8 that there exists a representative composed of rotation in place and swing segments with a known structure, in every equivalence class.

Lemma 3.5. *Let $q(t)$ be a loose extremal associated with the control $u(t)$, and let ϑ be the length of the projection of $q(t)$ onto $\mathbb{S}^1$; in other words,*

$$\vartheta = \int_0^T |\dot{\theta}| dt. \tag{3.29}$$

In this case we have $J(u) = b\vartheta$.

Proof. Since $2b|\dot{\theta}| = |u_1 - u_2|$, it is enough to show that $|u_1 - u_2| = |u_1| + |u_2|$ along a loose extremal. According to Lemma 3.3, $u_1 u_2 \leq 0$ along a loose extremal. Thus, $|u_1 u_2| = -u_1 u_2$ which means $(|u_1| + |u_2|)^2 = (u_1 - u_2)^2$. It is obvious then that $|u_1| + |u_2| = |u_1 - u_2|$. ■

Lemma 3.6. *Let $(q(t), u(t))$ be a loose trajectory-control pair that tranfers the initial configuration (x_0, y_0, θ_0) to the goal configuration (x_1, y_1, θ_1). It follows that $J(u) = b|\theta_1 - \theta_0 + 2k\pi|$ for some integer k. Furthermore, if $q(t)$ is optimal, then $J(u) \leq 5b\pi$.*

Proof. According to Lemma 3.5, the cost of a loose extremal is $b\vartheta$, in which ϑ is (3.29). In this case, $\vartheta = |\theta_1 - \theta_0 + 2k\pi|$ for some integer k and the cost is $J(u) = b|\theta_1 - \theta_0 + 2k\pi|$. For the second part, suppose $q(t)$ is optimal while $|\theta_1 - \theta_0 + 2k\pi| > 5\pi$. It can geometrically be shown that $\sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2} \leq 2bm$, in which m is an integer that satisfies the inequality $(m-1)\pi < |\theta_1 - \theta_0 + 2k\pi| \leq m\pi$. Since $|\theta_1 - \theta_0 + 2k\pi| > 5\pi$, we have $m \geq 6$. The cost of the trivial trajectory which is composed of rotation in place, going straight, and again rotation in place is not more than $2bm + b\pi$. Thus, we have $J(u) = b|\theta_1 - \theta_0 + 2k\pi| > b(m-1)\pi > 2bm + b\pi$ because $m \geq 6$. This is contradictory to the optimality of $q(t)$. ■

Corollary 3.7. *Starting from an initial configuration, loose optimal trajectories are of bounded cost and bounded reach in the x-y plane. We call such optimal extremals* type III.

Lemma 3.8. *Let $(q(t), u(t))$ be a loose optimal trajectory-control pair that tranfers the initial configuration q_0 to the goal configuration q_1. There exists a trajectory-control pair*

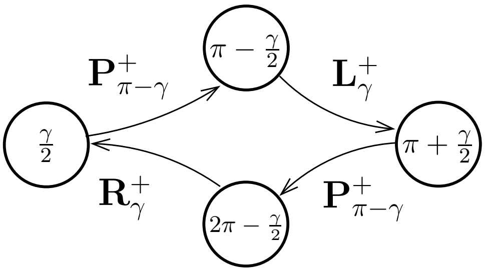


Figure 3.9: $\mathcal{E}_1$ provides a representative subclass of loose extremals in + direction.

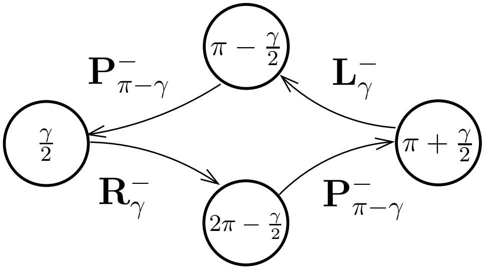


Figure 3.10: $\mathcal{E}_2$ provides a representative subclass of loose extremals in − direction.

$(\check{q}(t), \check{u}(t))$ transferring q_0 to q_1, in which $\check{u}$ is composed of a sequence of alternating rotation in place and swing segments in the same direction. Furthermore, $q(t)$ and $\check{q}(t)$ have the same wheel rotation, i.e. $J(u) = J(\check{u})$.

Sketch of proof. Look at the time-optimal trajectories for the system described in (2.1) with $u_1 \in [-1, 0], u_2 \in [0, 1]$ (our claim for the case in which $u_1 \in [0, 1], u_2 \in [-1, 0]$ follows from a similar argument). We know the time-optimal trajectories for this modified system exist because its input space is convex. Upon applying the Pontryagin Maximum Principle with the time as the cost functional, the extremals are composed of a sequence of rotation in place and swing segments. Let $(\check{q}(t), \check{u}(t))$ be the time optimal trajectory-control pair, i.e. $\check{u}$ is composed of a sequence of rotation in place and swing segments. Lemma 3.5 implies that $J(u) = b\vartheta$ and $J(\check{u}) = b\check{\vartheta}$, in which ϑ and $\check{\vartheta}$ are as in (3.29). Since $\vartheta \equiv \pm\check{\vartheta}$ up to a multiple of 2π, and Lemma 3.6 holds for $(q(t), u(t))$, we have $J(u) = J(\check{u})$ because otherwise, it can be verified that $\check{u}$ is not time optimal. ■

We use **P** to denote rotation in place. In order to present the representative subclass of loose extremals whose existence is established in Lemma 3.8, we define finite state machines

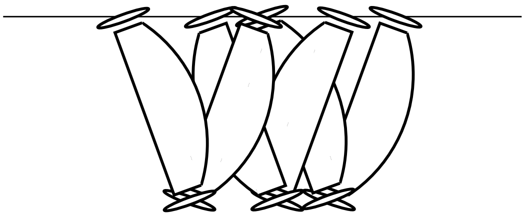


Figure 3.11: An $\mathcal{E}_1$ trajectory.

$\mathcal{E}_1$ and $\mathcal{E}_2$. Let $0 \leq \gamma \leq \pi$ and $Q = \{\frac{\gamma}{2}, \pi - \frac{\gamma}{2}, \pi + \frac{\gamma}{2}, 2\pi - \frac{\gamma}{2}\}$ be the set of states which represent the robot orientation. Let the input alphabet be $\Sigma = \{\mathbf{L}_\gamma^+, \mathbf{L}_\gamma^-, \mathbf{R}_\gamma^+, \mathbf{R}_\gamma^-, \mathbf{P}_{\pi-\gamma}^+, \mathbf{P}_{\pi-\gamma}^-\}$. Define $\mathcal{E}_1$ and $\mathcal{E}_2$ by the transition functions that are depicted in Figures 3.9 and 3.10 respectively. $\mathcal{E}_1$ provides a representative subclass of loose extremals in $+$ direction and $\mathcal{E}_2$ in $-$ direction.

3.4 Minimum Wheel-Rotation Trajectories

Eventually, in this section we give type I, II, and III minimum wheel-rotation trajectories up to symmetries. In Section 3.3.1 we described the symmetries of this problem. In the following we denote straight segment by **S**, swinging around right and left wheels by **R** and **L** respectively, and rotation in place by **P**. Directions are denoted by superscript $+$ and $-$ whenever it is required, otherwise it is constant throughout the trajectory. Forward and counter-clockwise are denoted by $+$, and backward and clockwise by $-$. Subscripts denote angles.

Proposition 3.9. *Any subpath of an optimal path is necessarily optimal.*

Proof. For otherwise, one gets a better path by substituting the optimal alternative for the subpath, which is a contradiction. ∎

We need to explicitly list only those minimum wheel-rotation trajectories that are maximal with respect to the subpath partial order. Other minimum wheel-rotation trajectories are subpaths of the listed ones and derivable from them. In other words, we will explicitly characterize only *maximally optimal* trajectories.

Table 3.1: Maximal minimum wheel-rotation trajectories sorted by symmetry class

	(A)	(B)
Base	$\mathbf{L}_{\alpha}^{-}\mathbf{R}_{\frac{\pi}{2}}^{-}\mathbf{S}^{+}\mathbf{R}_{\beta}^{-}$	$\mathbf{L}_{\alpha}^{-}\mathbf{R}_{\frac{\pi}{2}}^{-}\mathbf{S}^{+}\mathbf{L}_{\frac{\pi}{2}}^{+}\mathbf{R}_{\beta}^{+}$
$\mathcal{O}_1$	$\mathbf{R}_{\beta}^{-}\mathbf{S}^{+}\mathbf{R}_{\frac{\pi}{2}}^{-}\mathbf{L}_{\alpha}^{-}$	$\mathbf{R}_{\beta}^{+}\mathbf{L}_{\frac{\pi}{2}}^{+}\mathbf{S}^{+}\mathbf{R}_{\frac{\pi}{2}}^{-}\mathbf{L}_{\alpha}^{-}$
$\mathcal{O}_2$	$\mathbf{L}_{\alpha}^{+}\mathbf{R}_{\frac{\pi}{2}}^{+}\mathbf{S}^{-}\mathbf{R}_{\beta}^{+}$	$\mathbf{L}_{\alpha}^{+}\mathbf{R}_{\frac{\pi}{2}}^{+}\mathbf{S}^{-}\mathbf{L}_{\frac{\pi}{2}}^{-}\mathbf{R}_{\beta}^{-}$
$\mathcal{O}_3$	$\mathbf{R}_{\alpha}^{+}\mathbf{L}_{\frac{\pi}{2}}^{+}\mathbf{S}^{+}\mathbf{L}_{\beta}^{+}$	$\mathbf{R}_{\alpha}^{+}\mathbf{L}_{\frac{\pi}{2}}^{+}\mathbf{S}^{+}\mathbf{R}_{\frac{\pi}{2}}^{-}\mathbf{L}_{\beta}^{-}$
$\mathcal{O}_1 \circ \mathcal{O}_2$	$\mathbf{R}_{\beta}^{+}\mathbf{S}^{-}\mathbf{R}_{\frac{\pi}{2}}^{+}\mathbf{L}_{\alpha}^{+}$	$\mathbf{R}_{\beta}^{-}\mathbf{L}_{\frac{\pi}{2}}^{-}\mathbf{S}^{-}\mathbf{R}_{\frac{\pi}{2}}^{+}\mathbf{L}_{\alpha}^{+}$
$\mathcal{O}_1 \circ \mathcal{O}_3$	$\mathbf{L}_{\beta}^{+}\mathbf{S}^{+}\mathbf{L}_{\frac{\pi}{2}}^{+}\mathbf{R}_{\alpha}^{+}$	$\mathbf{L}_{\beta}^{-}\mathbf{R}_{\frac{\pi}{2}}^{-}\mathbf{S}^{+}\mathbf{L}_{\frac{\pi}{2}}^{+}\mathbf{R}_{\alpha}^{+}$
$\mathcal{O}_2 \circ \mathcal{O}_3$	$\mathbf{R}_{\alpha}^{-}\mathbf{L}_{\frac{\pi}{2}}^{-}\mathbf{S}^{-}\mathbf{L}_{\beta}^{-}$	$\mathbf{R}_{\alpha}^{-}\mathbf{L}_{\frac{\pi}{2}}^{-}\mathbf{S}^{-}\mathbf{R}_{\frac{\pi}{2}}^{+}\mathbf{L}_{\beta}^{+}$
$\mathcal{O}_1 \circ \mathcal{O}_2 \circ \mathcal{O}_3$	$\mathbf{L}_{\beta}^{-}\mathbf{S}^{-}\mathbf{L}_{\frac{\pi}{2}}^{-}\mathbf{R}_{\alpha}^{-}$	$\mathbf{L}_{\beta}^{+}\mathbf{R}_{\frac{\pi}{2}}^{+}\mathbf{S}^{-}\mathbf{L}_{\frac{\pi}{2}}^{-}\mathbf{R}_{\alpha}^{-}$
	$\alpha + \beta \leq \frac{\pi}{2}$	$\alpha + \beta \leq 2$

	(C)	(D)
Base	$\mathbf{L}_{\alpha}^{-}\mathbf{R}_{\gamma}^{-}\mathbf{L}_{\gamma}^{+}\mathbf{R}_{\beta}^{+}$	$\mathbf{L}_{\alpha}^{+}\mathbf{R}_{\gamma}^{-}\mathbf{L}_{\gamma}^{-}\mathbf{R}_{\beta}^{+}$
$\mathcal{O}_1$	$\mathbf{R}_{\beta}^{+}\mathbf{L}_{\gamma}^{+}\mathbf{R}_{\gamma}^{-}\mathbf{L}_{\alpha}^{-}$	$\mathbf{R}_{\beta}^{+}\mathbf{L}_{\gamma}^{-}\mathbf{R}_{\gamma}^{-}\mathbf{L}_{\alpha}^{+}$
$\mathcal{O}_2$	$\mathbf{L}_{\alpha}^{+}\mathbf{R}_{\gamma}^{+}\mathbf{L}_{\gamma}^{-}\mathbf{R}_{\beta}^{-}$	$\mathbf{L}_{\alpha}^{-}\mathbf{R}_{\gamma}^{+}\mathbf{L}_{\gamma}^{+}\mathbf{R}_{\beta}^{-}$
$\mathcal{O}_3$	$\mathbf{R}_{\alpha}^{+}\mathbf{L}_{\gamma}^{+}\mathbf{R}_{\gamma}^{-}\mathbf{L}_{\beta}^{-}$	$\mathbf{R}_{\alpha}^{-}\mathbf{L}_{\gamma}^{+}\mathbf{R}_{\gamma}^{+}\mathbf{L}_{\beta}^{-}$
$\mathcal{O}_1 \circ \mathcal{O}_2$	$\mathbf{R}_{\beta}^{-}\mathbf{L}_{\gamma}^{-}\mathbf{R}_{\gamma}^{+}\mathbf{L}_{\alpha}^{+}$	$\mathbf{R}_{\beta}^{-}\mathbf{L}_{\gamma}^{+}\mathbf{R}_{\gamma}^{+}\mathbf{L}_{\alpha}^{-}$
$\mathcal{O}_1 \circ \mathcal{O}_3$	$\mathbf{L}_{\beta}^{-}\mathbf{R}_{\gamma}^{-}\mathbf{L}_{\gamma}^{+}\mathbf{R}_{\alpha}^{+}$	$\mathbf{L}_{\beta}^{-}\mathbf{R}_{\gamma}^{+}\mathbf{L}_{\gamma}^{+}\mathbf{R}_{\alpha}^{-}$
$\mathcal{O}_2 \circ \mathcal{O}_3$	$\mathbf{R}_{\alpha}^{-}\mathbf{L}_{\gamma}^{-}\mathbf{R}_{\gamma}^{+}\mathbf{L}_{\beta}^{+}$	$\mathbf{R}_{\alpha}^{+}\mathbf{L}_{\gamma}^{-}\mathbf{R}_{\gamma}^{-}\mathbf{L}_{\beta}^{+}$
$\mathcal{O}_1 \circ \mathcal{O}_2 \circ \mathcal{O}_3$	$\mathbf{L}_{\beta}^{+}\mathbf{R}_{\gamma}^{+}\mathbf{L}_{\gamma}^{-}\mathbf{R}_{\alpha}^{-}$	$\mathbf{L}_{\beta}^{+}\mathbf{R}_{\gamma}^{-}\mathbf{L}_{\gamma}^{-}\mathbf{R}_{\alpha}^{+}$
	$\alpha, \beta < \gamma \leq \frac{\pi}{2}$	$\alpha, \beta < \gamma \leq \frac{\pi}{2}$

	(E)	(F)
Base	$\mathbf{R}_{\alpha}^{+}\mathbf{P}_{\gamma}^{+}\mathbf{L}_{\beta}^{+}$	$\mathbf{P}_{\alpha}^{+}\mathbf{R}_{\gamma}^{+}\mathbf{P}_{\beta}^{+}$
$\mathcal{O}_1$	$\mathbf{L}_{\beta}^{+}\mathbf{P}_{\gamma}^{+}\mathbf{R}_{\alpha}^{+}$	$\mathbf{P}_{\beta}^{+}\mathbf{R}_{\gamma}^{+}\mathbf{P}_{\alpha}^{+}$
$\mathcal{O}_2$	$\mathbf{R}_{\alpha}^{-}\mathbf{P}_{\gamma}^{-}\mathbf{L}_{\beta}^{-}$	$\mathbf{P}_{\alpha}^{-}\mathbf{R}_{\gamma}^{-}\mathbf{P}_{\beta}^{-}$
$\mathcal{O}_3$	$\mathbf{L}_{\alpha}^{-}\mathbf{P}_{\gamma}^{-}\mathbf{R}_{\beta}^{-}$	$\mathbf{P}_{\alpha}^{-}\mathbf{L}_{\gamma}^{-}\mathbf{P}_{\beta}^{-}$
$\mathcal{O}_1 \circ \mathcal{O}_2$	$\mathbf{L}_{\beta}^{-}\mathbf{P}_{\gamma}^{-}\mathbf{R}_{\alpha}^{-}$	$\mathbf{P}_{\beta}^{-}\mathbf{R}_{\gamma}^{-}\mathbf{P}_{\alpha}^{-}$
$\mathcal{O}_1 \circ \mathcal{O}_3$	$\mathbf{R}_{\beta}^{-}\mathbf{P}_{\gamma}^{-}\mathbf{L}_{\alpha}^{-}$	$\mathbf{P}_{\beta}^{-}\mathbf{L}_{\gamma}^{-}\mathbf{P}_{\alpha}^{-}$
$\mathcal{O}_2 \circ \mathcal{O}_3$	$\mathbf{L}_{\alpha}^{+}\mathbf{P}_{\gamma}^{+}\mathbf{R}_{\beta}^{+}$	$\mathbf{P}_{\alpha}^{+}\mathbf{L}_{\gamma}^{+}\mathbf{P}_{\beta}^{+}$
$\mathcal{O}_1 \circ \mathcal{O}_2 \circ \mathcal{O}_3$	$\mathbf{R}_{\beta}^{+}\mathbf{P}_{\gamma}^{+}\mathbf{L}_{\alpha}^{+}$	$\mathbf{P}_{\beta}^{+}\mathbf{L}_{\gamma}^{+}\mathbf{P}_{\alpha}^{+}$
	$\alpha + \gamma + \beta \leq \pi$	$\alpha + \gamma + \beta \leq \pi$

Lemma 3.4 implies that wheel-rotation is equal to the length of the curve that is traversed by the center of robot in the x-y plane along tight extremals. Since equations of motion of the differential-drive is the same as that of Reeds-Shepp car along a tight extremal, the center of robot in the x-y plane traverses a Reeds-Shepp curve along a tight minimum wheel-rotation trajectory. Here we use previous results about Reeds-Shepp curves in [161] to characterize tight minimum wheel-rotation trajectories.

Lemma 3.10. *If $\alpha > 0$ then $\mathbf{R}_\pi\mathbf{L}_\alpha$ is not minimum wheel-rotation.*

Proof. For any $\beta > 0$, we first show that $\mathbf{L}_\beta\mathbf{R}_\pi\mathbf{L}_\beta$ is not optimal. Observe that $\mathbf{L}^-_\beta\mathbf{R}^-_\pi\mathbf{L}^-_\beta$ has $(\pi + 2\beta)b$ wheel rotation. Let $e = 4(1 - \cos\beta)b$. The trajectory $\mathbf{R}^+_{\frac{\pi}{2}-\beta}\mathbf{S}^-_e\mathbf{R}^+_{\frac{\pi}{2}-\beta}$ has $(\pi - 2\beta)b + e$ wheel rotation. Since $1 - \cos\beta \leq \beta$ we must have $(\pi - 2\beta)b + e \leq (\pi + 2\beta)b$. Second, we show that $\mathbf{R}_\pi\mathbf{L}_\alpha$ is not optimal. Let $0 < \epsilon < \alpha$ be a small positive number such that $2(1 - \cos\epsilon) < \epsilon$. We know that such ϵ exists. Let $g = 4(1 - \cos\epsilon)b$. Consider the trajectory $\mathbf{L}^+_\epsilon\mathbf{R}^+_{\frac{\pi}{2}-\epsilon}\mathbf{S}^-_g\mathbf{R}^+_{\frac{\pi}{2}-\epsilon}$ which has the same end configuration as $\mathbf{R}_\pi\mathbf{L}_\epsilon$. However, it has less wheel rotation than $\mathbf{R}_\pi\mathbf{L}_\epsilon$ because $g < 2b\epsilon$. Since any subpath of an optimal path should be optimal, $\mathbf{R}_\pi\mathbf{L}_\alpha$ is not optimal. ■

Theorem 3.11. *A type I minimum wheel-rotation trajectory has one of the following forms:*

- $\mathbf{L}^-_\alpha\mathbf{R}^-_{\frac{\pi}{2}}\mathbf{S}^+\mathbf{R}^-_\beta$
- $\mathbf{L}^-_\zeta\mathbf{R}^-_{\frac{\pi}{2}}\mathbf{S}^+\mathbf{L}^+_{\frac{\pi}{2}}\mathbf{R}^+_\gamma$,

in which $\alpha + \beta \leq \frac{\pi}{2}$ and $\zeta + \gamma \leq 2$.

Proof. In Section 3.3.2 case 1, we showed that type I extremals are of the following forms:

- $(\mathbf{R}^-_\pi\mathbf{L}^-_\pi)^*\mathbf{R}^-_{\frac{\pi}{2}}\mathbf{S}^+\mathbf{R}^-_{\frac{\pi}{2}}(\mathbf{L}^-_\pi\mathbf{R}^-_\pi)^*$
- $(\mathbf{R}^-_\pi\mathbf{L}^-_\pi)^*\mathbf{R}^-_{\frac{\pi}{2}}\mathbf{S}^+\mathbf{L}^+_{\frac{\pi}{2}}(\mathbf{R}^+_\pi\mathbf{L}^+_\pi)^*$.

Lemma 3.10 shows that if $\eta > 0$ then $\mathbf{L}_\pi\mathbf{R}_\eta$ cannot be minimum wheel-rotation. It is enough to note that any subpath of an optimal path is necessarily optimal. Hence, the only possibilities are of the following form:

- $\mathbf{L}_{\alpha}^{-}\mathbf{R}_{\frac{\pi}{2}}^{-}\mathbf{S}^{+}\mathbf{R}_{\frac{\pi}{2}}^{-}\mathbf{L}_{\eta}^{-}$

- $\mathbf{L}_{\zeta}^{-}\mathbf{R}_{\frac{\pi}{2}}^{-}\mathbf{S}^{+}\mathbf{L}_{\frac{\pi}{2}}^{+}\mathbf{R}_{\gamma}^{+}$,

in which $\alpha, \eta, \zeta, \gamma < \pi$. Assume $\alpha > 0$. We claim that $\eta = 0$, because a path of type $\mathbf{R}^{+}\mathbf{S}^{-}\mathbf{R}^{+}$ is shorter than $\mathbf{L}_{\alpha}^{-}\mathbf{R}_{\frac{\pi}{2}}^{-}\mathbf{S}^{+}\mathbf{R}_{\frac{\pi}{2}}^{-}\mathbf{L}_{\eta}^{-}$. Hence, $\mathbf{L}_{\alpha}^{-}\mathbf{R}_{\frac{\pi}{2}}^{-}\mathbf{S}^{+}\mathbf{R}_{\beta}^{-}$ is possibly optimal in which $\beta \leq \frac{\pi}{2}$. If $\alpha > \frac{\pi}{2}$, then a path of type $\mathbf{R}^{+}\mathbf{L}_{\frac{\pi}{2}}^{+}\mathbf{S}^{+}\mathbf{R}^{-}$ is shorter than $\mathbf{L}_{\alpha}^{-}\mathbf{R}_{\frac{\pi}{2}}^{-}\mathbf{S}^{+}$. Thus, $\alpha, \beta, \zeta, \gamma \leq \frac{\pi}{2}$. Also, charaterization of Reeds-Shepp curves of type $C|CSC$ in [161] implies that $\alpha + \beta \leq \frac{\pi}{2}$. Finally, if $\zeta + \gamma > 2$, then $\mathbf{L}_{\frac{\pi}{2}-\zeta}^{+}\mathbf{S}^{-}\mathbf{R}_{\frac{\pi}{2}-\gamma}^{-}$ is shorter than $\mathbf{L}_{\zeta}^{-}\mathbf{R}_{\frac{\pi}{2}}^{-}\mathbf{S}^{+}\mathbf{L}_{\frac{\pi}{2}}^{+}\mathbf{R}_{\gamma}^{+}$. Hence, $\zeta + \gamma \leq 2$. For such an optimal trajectory see (A) and (B) in Figure 3.1. ■

Theorem 3.12. *A type II minimum wheel-rotation trajectory has one of the following forms:*

- $\mathbf{L}_{\alpha}^{-}\mathbf{R}_{\gamma}^{-}\mathbf{L}_{\gamma}^{+}\mathbf{R}_{\beta}^{+}$

- $\mathbf{L}_{\alpha}^{+}\mathbf{R}_{\gamma}^{-}\mathbf{L}_{\gamma}^{-}\mathbf{R}_{\beta}^{+}$,

in which $0 \leq \alpha, \beta \leq \gamma \leq \frac{\pi}{2}$.

Proof. In Section 3.3.2 case 3, we showed that type II extremals are of the form $(\mathbf{L}_{\gamma}^{-}\mathbf{R}_{\gamma}^{-}\mathbf{L}_{\gamma}^{+}\mathbf{R}_{\gamma}^{+})^{*}$. We prove that a trajectory containing two complete sets of four swings is not optimal, i.e. $\mathbf{R}_{\gamma}^{+}\mathbf{L}_{\gamma}^{-}\mathbf{R}_{\gamma}^{-}\mathbf{L}_{\gamma}^{+}\mathbf{R}_{\gamma}^{+}\mathbf{L}_{\gamma}^{-}\mathbf{R}_{\gamma}^{-}\mathbf{L}_{\gamma}^{+}$ is not optimal. In each set, the amount of robot displacement in x-y plane is $8b\sin^2\frac{\gamma}{2}$, in which γ is the angle of swings. If $0 < \gamma < \frac{\pi}{4}$, then let ζ be such that $\sin^2\frac{\zeta}{2} = 2\sin^2\frac{\gamma}{2}$. It follows that $\zeta < 2\gamma < \frac{\pi}{2}$. A type II extremal that is composed of four swings of angle ζ has less wheel rotation. If $\frac{\pi}{4} \leq \gamma \leq \frac{\pi}{2}$, then $b\pi + 16b\sin^2\frac{\gamma}{2} < 8b\gamma$, and the trivial trajectory which is composed of rotation in place, going straight, and again rotation in place gives less wheel rotation. A similar argument, based on what we just showed, proves that $\mathbf{L}_{\gamma}^{-}\mathbf{R}_{\gamma}^{-}\mathbf{L}_{\gamma}^{+}\mathbf{R}_{\gamma}^{+}\mathbf{L}_{\gamma}^{-}\mathbf{R}_{\gamma}^{-}\mathbf{L}_{\gamma}^{+}\mathbf{R}_{\gamma}^{+}$ is not minimum wheel-rotation either. Moreover, Lemma 3.4 implies that wheel-rotation is equal to the length of the curve that is traversed by the center of robot in the x-y plane along tight extremals. Since the center of robot in the x-y plane traverses a Reeds-Shepp curve along a tight minimum wheel-rotation trajectory, the only possibilities [161] are

- $\mathbf{L}^-_\alpha\mathbf{R}^-_\gamma\mathbf{L}^+_\gamma\mathbf{R}^+_\beta$
- $\mathbf{L}^+_\alpha\mathbf{R}^-_\gamma\mathbf{L}^-_\gamma\mathbf{R}^+_\beta$,

in which $\alpha, \beta \leq \gamma \leq \frac{\pi}{2}$. For such an optimal trajectory see (C) and (D) in Figure 3.1. ■

Lemma 3.13. *If $\alpha > 0$ then $\mathbf{P}_{\pi-\gamma}\mathbf{R}_\gamma\mathbf{P}_\alpha$ is not minimum wheel-rotation, in which $0 \leq \gamma \leq \pi$.*

Proof. It is enough to note that $\mathbf{P}^-_{\pi-\gamma}\mathbf{R}^-_\gamma\mathbf{P}^-_\alpha$ has $\pi + \alpha$ wheel rotation whereas $\mathbf{L}^+_\gamma\mathbf{P}^+_{\pi-\gamma-\alpha}$ has $\pi - \alpha$ wheel rotation. Since they connect the same initial and goal configurations, the former cannot be minimum wheel-rotation. ■

Lemma 3.14. *If $0 \leq \zeta, \eta \leq \gamma \leq \pi$ and $\zeta + \eta > \gamma$ then $\mathbf{R}_\zeta\mathbf{P}_{\pi-\gamma}\mathbf{L}_\eta$ is not minimum wheel-rotation.*

Proof. Suppose $\mathbf{R}_\zeta\mathbf{P}_{\pi-\gamma}\mathbf{L}_\eta$ is minimum wheel-rotation. Let $\delta = \gamma - \zeta$. By assumption we have $0 \leq \delta < \eta$. We replace the subpath $\mathbf{R}^-_\zeta\mathbf{P}^-_{\pi-\gamma}\mathbf{L}^-_\delta$ of $\mathbf{R}^-_\zeta\mathbf{P}^-_{\pi-\gamma}\mathbf{L}^-_\eta$ by an equivalent trajectory $\mathbf{L}^+_\delta\mathbf{P}^+_{\pi-\gamma}\mathbf{R}^+_\zeta$ to get $\mathbf{L}^+_\delta\mathbf{P}^+_{\pi-\gamma}\mathbf{R}^+_\zeta\mathbf{L}^-_{\eta-\delta}$. Boundary points and wheel rotation of this trajectory is equal to boundary points and wheel rotation of the original trajectory $\mathbf{R}^-_\zeta\mathbf{P}^-_{\pi-\gamma}\mathbf{L}^-_\eta$. Hence, $\mathbf{L}^+_\delta\mathbf{P}^+_{\pi-\gamma}\mathbf{R}^+_\zeta\mathbf{L}^-_{\eta-\delta}$ is a minimum wheel-rotation trajectory. In particular, it must satisfy the Pontryagin Maximum Principle. This is a contradiction because $\mathbf{L}^+_\delta\mathbf{P}^+_{\pi-\gamma}\mathbf{R}^+_\zeta\mathbf{L}^-_{\eta-\delta}$ is not an extremal. ■

Theorem 3.15. *A type III minimum wheel-rotation trajectory is one of the following forms:*

- $\mathbf{R}_\alpha\mathbf{P}_\gamma\mathbf{L}_\beta$
- $\mathbf{P}_\alpha\mathbf{R}_\gamma\mathbf{P}_\beta$*,*

in which $\alpha + \gamma + \beta \leq \pi$.

Proof. In Section 3.3.3, we showed for any loose extremal there is an equivalent trajectory which is composed of swing and rotation in place, i.e. $(\mathbf{R}_\gamma\mathbf{P}_{\pi-\gamma}\mathbf{L}_\gamma\mathbf{P}_{\pi-\gamma})^*$. Lemma 3.13 implies that a 4-piece trajectory of this type cannot be minimum wheel-rotation. Thus, the only possible type III minimum wheel-rotation trajectories are of the following forms:

Table 3.2: Complete list of minimum wheel-rotation trajectories

Trajectory	Range
$\mathbf{C}_\alpha\mathbf{P}_\gamma\mathbf{C}_\beta$	$\alpha+\gamma+\beta \leq \pi$
$\mathbf{P}_\alpha\mathbf{C}_\gamma\mathbf{P}_\beta$	$\alpha+\gamma+\beta \leq \pi$
$\mathbf{C}_\alpha \vert \mathbf{C}_\gamma\mathbf{C}_\beta$	$\alpha, \beta \leq \gamma \leq \frac{\pi}{2}$
$\mathbf{C}_\alpha\mathbf{C}_\gamma \vert \mathbf{C}_\beta$	$\alpha, \beta \leq \gamma \leq \frac{\pi}{2}$
$\mathbf{C}_\alpha\mathbf{C}_\gamma \vert \mathbf{C}_\gamma\mathbf{C}_\beta$	$\alpha, \beta \leq \gamma \leq \frac{\pi}{2}$
$\mathbf{C}_\alpha \vert \mathbf{C}_\gamma\mathbf{C}_\gamma \vert \mathbf{C}_\beta$	$\alpha, \beta \leq \gamma \leq \frac{\pi}{2}$
$\mathbf{C}_\alpha\mathbf{S}_d\mathbf{C}_\beta$	$\alpha, \beta \leq \frac{\pi}{2}$ and $0 \leq d$
$\mathbf{C}_\alpha\mathbf{C}_{\frac{\pi}{2}}\mathbf{S}_d\mathbf{C}_\beta$	$\alpha+\beta \leq \frac{\pi}{2}$ and $0 \leq d$
$\mathbf{C}_\alpha\mathbf{S}_d\mathbf{C}_{\frac{\pi}{2}}\mathbf{C}_\beta$	$\alpha+\beta \leq \frac{\pi}{2}$ and $0 \leq d$
$\mathbf{L}_\alpha\mathbf{R}_{\frac{\pi}{2}}\mathbf{S}_d\mathbf{L}_{\frac{\pi}{2}}\mathbf{R}_\beta$	$\alpha+\beta \leq 2$ and $0 \leq d$
$\mathbf{R}_\alpha\mathbf{L}_{\frac{\pi}{2}}\mathbf{S}_d\mathbf{R}_{\frac{\pi}{2}}\mathbf{L}_\beta$	$\alpha+\beta \leq 2$ and $0 \leq d$

$\mathbf{R}_\zeta\mathbf{P}_{\pi-\gamma}\mathbf{L}_\eta$ and $\mathbf{P}_\alpha\mathbf{R}_\gamma\mathbf{P}_\beta$, in which $\alpha, \beta \leq \pi-\gamma$ and $\zeta, \eta \leq \gamma$. If $\alpha+\gamma+\beta > \pi$ then $\mathbf{P}^-_\alpha\mathbf{R}^-_\gamma\mathbf{P}^-_\beta$ is not minimum wheel-rotation, because $\mathbf{P}^+_{\pi-\gamma-\alpha}\mathbf{R}^+_\gamma\mathbf{P}^+_{\pi-\gamma-\beta}$ is shorter. If $\zeta+\eta > \gamma$ then Lemma 3.14 proves that $\mathbf{R}_\zeta\mathbf{P}_{\pi-\gamma}\mathbf{L}_\eta$ is not minimum wheel-rotation. Hence, $\zeta+(\pi-\gamma)+\eta \leq \pi$, and by renaming parameters we obtain the result. For such an optimal trajectory see (E) and (F) in Figure 3.1. ■

Taking the symmetries in Section 3.3.1 into account, all the maximally optimal trajectories with their symmetric clones are given in Table 3.1. Since the symmetry operators $\mathcal{O}_1, \mathcal{O}_2$, and $\mathcal{O}_3$ commute, we do not need to worry about their order. Let $\mathbf{C}$ represent a swing, $\mathbf{L}$ or $\mathbf{R}$, and $|$ represent a change of direction. Let α, β, and γ be non-negative angles. A complete list of the words that describe all of 52 minimum wheel rotation trajectories is given in Table 3.2.

We include the following lemma to compare minimum wheel-rotation with optimal time:

Lemma 3.16. *Let $T^\star$ be the optimal time given in [13] and $J^\diamond$ the minimum wheel-rotation. It follows that $\frac{1}{2}T^\star \leq J^\diamond \leq T^\star$.*

3.5 Relation with the Reeds-Shepp car

Here we show that minimum time for the Reeds-Shepp car is equal to minimum wheel-rotation for the differential drive. It is enough to show the result for the convexified Reeds-Shepp car, because minimum time for the convexified Reeds-Shepp car is equal to minimum time for the Reeds-Shepp car [164]. Moreover, we show that minimum wheel-rotation paths for the differential drive are exactly minimum time paths for the convexified Reeds-Shepp car.

The convexified Reeds-Shepp car is the following system with the same configuration space as that of the differential drive $\mathcal{C} = \mathbb{R}^2 \times \mathbb{S}^1$:

$$\dot{q} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} v_1 \cos\theta \\ v_1 \sin\theta \\ \frac{v_2}{b} \end{pmatrix}, \tag{3.30}$$

in which $v_1, v_2 \in [-1, 1]$ are the inputs and b is the minimum turning radius. We denote the vector of inputs (v_1, v_2) by v.

Any differential drive trajectory is also a feasible trajectory for the convexified Reeds-Shepp car by the following input transformation

$$v_1 = \frac{u_1 + u_2}{2}, \tag{3.31}$$

$$v_2 = \frac{u_2 - u_1}{2}. \tag{3.32}$$

However, the inverse is

$$u_1 = v_1 - v_2, \tag{3.33}$$

$$u_2 = v_1 + v_2. \tag{3.34}$$

It is clear that the inverse is not a useful transformation because if for example $v_1 = v_2 = 1$ then $u_2 = 2 \notin [-1, 1]$. Hence, we need a more sophisticated analysis than a simple input transformation. We will show then that the optimal paths for the two problems are equivalent up to an input transformation and time reparametrization in the following two lemmas.

Lemma 3.17. *Let $q(s)$ be a trajectory of the differential drive defined on $[0,T]$ and associated with control $u(s) = (u_1(s), u_2(s))$, where u is piecewise constant and non-zero. There exists a time reparametrization $\tau : [0,T_1] \to [0,T]$, where $\tau(0) = 0$ and $\tau(T_1) = T$, such that $q(\tau(t))$ is an admissible path of the convexified Reeds-Shepp car defined on $[0,T_1]$. Moreover, T_1 is equal to the wheel-rotation of the differential drive on its trajectory $q(s)$.*

Proof. We need to show that a time reparametrization $\tau : [0,T_1] \to [0,T]$ and controls $v = (v_1, v_2) : [0,T_1] \to [-1,1]^2$ exist such that

$$\frac{d}{dt}q(\tau(t)) = \begin{pmatrix} v_1(t)\cos\theta(\tau(t)) \\ v_1(t)\sin\theta(\tau(t)) \\ \frac{v_2(t)}{b} \end{pmatrix}. \tag{3.35}$$

Moreover, we want $T_1 = J(u)$. In other words, we want

$$\int_0^{T_1} dt = \frac{1}{2}\int_0^{T_1} (|u_1(\tau(t))| + |u_2(\tau(t))|)\dot{\tau} dt. \tag{3.36}$$

Expanding the left handside of (3.35) we get

$$\frac{d}{dt}q(\tau(t)) = \dot{q}(\tau(t))\dot{\tau}(t). \tag{3.37}$$

Thus, (2.1), (2.2), and (3.37) imply that it is enough to have the following for (3.35) to hold:

$$v_1(t) = \frac{u_1(\tau(t)) + u_2(\tau(t))}{2}\dot{\tau}(t), \tag{3.38}$$

$$v_2(t) = \frac{u_2(\tau(t)) - u_1(\tau(t))}{2}\dot{\tau}(t). \tag{3.39}$$

For (3.36) to hold it is enough to have

$$1 = \frac{|u_1(\tau(t))| + |u_2(\tau(t))|}{2}\dot{\tau}(t). \tag{3.40}$$

Remember that $u_1(s)$ and $u_2(s)$ are given, and we need to find $\tau(t)$ with the above properties. Let τ be the solution of the following ordinary differential equation:

$$\dot{\tau} = \frac{2}{|u_1(\tau)| + |u_2(\tau)|}. \tag{3.41}$$

Equation (3.41) may not have a solution in general, because its right handside need not be Lipschitz in τ. Since u is assumed to be piecewise constant and non-zero, a solution τ exists for (3.41). Now let v_1 and v_2 be defined by (3.38) and (3.39). Equations (3.38), (3.39), and (3.41) imply that $v_1, v_2 \in [-1, 1]$. Thus, we showed existence of τ and v_1 and v_2 that satisfy (3.35). ■

Lemma 3.18. *Let $q(t)$ be a minimum time curve for the convexified Reeds-Shepp car, defined on $[0,T]$ and associated with control $v(t) = (v_1(t), v_2(t))$. There exists a time reparametrization $\sigma : [0,T_0] \to [0,T]$, where $\sigma(0) = 0$ and $\sigma(T_0) = T$, such that $q(\sigma(s))$ is an admissible trajectory of the differential-drive defined on $[0,T_0]$. Moreover, wheel-rotation of the differential drive on its trajectory $q(\sigma(s))$ is equal to T.*

Proof. We need to show that a time reparametrization $\sigma : [0,T_0] \to [0,T]$ and controls $u : [0,T_0] \to U$ exist such that

$$\frac{d}{ds}q(\sigma(s)) = u_1(s)f_1(q(\sigma(s))) + u_2(s)f_2(q(\sigma(s))), \tag{3.42}$$

where f_i's are defined in (2.2). Moreover, we seek a σ such that $J(u) = T$. In other words, we want

$$\frac{1}{2}\int_0^{T_0} (|u_1(s)| + |u_2(s)|)ds = \int_0^{T_0} \dot{\sigma}ds. \tag{3.43}$$

Expanding the left handside of (3.42) we get

$$\frac{d}{ds}q(\sigma(s)) = \dot{q}(\sigma(s))\dot{\sigma}(s). \tag{3.44}$$

In that case, (2.1), (2.2), and (3.44) imply that it is enough to have the following for (3.42) to hold:

$$u_1(s) = (v_1(\sigma(s)) - v_2(\sigma(s)))\dot{\sigma}(s), \tag{3.45}$$

$$u_2(s) = (v_1(\sigma(s)) + v_2(\sigma(s)))\dot{\sigma}(s). \tag{3.46}$$

In order to make $J(u) = T$, it is enough to have

$$\dot{\sigma}(s) = \frac{|u_1(s)| + |u_2(s)|}{2}. \tag{3.47}$$

Remember that $v_1(t)$ and $v_2(t)$ are given, and we need to find $\sigma(s)$ with the above properties. We will prove that the solution of the following differential equation is the desired σ:

$$\dot{\sigma} = \frac{1}{|v_1(\sigma)| + |v_2(\sigma)|}. \tag{3.48}$$

Since q is assumed to be a minimum time trajectory for the convexified Reeds-Shepp car, for all $t \in [0, T]$ we have one of the following cases:

1. $v_1(t) = \pm 1$, $v_2(t) = 0$, i.e. straight segment,
2. $v_1(t) = \pm 1$, $v_2(t) = \pm 1$, i.e. curve segment,
3. $v_1(t) \in [-1, 1]$, $v_2(t) = \pm 1$, i.e. three point turn.

It is clear then that (3.48) has a solution. Now let u_1 and u_2 be defined by (3.45) and (3.46). Equations (3.45), (3.46), and (3.48) imply that $u_1, u_2 \in [-1, 1]$. Finally, (3.47) follows from the fact that

$$|v_1(t) - v_2(t)| + |v_1(t) + v_2(t)| = 2 \tag{3.49}$$

in all the three cases above, and (3.45), (3.46), and $\dot{\sigma} > 0$. ■

Theorem 3.19. *Minimum time for the Reeds-Shepp car is equal to minimum wheel-rotation for the differential drive. Moreover, minimum wheel-rotation paths for the differential drive are exactly minimum time paths for the convexified Reeds-Shepp car.*

Proof. Let $q(s)$ be the minimum wheel-rotation path for the differential-drive associated with control u. Our analysis in previous sections proves that u is piecewise constant. Lemma 3.17 guarantees the existence of $q(\tau)$, an admissible path for the convexified Reeds-Shepp car, such that the duration of $q(\tau)$ is equal to wheel-rotation of $q(s)$. Lemma 3.18 implies that $q(\tau)$ has to be minimum time, because otherwise there exists a trajectory for the differential-drive with less wheel rotation than that of $q(s)$. In the same way if $q(t)$ is a minimum time path for the convexified Reeds-Shepp car, then $q(\sigma)$ in Lemma 3.18 has to be minimum wheel-rotation. It is a known fact that minimum time for the convexified Reeds-Shepp car is the same as minimum time for the Reeds-Shepp car [164]. ■

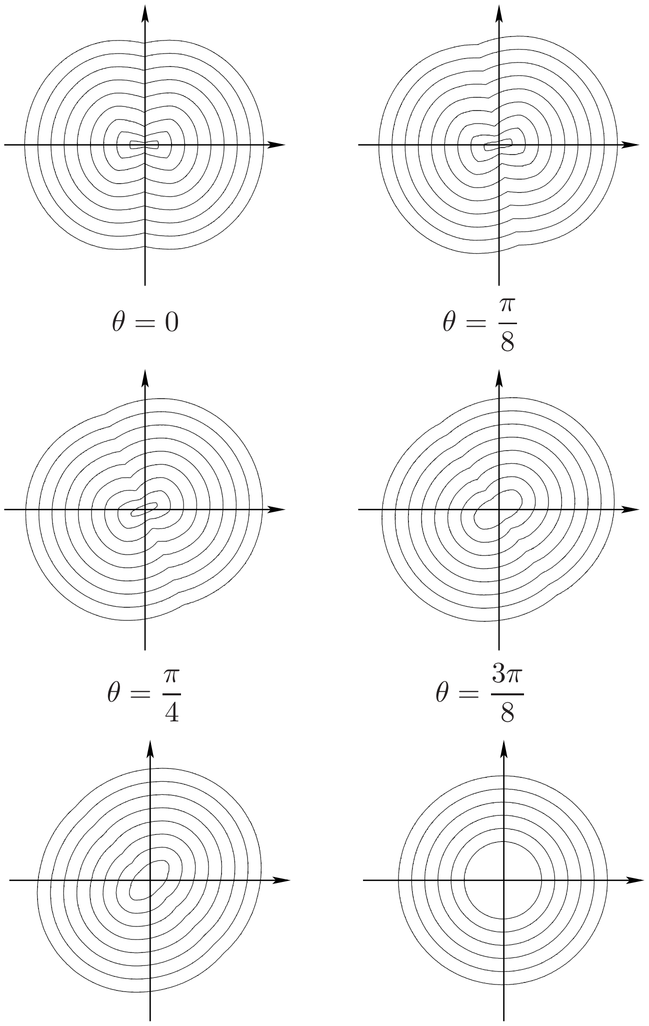
3.6 Cost-to-go Function

Level sets of the cost-to-go function for some goal orientations are presented in Figure 3.12. In computing the cost-to-go function, initial configuration is assumed to be $(0, 0, 0)$, and goal orientation θ is assumed to be $0, \frac{\pi}{8}, \frac{\pi}{4}, \frac{3\pi}{8}, \frac{\pi}{2}$, and π. Numerical computations verify that minimum wheel-rotation cost-to-go function is equal to the Reeds-Shepp cost-to-go function.

3.7 Optimal Control Synthesis

In this section, we give the cost and goal configuration of every minimum wheel-rotation trajectory in terms of its parameters. Therefore, finding the shortest path for every pair of initial and goal configurations reduces to solving systems of equations for the path parameters. Note that in the following, orientation of the robot θ must always be considered an element of $\mathbb{S}^1$. In other words, θ is evaluated mod 2π.

Let $q(t)$ be an arbitrary trajectory defined on $[0, T]$ corresponding to the input $u(t)$, and let $q(0) = (x_i, y_i, \theta_i)$. Let $\hat{q}(t)$ be the trajectory corresponding to the input $u(t)$ such that



$\theta = 0$ $\theta = \frac{\pi}{8}$

$\theta = \frac{\pi}{4}$ $\theta = \frac{3\pi}{8}$

$\theta = \frac{\pi}{2}$ $\theta = \pi$

Figure 3.12: Level sets of the cost-to-go function for $\theta = 0, \frac{\pi}{8}, \frac{\pi}{4}, \frac{3\pi}{8}, \frac{\pi}{2}$, and π

$\hat{q}(0) = (0, 0, 0)$. Suppose the goal configuration of $\hat{q}$ is (x, y, θ), i.e. $\hat{q}(T) = (x, y, \theta)$. In that case, the goal configuration of q is

$$\begin{aligned} x_g &= x_i + x\cos\theta_i - y\sin\theta_i && (3.50) \\ y_g &= y_i + x\sin\theta_i + y\cos\theta_i && (3.51) \\ \theta_g &= \theta_i + \theta, && (3.52) \end{aligned}$$

i.e. $q(T) = (x_g, y_g, \theta_g)$. Thus, we may assume without loss of generality that the initial

Table 3.3: $\alpha+\gamma+\beta \le \pi$

$\mathbf{C}_\alpha\mathbf{P}_\gamma\mathbf{C}_\beta$	$\mathbf{P}_\alpha\mathbf{C}_\gamma\mathbf{P}_\beta$	κ_1	κ_2	κ_3	c
$\mathbf{R}^+_\alpha\mathbf{P}^+_\gamma\mathbf{L}^+_\beta$	$\mathbf{P}^+_\alpha\mathbf{R}^+_\gamma\mathbf{P}^+_\beta$	α	$\alpha+\gamma$	$\alpha+\gamma+\beta$	b
$\mathbf{L}^+_\alpha\mathbf{P}^+_\gamma\mathbf{R}^+_\beta$	$\mathbf{P}^+_\alpha\mathbf{L}^+_\gamma\mathbf{P}^+_\beta$	α	$\alpha+\gamma$	$\alpha+\gamma+\beta$	$-b$
$\mathbf{R}^-_\alpha\mathbf{P}^-_\gamma\mathbf{L}^-_\beta$	$\mathbf{P}^-_\alpha\mathbf{R}^-_\gamma\mathbf{P}^-_\beta$	$-\alpha$	$-\alpha-\gamma$	$-\alpha-\gamma-\beta$	b
$\mathbf{L}^-_\alpha\mathbf{P}^-_\gamma\mathbf{R}^-_\beta$	$\mathbf{P}^-_\alpha\mathbf{L}^-_\gamma\mathbf{P}^-_\beta$	$-\alpha$	$-\alpha-\gamma$	$-\alpha-\gamma-\beta$	$-b$

configuration of the robot is $(0,0,0)$ throughout this section.

3.7.1 $\mathbf{C}_\alpha\mathbf{P}_\gamma\mathbf{C}_\beta$ and $\mathbf{P}_\alpha\mathbf{C}_\gamma\mathbf{P}_\beta$

In Table 3.3, the list of minimum wheel-rotation trajectories of type $\mathbf{C}_\alpha\mathbf{P}_\gamma\mathbf{C}_\beta$ and $\mathbf{P}_\alpha\mathbf{C}_\gamma\mathbf{P}_\beta$ can be found. The goal configuration of $\mathbf{C}_\alpha\mathbf{P}_\gamma\mathbf{C}_\beta$ is

$$x = -c(\sin\kappa_1+\sin\kappa_2-\sin\kappa_3) \tag{3.53}$$

$$y = c(\cos\kappa_1-1+\cos\kappa_2-\cos\kappa_3) \tag{3.54}$$

$$\theta = \kappa_3, \tag{3.55}$$

and the goal configuration of $\mathbf{P}_\alpha\mathbf{C}_\gamma\mathbf{P}_\beta$ is

$$x = c(\sin\kappa_1-\sin\kappa_2) \tag{3.56}$$

$$y = c(\cos\kappa_2-\cos\kappa_1) \tag{3.57}$$

$$\theta = \kappa_3, \tag{3.58}$$

in which $\kappa_1, \kappa_2, \kappa_3$, and c are the parameters in Table 3.3. Wheel-rotation of such trajectories is $\alpha+\gamma+\beta$.

Table 3.4: $\alpha, \beta \leq \gamma \leq \frac{\pi}{2}$

$\mathbf{C}_\alpha\vert\mathbf{C}_\gamma\mathbf{C}_\beta$	κ_1	κ_2	κ_3	c
$\mathbf{R}_\alpha^+\mathbf{L}_\gamma^-\mathbf{R}_\beta^-$	α	$\alpha-\gamma$	$\alpha-\gamma-\beta$	b
$\mathbf{L}_\alpha^+\mathbf{R}_\gamma^-\mathbf{L}_\beta^-$	α	$\alpha-\gamma$	$\alpha-\gamma-\beta$	$-b$
$\mathbf{R}_\alpha^-\mathbf{L}_\gamma^+\mathbf{R}_\beta^+$	$-\alpha$	$-\alpha+\gamma$	$-\alpha+\gamma+\beta$	b
$\mathbf{L}_\alpha^-\mathbf{R}_\gamma^+\mathbf{L}_\beta^+$	$-\alpha$	$-\alpha+\gamma$	$-\alpha+\gamma+\beta$	$-b$
$\mathbf{C}_\alpha\mathbf{C}_\gamma\vert\mathbf{C}_\beta$	κ_1	κ_2	κ_3	c
$\mathbf{R}_\alpha^+\mathbf{L}_\gamma^+\mathbf{R}_\beta^-$	α	$\alpha+\gamma$	$\alpha+\gamma-\beta$	b
$\mathbf{L}_\alpha^+\mathbf{R}_\gamma^+\mathbf{L}_\beta^-$	α	$\alpha+\gamma$	$\alpha+\gamma-\beta$	$-b$
$\mathbf{R}_\alpha^-\mathbf{L}_\gamma^-\mathbf{R}_\beta^+$	$-\alpha$	$-\alpha-\gamma$	$-\alpha-\gamma+\beta$	b
$\mathbf{L}_\alpha^-\mathbf{R}_\gamma^-\mathbf{L}_\beta^+$	$-\alpha$	$-\alpha-\gamma$	$-\alpha-\gamma+\beta$	$-b$

3.7.2 $\mathbf{C}_\alpha|\mathbf{C}_\gamma\mathbf{C}_\beta$ and $\mathbf{C}_\alpha\mathbf{C}_\gamma|\mathbf{C}_\beta$

In Table 3.4, the list of minimum wheel-rotation trajectories of type $\mathbf{C}_\alpha|\mathbf{C}_\gamma\mathbf{C}_\beta$ and $\mathbf{C}_\alpha\mathbf{C}_\gamma|\mathbf{C}_\beta$ can be found. The goal configuration of both $\mathbf{C}_\alpha|\mathbf{C}_\gamma\mathbf{C}_\beta$ and $\mathbf{C}_\alpha\mathbf{C}_\gamma|\mathbf{C}_\beta$ is

$$x = -c(2\sin\kappa_1 - 2\sin\kappa_2 + \sin\kappa_3) \tag{3.59}$$

$$y = c(2\cos\kappa_1 - 1 - 2\cos\kappa_2 + \cos\kappa_3) \tag{3.60}$$

$$\theta = \kappa_3, \tag{3.61}$$

in which $\kappa_1, \kappa_2, \kappa_3$, and c are the parameters in Table 3.4. Wheel-rotation of such trajectories is $\alpha + \gamma + \beta$.

3.7.3 $\mathbf{C}_\alpha\mathbf{C}_\gamma|\mathbf{C}_\gamma\mathbf{C}_\beta$ and $\mathbf{C}_\alpha|\mathbf{C}_\gamma\mathbf{C}_\gamma|\mathbf{C}_\beta$

In Table 3.5, the list of minimum wheel-rotation trajectories of type $\mathbf{C}_\alpha\mathbf{C}_\gamma|\mathbf{C}_\gamma\mathbf{C}_\beta$ and $\mathbf{C}_\alpha|\mathbf{C}_\gamma\mathbf{C}_\gamma|\mathbf{C}_\beta$ can be found. The goal configuration of $\mathbf{C}_\alpha\mathbf{C}_\gamma|\mathbf{C}_\gamma\mathbf{C}_\beta$ is

$$x = -c(4\sin\kappa_1 - 2\sin\kappa_2 - \sin\kappa_3) \tag{3.62}$$

$$y = c(4\cos\kappa_1 - 1 - 2\cos\kappa_2 - \cos\kappa_3) \tag{3.63}$$

$$\theta = \kappa_3, \tag{3.64}$$

Table 3.5: $\alpha, \beta \leq \gamma \leq \frac{\pi}{2}$

$\mathbf{C}_\alpha\mathbf{C}_\gamma \vert \mathbf{C}_\gamma\mathbf{C}_\beta$	κ_1	κ_2	κ_3	c
$\mathbf{R}^+_\alpha\mathbf{L}^+_\gamma\mathbf{R}^-_\gamma\mathbf{L}^-_\beta$	α	$\alpha+\gamma$	$\alpha-\beta$	b
$\mathbf{L}^+_\alpha\mathbf{R}^+_\gamma\mathbf{L}^-_\gamma\mathbf{R}^-_\beta$	α	$\alpha+\gamma$	$\alpha-\beta$	$-b$
$\mathbf{R}^-_\alpha\mathbf{L}^-_\gamma\mathbf{R}^+_\gamma\mathbf{L}^+_\beta$	$-\alpha$	$-\alpha-\gamma$	$-\alpha+\beta$	b
$\mathbf{L}^-_\alpha\mathbf{R}^-_\gamma\mathbf{L}^+_\gamma\mathbf{R}^+_\beta$	$-\alpha$	$-\alpha-\gamma$	$-\alpha+\beta$	$-b$

$\mathbf{C}_\alpha \vert \mathbf{C}_\gamma\mathbf{C}_\gamma \vert \mathbf{C}_\beta$	κ_1	κ_2	κ_3	κ_4	c
$\mathbf{R}^+_\alpha\mathbf{L}^-_\gamma\mathbf{R}^-_\gamma\mathbf{L}^+_\beta$	α	$\alpha-\gamma$	$\alpha-2\gamma$	$\alpha-2\gamma+\beta$	b
$\mathbf{L}^+_\alpha\mathbf{R}^-_\gamma\mathbf{L}^-_\gamma\mathbf{R}^+_\beta$	α	$\alpha-\gamma$	$\alpha-2\gamma$	$\alpha-2\gamma+\beta$	$-b$
$\mathbf{R}^-_\alpha\mathbf{L}^+_\gamma\mathbf{R}^+_\gamma\mathbf{L}^-_\beta$	$-\alpha$	$-\alpha+\gamma$	$-\alpha+2\gamma$	$-\alpha+2\gamma-\beta$	b
$\mathbf{L}^-_\alpha\mathbf{R}^+_\gamma\mathbf{L}^+_\gamma\mathbf{R}^-_\beta$	$-\alpha$	$-\alpha+\gamma$	$-\alpha+2\gamma$	$-\alpha+2\gamma-\beta$	$-b$

and the goal configuration of $\mathbf{C}_\alpha|\mathbf{C}_\gamma\mathbf{C}_\gamma|\mathbf{C}_\beta$ is

$$x = -c(2\sin\kappa_1 - 2\sin\kappa_2 + 2\sin\kappa_3 - \sin\kappa_4) \tag{3.65}$$

$$y = c(2\cos\kappa_1 - 1 - 2\cos\kappa_2 + 2\cos\kappa_3 - \cos\kappa_4) \tag{3.66}$$

$$\theta = \kappa_4, \tag{3.67}$$

in which $\kappa_1, \kappa_2, \kappa_3, \kappa_4$, and c are the parameters in Table 3.5. Wheel-rotation of such trajectories is $\alpha + 2\gamma + \beta$.

3.7.4 $\mathbf{C}_\alpha\mathbf{S}_d\mathbf{C}_\beta$

In Table 3.6, the list of minimum wheel-rotation trajectories of type $\mathbf{C}_\alpha\mathbf{S}_d\mathbf{C}_\beta$ can be found. The goal configuration of $\mathbf{C}_\alpha\mathbf{S}_d\mathbf{C}_\beta$ is

$$x = c_1\cos\kappa_1 + c_2\sin\kappa_1 + c_3\sin\kappa_2 \tag{3.68}$$

$$y = c_1\sin\kappa_1 - c_2\cos\kappa_1 - c_3\cos\kappa_2 + c_4 \tag{3.69}$$

$$\theta = \kappa_2, \tag{3.70}$$

in which $\kappa_1, \kappa_2, c_1, c_2, c_3$, and c_4 are the parameters in Table 3.6. Wheel-rotation of such trajectories is $\alpha + d + \beta$.

Table 3.6: $\alpha, \beta \leq \frac{\pi}{2}$ and $d \geq 0$

$\mathbf{C}_\alpha\mathbf{S}_d\mathbf{C}_\beta$	κ_1	κ_2	c_1	c_2	c_3	c_4
$\mathbf{R}^+_\alpha\mathbf{S}^-_d\mathbf{R}^+_\beta$	α	$\alpha+\beta$	$-d$	0	$-b$	$-b$
$\mathbf{L}^+_\alpha\mathbf{S}^+_d\mathbf{L}^+_\beta$	α	$\alpha+\beta$	d	0	b	b
$\mathbf{R}^+_\alpha\mathbf{S}^-_d\mathbf{L}^-_\beta$	α	$\alpha-\beta$	$-d$	$-2b$	b	$-b$
$\mathbf{L}^+_\alpha\mathbf{S}^+_d\mathbf{R}^-_\beta$	α	$\alpha-\beta$	d	$2b$	$-b$	b
$\mathbf{R}^-_\alpha\mathbf{S}^+_d\mathbf{R}^-_\beta$	$-\alpha$	$-\alpha-\beta$	d	0	$-b$	$-b$
$\mathbf{L}^-_\alpha\mathbf{S}^-_d\mathbf{L}^-_\beta$	$-\alpha$	$-\alpha-\beta$	$-d$	0	b	b
$\mathbf{R}^-_\alpha\mathbf{S}^+_d\mathbf{L}^+_\beta$	$-\alpha$	$-\alpha+\beta$	d	$-2b$	b	$-b$
$\mathbf{L}^-_\alpha\mathbf{S}^-_d\mathbf{R}^+_\beta$	$-\alpha$	$-\alpha+\beta$	$-d$	$2b$	$-b$	b

3.7.5 $\mathbf{C}_\alpha\mathbf{C}_{\frac{\pi}{2}}\mathbf{S}_d\mathbf{C}_\beta$ and $\mathbf{C}_\alpha\mathbf{S}_d\mathbf{C}_{\frac{\pi}{2}}\mathbf{C}_\beta$

In Table 3.7, the list of minimum wheel-rotation trajectories of type $\mathbf{C}_\alpha\mathbf{C}_{\frac{\pi}{2}}\mathbf{S}_d\mathbf{C}_\beta$ and $\mathbf{C}_\alpha\mathbf{S}_d\mathbf{C}_{\frac{\pi}{2}}\mathbf{C}_\beta$ can be found. The goal configuration of such trajectories is

$$x = c_1 \sin\kappa_1 + c_2 \cos\kappa_1 + c_3 \sin\kappa_2 \tag{3.71}$$

$$y = -c_1 \cos\kappa_1 + c_2 \sin\kappa_1 - c_3 \cos\kappa_2 + c_4 \tag{3.72}$$

$$\theta = \kappa_2, \tag{3.73}$$

in which $\kappa_1, \kappa_2, c_1, c_2, c_3$, and c_4 are the parameters in Table 3.7. Wheel-rotation of such trajectories is $\alpha + \frac{\pi}{2} + d + \beta$.

3.7.6 $\mathbf{L}_\alpha\mathbf{R}_{\frac{\pi}{2}}\mathbf{S}_d\mathbf{L}_{\frac{\pi}{2}}\mathbf{R}_\beta$ and $\mathbf{R}_\alpha\mathbf{L}_{\frac{\pi}{2}}\mathbf{S}_d\mathbf{R}_{\frac{\pi}{2}}\mathbf{L}_\beta$

In Table 3.8, the list of minimum wheel-rotation trajectories of type $\mathbf{L}_\alpha\mathbf{R}_{\frac{\pi}{2}}\mathbf{S}_d\mathbf{L}_{\frac{\pi}{2}}\mathbf{R}_\beta$ and $\mathbf{R}_\alpha\mathbf{L}_{\frac{\pi}{2}}\mathbf{S}_d\mathbf{R}_{\frac{\pi}{2}}\mathbf{L}_\beta$ can be found. The goal configuration of such trajectories is

$$x = c_1 \sin\kappa_1 + c_2 \cos\kappa_1 + c_3 \sin\kappa_2 \tag{3.74}$$

$$y = -c_1 \cos\kappa_1 + c_2 \sin\kappa_1 - c_3 \cos\kappa_2 - c_3 \tag{3.75}$$

$$\theta = \kappa_2, \tag{3.76}$$

Table 3.7: $d \geq 0$

Range	$\mathbf{C}_\alpha \mathbf{C}_{\frac{\pi}{2}} \mathbf{S}_d \mathbf{C}_\beta$	κ_1	κ_2	c_1	c_2	c_3	c_4
$\alpha, \beta \leq \frac{\pi}{2}$	$\mathbf{R}_\alpha^+ \mathbf{L}_{\frac{\pi}{2}}^+ \mathbf{S}_d^+ \mathbf{R}_\beta^-$	α	$\alpha + \frac{\pi}{2} - \beta$	$-2b - d$	$2b$	$-b$	$-b$
$\alpha, \beta \leq \frac{\pi}{2}$	$\mathbf{L}_\alpha^+ \mathbf{R}_{\frac{\pi}{2}}^+ \mathbf{S}_d^- \mathbf{L}_\beta^-$	α	$\alpha + \frac{\pi}{2} - \beta$	$2b + d$	$-2b$	b	b
$\alpha + \beta \leq \frac{\pi}{2}$	$\mathbf{R}_\alpha^+ \mathbf{L}_{\frac{\pi}{2}}^+ \mathbf{S}_d^+ \mathbf{L}_\beta^+$	α	$\alpha + \frac{\pi}{2} + \beta$	$-2b - d$	0	b	$-b$
$\alpha + \beta \leq \frac{\pi}{2}$	$\mathbf{L}_\alpha^+ \mathbf{R}_{\frac{\pi}{2}}^+ \mathbf{S}_d^- \mathbf{R}_\beta^+$	α	$\alpha + \frac{\pi}{2} + \beta$	$2b + d$	0	$-b$	b
$\alpha, \beta \leq \frac{\pi}{2}$	$\mathbf{R}_\alpha^- \mathbf{L}_{\frac{\pi}{2}}^- \mathbf{S}_d^- \mathbf{R}_\beta^+$	$-\alpha$	$-\alpha - \frac{\pi}{2} + \beta$	$-2b - d$	$-2b$	$-b$	$-b$
$\alpha, \beta \leq \frac{\pi}{2}$	$\mathbf{L}_\alpha^- \mathbf{R}_{\frac{\pi}{2}}^- \mathbf{S}_d^+ \mathbf{L}_\beta^+$	$-\alpha$	$-\alpha - \frac{\pi}{2} + \beta$	$2b + d$	$2b$	b	b
$\alpha + \beta \leq \frac{\pi}{2}$	$\mathbf{R}_\alpha^- \mathbf{L}_{\frac{\pi}{2}}^- \mathbf{S}_d^- \mathbf{L}_\beta^-$	$-\alpha$	$-\alpha - \frac{\pi}{2} - \beta$	$-2b - d$	0	b	$-b$
$\alpha + \beta \leq \frac{\pi}{2}$	$\mathbf{L}_\alpha^- \mathbf{R}_{\frac{\pi}{2}}^- \mathbf{S}_d^+ \mathbf{R}_\beta^-$	$-\alpha$	$-\alpha - \frac{\pi}{2} - \beta$	$2b + d$	0	$-b$	b
Range	$\mathbf{C}_\alpha \mathbf{S}_d \mathbf{C}_{\frac{\pi}{2}} \mathbf{C}_\beta$	κ_1	κ_2	c_1	c_2	c_3	c_4
$\alpha, \beta \leq \frac{\pi}{2}$	$\mathbf{R}_\alpha^+ \mathbf{S}_d^- \mathbf{L}_{\frac{\pi}{2}}^- \mathbf{R}_\beta^-$	α	$\alpha - \frac{\pi}{2} - \beta$	$-2b$	$-2b - d$	$-b$	$-b$
$\alpha, \beta \leq \frac{\pi}{2}$	$\mathbf{L}_\alpha^+ \mathbf{S}_d^+ \mathbf{R}_{\frac{\pi}{2}}^- \mathbf{L}_\beta^-$	α	$\alpha - \frac{\pi}{2} - \beta$	$2b$	$2b + d$	b	b
$\alpha + \beta \leq \frac{\pi}{2}$	$\mathbf{R}_\alpha^+ \mathbf{S}_d^- \mathbf{R}_{\frac{\pi}{2}}^+ \mathbf{L}_\beta^+$	α	$\alpha + \frac{\pi}{2} + \beta$	0	$-2b - d$	b	$-b$
$\alpha + \beta \leq \frac{\pi}{2}$	$\mathbf{L}_\alpha^+ \mathbf{S}_d^+ \mathbf{L}_{\frac{\pi}{2}}^+ \mathbf{R}_\beta^+$	α	$\alpha + \frac{\pi}{2} + \beta$	0	$2b + d$	$-b$	b
$\alpha, \beta \leq \frac{\pi}{2}$	$\mathbf{R}_\alpha^- \mathbf{S}_d^+ \mathbf{L}_{\frac{\pi}{2}}^+ \mathbf{R}_\beta^+$	$-\alpha$	$-\alpha + \frac{\pi}{2} + \beta$	$-2b$	$2b + d$	$-b$	$-b$
$\alpha, \beta \leq \frac{\pi}{2}$	$\mathbf{L}_\alpha^- \mathbf{S}_d^- \mathbf{R}_{\frac{\pi}{2}}^+ \mathbf{L}_\beta^+$	$-\alpha$	$-\alpha + \frac{\pi}{2} + \beta$	$2b$	$-2b - d$	b	b
$\alpha + \beta \leq \frac{\pi}{2}$	$\mathbf{R}_\alpha^- \mathbf{S}_d^+ \mathbf{R}_{\frac{\pi}{2}}^- \mathbf{L}_\beta^-$	$-\alpha$	$-\alpha - \frac{\pi}{2} - \beta$	0	$2b + d$	b	$-b$
$\alpha + \beta \leq \frac{\pi}{2}$	$\mathbf{L}_\alpha^- \mathbf{S}_d^- \mathbf{L}_{\frac{\pi}{2}}^- \mathbf{R}_\beta^-$	$-\alpha$	$-\alpha - \frac{\pi}{2} - \beta$	0	$-2b - d$	$-b$	b

Table 3.8: $\alpha + \beta < 2$ and $d \geq 0$

$\mathbf{C}_\alpha \mathbf{C}_{\frac{\pi}{2}} \mathbf{S}_d \mathbf{C}_{\frac{\pi}{2}} \mathbf{C}_\beta$	κ_1	κ_2	c_1	c_2	c_3
$\mathbf{R}^+_\alpha \mathbf{L}^+_{\frac{\pi}{2}} \mathbf{S}^+_d \mathbf{R}^-_{\frac{\pi}{2}} \mathbf{L}^-_\beta$	α	$\alpha - \beta$	$-4b - d$	$2b$	b
$\mathbf{L}^+_\alpha \mathbf{R}^+_{\frac{\pi}{2}} \mathbf{S}^-_d \mathbf{L}^-_{\frac{\pi}{2}} \mathbf{R}^-_\beta$	α	$\alpha - \beta$	$4b + d$	$-2b$	$-b$
$\mathbf{R}^-_\alpha \mathbf{L}^-_{\frac{\pi}{2}} \mathbf{S}^-_d \mathbf{R}^+_{\frac{\pi}{2}} \mathbf{L}^+_\beta$	$-\alpha$	$-\alpha + \beta$	$-4b - d$	$-2b$	b
$\mathbf{L}^-_\alpha \mathbf{R}^-_{\frac{\pi}{2}} \mathbf{S}^+_d \mathbf{L}^+_{\frac{\pi}{2}} \mathbf{R}^+_\beta$	$-\alpha$	$-\alpha + \beta$	$4b + d$	$2b$	$-b$

in which $\kappa_1, \kappa_2, c_1, c_2$, and c_3 are the parameters in Table 3.8. Wheel-rotation of such trajectories is $\alpha + \pi + d + \beta$.

3.8 Summary

By applying the Pontryagin Maximum Principle [141] and developing geometric arguments, we derived necessary optimality conditions which helped to rule out non-optimal trajectories. The remaining trajectories form 28 different maximally optimal trajectories, which were listed in Table 3.1. A complete list of words that describe all of 52 minimum wheel-rotation trajectories was given in Table 3.2. We also proved that minimum wheel-rotation for the differential drive is equal to minimum time for the Reeds-Shepp car. Moreover, minimum wheel-rotation paths for the differential drive are exactly minimum time paths for the convexified Reeds-Shepp car. However, it is currently unknown whether there is a simpler way to show this equivalence. Based on the characterization of minimum wheel-rotation trajectories, a method to further determine the applicable trajectory for every pair of initial and goal configurations was presented in Section 3.7.

Chapter 4

Minimum Wheel-Rotation Paths for a Differential-Drive Disc Among Convex Obstacles

Every minimum wheel-rotation trajectory among obstacles is composed of two kinds of subtrajectories: on the boundary of obstacles and in the interior of collision-free space. Chapter 3 presented the first step of our approach to finding minimum wheel-rotation trajectories for a differential drive disc among convex obstacles. In this chapter, we present the remaining three steps: 1) Characterization of minimum wheel-rotation trajectories on the boundary of obstacle region using the Pontryagin Maximum Principle, 2) Characterization of intersection points between free and boundary minimum wheel-rotation trajectories using the Pontryagin Jump Condition [141], and 3) Definition of a nonholonomic bitangency graph in which the solution is sought. We show those subtrajectories that lie in the interior of collision-free space are tangent to the obstacles at both ends. The bitangency condition results in a nonholonomic bitangency graph which is used to find minimum wheel-rotation trajectories. Vertices of the nonholonomic bitangency graph are points on the obstacle boundary, and its edges are minimum wheel-rotation segments, each of which is either thoroughly on the obstacle boundary or bitangent. In general, the nonholonomic bitangency graph is a 2-dimensional subset of the 3-dimensional configuration space. For each shortest path query, initial and goal configurations are appropriately appended to the graph, and a continuous optimization or a search on parts or all of the graph gives the solution.

4.1 Related Work

Agarwal et al. obtained an algorithm for computing shortest paths for the Reeds-Shepp car amongst moderate obstacles [2]. An obstacle is said to be *moderate* if it is convex and its boundary is a differentiable curve whose curvature is everywhere not more than 1. Their algorithm computes a collision-free path for the Reeds-Shepp car whose length is either optimal or within an additive constant of optimal. To compute the solution, their algorithm starts with the Euclidean shortest path among obstacles. To adjust initial and goal orientations, a beginning and terminal segment are concatenated to it. Through a shortening process, their algorithm obtains the solution. Their algorithm computes approximate solutions, whereas our framework considers exact solutions. Their approach is specific to the Reeds-Shepp car, and cannot be utilized for other nonholonomic systems, whereas our framework covers a wide class of problems. In a different approach, Desaulniers et al. gave an approximation algorithm to compute a shortest path for the Reeds-Shepp car among polygonal obstacles by decomposing the space into polygonal regions and discretizing the boundaries of regions [54].

Our problem is essentially similar, but not identical, to the one Agarwal et al. study. It is similar because the configuration space obstacles are moderate in our problem; see Proposition 2.1. In addition, minimum time, equivalently minimum length, for the Reeds-Shepp car is equal to minimum wheel-rotation for the differential drive; see Section 3.5. However, the two problems are not identical because the differential drive is a different system from the Reeds-Shepp car, to the extent that the shortest paths for the Reeds-Shepp car need not always exist among general obstacles [53], whereas minimum wheel-rotation paths for the differential drive always exist for any kind of obstacles; see Section 2.2.1.

The visibility graph is one of the main ingredients of our framework, which was reviewed in Section 2.4. Figure 4.1 illustrates the visibility graph of three circular obstacles in the plane. In this example, the edges of the graph are straight lines in the interior of the free space and arcs on the obstacle boundary. There are no dynamic constraints and the metric is Euclidean. To find the shortest path from point A_1 to point A_2, one finds the tangents from A_i to the obstacles, and searches for the shortest path from A_1 to A_2 in the resulting graph which is shown in Figure 4.2.

The approach that we use to find shortest paths is similar in spirit to the visibility graph method. Edges of the shown visibility graph, in Figure 4.1, are Euclidean shortest paths.

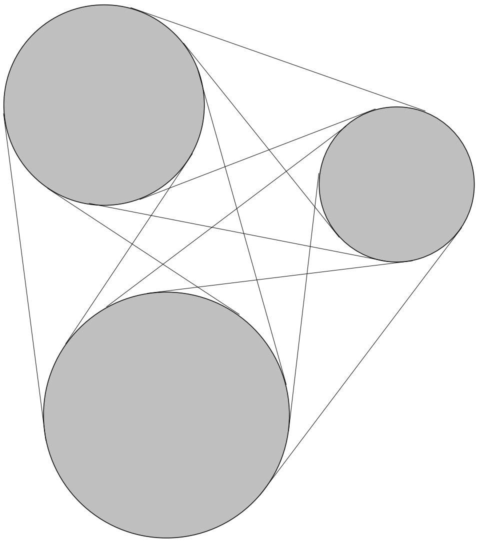


Figure 4.1: Visibility graph of three circular obstacles.

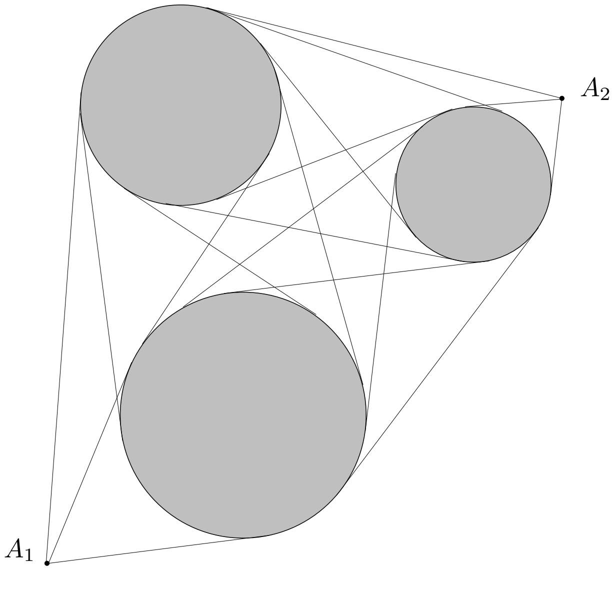


Figure 4.2: Finding the shortest path from A_1 to A_2 using the visibility graph.

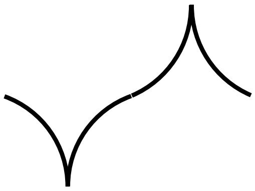


Figure 4.3: A sample minimum wheel-rotation path.

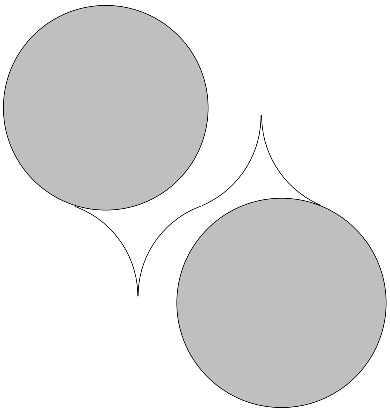


Figure 4.4: A sample edge of the nonholonomic bitangency graph.

We propose nonholonomic bitangency graph whose edges are nonholonomic shortest paths. Figure 4.3 illustrates a sample minimum wheel-rotation path for the differential drive in an unobstructed environment, and Figure 4.4 illustrates a sample edge of the nonholonomic bitangency graph.

Minimum wheel-rotation trajectories among obstacles consist of a finite number of segments each of which is either in the interior of the free region or on the obstacle boundary. Those segments that are in the interior of the free region were characterized in Chapter 3. In the following section, we apply the Pontryagin Maximum Principle in $\partial\mathcal{C}_{free}$ to prune out some non-optimal boundary segments.

4.2 Minimum Wheel-Rotation Paths on the Obstacle Boundary

Let $q(t)$ be an optimal trajectory defined on $[0, T]$ and associated with control $u(t)$. Assume $q([0, T]) \subset \partial\mathcal{C}_{free}$. Let $g\colon \mathcal{C} \to \mathbb{R}$ be a real-valued smooth function such that $g(q) = 0$ locally defines $\partial\mathcal{C}_{free}$. Note that $\partial g/\partial\theta = 0$ since the boundary components are $\partial P_i \times \mathbb{S}^1$. Let H be the Hamiltonian in Section 3.2.1. Let

$$p(q,u) = \langle \frac{\partial g}{\partial q}, f(q,u)\rangle. \tag{4.1}$$

The constrained Pontryagin Maximum Principle [141], which was summarized in Section 2.3, shows that there exists a constant $\lambda_0 \leq 0$, a vector-valued adjoint function $\lambda(t)$, that is nonzero if $\lambda_0 = 0$, and a real-valued function $\eta(t)$ with the following properties along $q(t)$:

$$\dot{\lambda} = -\frac{\partial H}{\partial q} + \eta(t)\frac{\partial p}{\partial q}, \tag{4.2}$$

$$H(\lambda(t), q(t), u(t)) = \max_{z\in U} H(\lambda(t), q(t), z), \tag{4.3}$$

$$H(\lambda(t), q(t), u(t)) \equiv 0. \tag{4.4}$$

Let the switching functions $\varphi_i(t)$ be defined in (3.5). Remember that the control law given by the Pontryagin Maximum Principle is

$$u_i(t) \in \begin{cases} [0,1] & \text{if} \quad \varphi_i(t) = 1 \\ \{0\} & \text{if} \quad |\varphi_i(t)| < 1 \\ [-1,0] & \text{if} \quad \varphi_i(t) = -1 \end{cases} \tag{4.5}$$

along an optimal trajectory. For details of this analysis see Lemma 3.3.

Lemma 4.1. *Let $q(t)$ be an optimal trajectory defined on $[0,T]$ such that $q([0,T]) \subset \partial\mathcal{C}_{free}$. In that case, $|\varphi_1(t)| = |\varphi_2(t)| = 1$ for $t \in [0,T]$.*

Proof. On the contrary, suppose $|\varphi_i(t_0)| < 1$ for some $t_0 \in [0,T]$ and some $i = 1,2$. Let j be the index of obstacle, i.e. $(x(t_0), y(t_0)) \in \partial P_j$ where $q(t_0) = (x(t_0), y(t_0), \theta(t_0))$. Since $\varphi_i(t)$ is continuous on $[0,T]$, there exists $\epsilon > 0$ such that $|\varphi_i(t)| < 1$ for $t \in [t_0 - \epsilon, t_0 + \epsilon]$. Thus, the control law (4.5) implies that the robot swings over the interval $[t_0 - \epsilon, t_0 + \epsilon]$, i.e. $u_i([t_0 - \epsilon, t_0 + \epsilon]) = 0$. We claim that this is impossible. By Proposition 2.1, the curvature of ∂P_j does not exceed $\frac{1}{r}$. The center of the robot follows a circle of radius b while swinging.

The curvature of this circle is $\frac{1}{b} > \frac{1}{r}$. Thus, the robot cannot follow the boundary of $\mathcal{C}_{free}$ while swinging. ■

Similar to their free counterparts in Section 3.2.3, we define tight and loose boundary trajectories as follows.

Definition 4.1. Let $q(t)$ be an optimal trajectory on the boundary, associated with adjoint $\lambda(t)$. We call $q(t)$ a *loose* optimal trajectory if $\lambda_1(0) = \lambda_2(0) = 0$ and $|\lambda_3(0)| = 2b$. We call $q(t)$ a *tight* optimal trajectory if $|\lambda_1(0)| + |\lambda_2(0)| \neq 0$.

Lemma 4.2. *Let $q(t)$ be a tight optimal trajectory defined on $[0,T]$ such that $q([0,T]) \subset \partial\mathcal{C}_{free}$. In that case, $\varphi_1(t) = \varphi_2(t)$ for $t \in [0,T]$.*

Proof. Lemma 4.1 showed that $|\varphi_1(t)| = |\varphi_2(t)| = 1$ for $t \in [0,T]$. Since $\varphi_i(t)$ are continuous over $[0,T]$, there are two possible cases: $\varphi_1 \equiv \varphi_2$ and $\varphi_1 \equiv -\varphi_2$. We show that $\varphi_1 \equiv \varphi_2$ in this case. On the contrary, suppose $\varphi_1 \equiv -\varphi_2$. The control law (4.5) implies that $u_1(t)u_2(t) \leq 0$. The robot rotates in place over the interval $[0,T]$, i.e. $u_1(t) = -u_2(t)$; otherwise, the center of the robot traverses a path whose curvature is more than $\frac{1}{r}$, which cannot be the boundary of obstacle by Proposition 2.1. Since $u_1(t) = -u_2(t)$ over the interval $[0,T]$,

$$\frac{\partial p}{\partial x}(q,u) = \frac{u_1+u_2}{2}\left(\frac{\partial^2 g}{\partial x^2}\cos\theta + \frac{\partial^2 g}{\partial x \partial y}\sin\theta\right) \equiv 0, \tag{4.6}$$

$$\frac{\partial p}{\partial y}(q,u) = \frac{u_1+u_2}{2}\left(\frac{\partial^2 g}{\partial y \partial x}\cos\theta + \frac{\partial^2 g}{\partial y^2}\sin\theta\right) \equiv 0. \tag{4.7}$$

Consequently, (4.2) implies that $\dot{\lambda}_1 \equiv \dot{\lambda}_2 \equiv 0$, in which case $\lambda_1 \equiv \lambda_1(0)$ and $\lambda_2 \equiv \lambda_2(0)$. Remember that we supposed $\varphi_1(t) = -\varphi_2(t)$, in which case (3.5) and (2.2) imply $\lambda_1 \cos\theta + \lambda_2 \sin\theta \equiv 0$. Since $q(t)$ is assumed to be tight, $|\lambda_1| + |\lambda_2| \neq 0$. This is true only if $\dot{\theta} \equiv 0$, which is a contradiction with robot rotating in place. ■

Lemma 4.3. *Let $q(t)$ be a loose optimal trajectory defined on $[0,T]$ such that $q([0,T]) \subset \partial\mathcal{C}_{free}$. In that case, $\varphi_1(t) = -\varphi_2(t)$ for $t \in [0,T]$, and the robot rotates in place, i.e. $u_1 \equiv -u_2$. Moreover, the trajectory remains loose over the time interval, i.e. $\lambda_1 \equiv \lambda_2 \equiv 0$ and $|\lambda_3| \equiv 2b$.*

Proof. Lemma 4.1 showed that $|\varphi_1(t)| = |\varphi_2(t)| = 1$ for $t \in [0, T]$. We show that $\varphi_1 \equiv -\varphi_2$ in this case. Since $\varphi_i(t)$ are continuous over $[0, T]$, it is enough to show that $\varphi_1(0) = -\varphi_2(0)$. Since $q(t)$ is loose, $\lambda_1(0) = \lambda_2(0) = 0$ and $|\lambda_3(0)| = 2b$. Equations (3.5) and (2.2) show that $\varphi_1(0) = -\frac{\lambda_3(0)}{2b} = -\varphi_2(0)$.

Now we show that the robot rotates in place over the time interval. The control law (4.5) shows that $u_1(t)u_2(t) \leq 0$ for $t \in [0, T]$. If there exist $\epsilon > 0$ and $t_0 \in [0, T - \epsilon]$ such that $u_1(t) \neq -u_2(t)$ for $t \in [t_0, t_0 + \epsilon]$, then the center of the robot traverses a path whose curvature is more than $\frac{1}{r}$, which cannot be the boundary of obstacle by Proposition 2.1. Therefore, $u_1 \equiv -u_2$ and the robot rotates in place.

Since $u_1(t) = -u_2(t)$ over $[0, T]$,

$$\frac{\partial p}{\partial x}(q, u) = \frac{u_1 + u_2}{2}\left(\frac{\partial^2 g}{\partial x^2}\cos\theta + \frac{\partial^2 g}{\partial x \partial y}\sin\theta\right) \equiv 0, \tag{4.8}$$

$$\frac{\partial p}{\partial y}(q, u) = \frac{u_1 + u_2}{2}\left(\frac{\partial^2 g}{\partial y \partial x}\cos\theta + \frac{\partial^2 g}{\partial y^2}\sin\theta\right) \equiv 0. \tag{4.9}$$

Consequently, (4.2) implies that $\dot{\lambda}_1 \equiv \dot{\lambda}_2 \equiv 0$ and $\lambda_1 \equiv 0$, $\lambda_2 \equiv 0$. Finally, $|\varphi_i| \equiv 1$ implies $|\lambda_3| \equiv 2b$. ■

4.3 Intersection Points

Segments of a minimum wheel-rotation trajectory among obstacles are either in the interior of the free region or on the obstacle boundary. In Section 4.2 and Chapter 3, we studied both segment types. We here characterize the intersections that can happen between two consecutive segments. First, we give the following definition to distinguish between two different types of intersection. We focus on a single isolated intersection in the definition.

Definition 4.2. Let $q(t)$ be an optimal trajectory defined on $[0, T]$. If $0 < \tau < T$ is such that $q([0, \tau)) \subset \mathcal{C}_{free} \backslash \partial\mathcal{C}_{free}$ and $q([\tau, T]) \subset \partial\mathcal{C}_{free}$, then we call τ the *junction time* of $q(t)$, and $q(\tau)$ the *junction point*. Likewise, we call τ the *junction time* of $q(t)$, and $q(\tau)$ the *junction point* if $q([0, \tau]) \subset \partial\mathcal{C}_{free}$ and $q((\tau, T]) \subset \mathcal{C}_{free} \backslash \partial\mathcal{C}_{free}$. If $q(\tau) \in \partial\mathcal{C}_{free}$ and $q([0, T] \backslash \{\tau\}) \subset \mathcal{C}_{free} \backslash \partial\mathcal{C}_{free}$, then we call τ the *reflection time* of $q(t)$, and $q(\tau)$ the

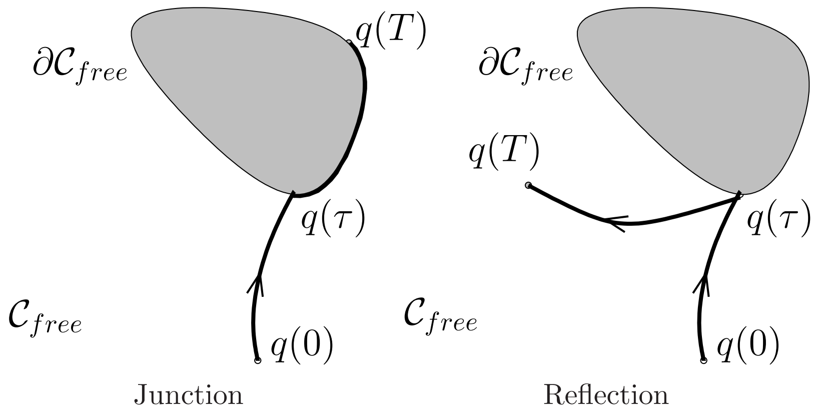


Figure 4.5: The junction/reflection time τ and the junction/reflection point $q(\tau)$ on a trajectory $q(t)$.

reflection point. See Figure 4.5 for an illustration. Note that $q(t)$ is not necessarily an optimal trajectory in Figure 4.5.

The Pontryagin Jump Condition [141] and the symmetry of our problem yield the following lemma.

Lemma 4.4 (Pontryagin Jump Condition [141])**.** *Let $q(t)$ be an optimal trajectory defined on $[0, T]$, associated with the adjoint $\lambda(t)$. Assume $q(\tau) \in \partial\mathcal{C}_{free}$. At $t = \tau$, denote the left and the right limits of $\lambda(t)$ by $\lambda(\tau^-)$ and $\lambda(\tau^+)$ respectively. In that case, $\lambda(\tau^+) = \lambda(\tau^-)$ if τ is a junction time. If τ is a reflection time, then $\lambda(\tau^+) = \lambda(\tau^-) + \mu(\partial g/\partial q)$, in which μ is a constant.*

Corollary 4.5. *If τ is a junction time of $q(t)$, then $\lambda(t)$ is continuous on the whole interval $[0, T]$.*

Corollary 4.5 implies that a loose optimal trajectory cannot intersect a tight optimal trajectory at a junction point. In the following section, we characterize possible junction points.

4.3.1 Characterization of Junction Points

Without loss of generality, we assume throughout this section that $q(t)$ is an optimal trajectory defined on $[0, T]$ such that $q([0, \tau)) \subset \mathcal{C}_{free} \backslash \partial\mathcal{C}_{free}$ and $q([\tau, T]) \subset \partial\mathcal{C}_{free}$. Corollary 4.5 shows that $\lambda(t)$ is continuous over $[0, T]$. Therefore, the switching functions $\varphi_1(t)$ and

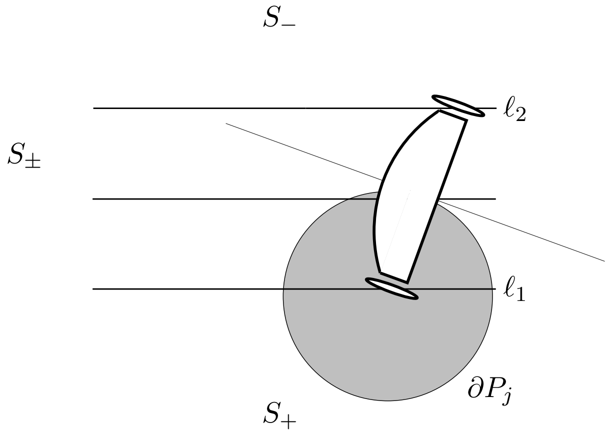


Figure 4.6: Orientation vector of the robot is tangent to the obstacle, and the robot center lies on the center line of $S_\pm$ region at a junction point along a tight minimum wheel-rotation trajectory. Note that the depicted obstacle is in the configuration space.

$\varphi_2(t)$ are also continuous. Junction points over loose optimal trajectories are not important, because a loose optimal trajectory remains loose on the whole time interval; see Lemma 4.3 and Corollary 4.5. In the following, we characterize junction points over tight optimal trajectories.

Theorem 4.6. *Let $q(t)$ be a tight optimal trajectory. Assume $q(\tau) = (x(\tau), y(\tau), \theta(\tau))$ is the junction point of $q(t)$. If $(x(\tau), y(\tau)) \in \partial P_j$, then the vector $(\cos\theta(\tau), \sin\theta(\tau))$ is tangent to ∂P_j at $(x(\tau), y(\tau))$. Moreover, $\lambda_3(\tau) = 0$.*

Proof. Obviously, $\dot{q}(\tau) = (\dot{x}(\tau), \dot{y}(\tau), \dot{\theta}(\tau))$ is tangent to $\partial P_j \times \mathbb{S}^1$ at $q(\tau)$. By (2.1) and (2.2),

$$(\dot{x}(\tau), \dot{y}(\tau)) = \frac{u_1(\tau) + u_2(\tau)}{2}(\cos\theta(\tau), \sin\theta(\tau)). \tag{4.10}$$

To obtain the result, it is enough to show that $u_1(\tau) + u_2(\tau) \neq 0$. By Lemma 4.2, $\varphi_1(t) = \varphi_2(t)$ over the interval $[\tau, T]$. Therefore, the control law (4.5) implies $u_1(\tau)u_2(\tau) \geq 0$. If $u_1(\tau) + u_2(\tau) = 0$, then $u_1(\tau) = u_2(\tau) = 0$ in which case the trajectory is motionless. Finally, $\varphi_1(\tau) = \varphi_2(\tau)$ with (2.2) and (3.5) implies $\lambda_3(\tau) = 0$. ■

According to Theorem 4.6, the robot orientation vector is tangent to the obstacle at a junction point over a tight optimal trajectory. In addition, the center of the robot lies on the center line of $S_\pm$ region, in which $S_\pm$ is defined in Section 3.2.4. See Figure 4.6 for an illustration. In the following section, we characterize possible reflection points.

4.3.2 Characterization of Reflection Points

Taking into account $\partial g/\partial\theta = 0$, Lemma 4.4 (Pontryagin Jump Condition) gives

$$\lambda_1(\tau^+) = \lambda_1(\tau^-) + \mu\frac{\partial g}{\partial x}, \tag{4.11}$$

$$\lambda_2(\tau^+) = \lambda_2(\tau^-) + \mu\frac{\partial g}{\partial y}, \tag{4.12}$$

$$\lambda_3(\tau^+) = \lambda_3(\tau^-). \tag{4.13}$$

Those equations with (3.5) and (2.2) imply that

$$\varphi_1(\tau^+) = \varphi_1(\tau^-) + \mu(\frac{\partial g}{\partial x}\cos\theta + \frac{\partial g}{\partial y}\sin\theta), \tag{4.14}$$

$$\varphi_2(\tau^+) = \varphi_2(\tau^-) + \mu(\frac{\partial g}{\partial x}\cos\theta + \frac{\partial g}{\partial y}\sin\theta). \tag{4.15}$$

Theorem 4.7. *Let $q(t)$ be an optimal trajectory and let τ be the reflection time of $q(t)$. Remember $q(\tau) = (x(\tau), y(\tau), \theta(\tau))$, and let $(x(\tau), y(\tau)) \in \partial P_j$. In that case, the following hold:*

1. *(Loose-Loose) If both $q|_{[0,\tau)}$ and $q|_{[\tau,T]}$ are loose optimal trajectories, then $q(t)$ is a loose extremal in Section 3.3.3.*

2. *(Tight-Loose) If $q|_{[0,\tau)}$ is a tight and $q|_{[\tau,T]}$ is a loose optimal trajectory, then $(\cos\theta(\tau), \sin\theta(\tau))$ is tangent to ∂P_j at $(x(\tau), y(\tau))$.*

3. *(Tight-Tight) If both $q|_{[0,\tau)}$ and $q|_{[\tau,T]}$ are tight optimal trajectories, then either $(\cos\theta(\tau), \sin\theta(\tau))$ is tangent to ∂P_j at $(x(\tau), y(\tau))$, or $\varphi_1(\tau^+) = -\varphi_2(\tau^-)$ and $\varphi_2(\tau^+) = -\varphi_1(\tau^-)$.*

Proof.

(Loose-Loose) If both $q|_{[0,\tau)}$ and $q|_{[\tau,T]}$ are loose optimal trajectories, in which case $\lambda_1(\tau^+) = \lambda_1(\tau^-) = 0$ and $\lambda_2(\tau^+) = \lambda_2(\tau^-) = 0$, then $\lambda(\tau^+) = \lambda(\tau^-)$. Therefore, $q(t)$ is a loose extremal in Section 3.3.3.

(Tight-Loose) Since $q|_{[\tau,T]}$ is loose, either $\varphi_1(\tau^+) = -\varphi_2(\tau^+) = 1$ or $\varphi_1(\tau^+) = -\varphi_2(\tau^+) = -1$. Since $q|_{[0,\tau)}$ is tight, either $|\varphi_1(\tau^-)| = 1$ or $|\varphi_2(\tau^-)| = 1$. In any case,

$$\mu(\frac{\partial g}{\partial x}\cos\theta + \frac{\partial g}{\partial y}\sin\theta) = 0 \tag{4.16}$$

because

$$\varphi_i(\tau^+) = \varphi_i(\tau^-) + \mu(\frac{\partial g}{\partial x}\cos\theta + \frac{\partial g}{\partial y}\sin\theta), \tag{4.17}$$

and $|\varphi_i| \leq 1$. Therefore, $(\cos\theta(\tau), \sin\theta(\tau))$ is tangent to ∂P_j at $(x(\tau), y(\tau))$.

(Tight-Tight) Suppose $(\cos\theta(\tau), \sin\theta(\tau))$ is not tangent to ∂P_j at $(x(\tau), y(\tau))$. In that case,

$$\mu(\frac{\partial g}{\partial x}\cos\theta + \frac{\partial g}{\partial y}\sin\theta) \neq 0. \tag{4.18}$$

Since $q|_{[0,\tau)}$ is tight, either $|\varphi_1(\tau^-)| = 1$ or $|\varphi_2(\tau^-)| = 1$. Since $q|_{[\tau,T]}$ is tight, either $|\varphi_1(\tau^+)| = 1$ or $|\varphi_2(\tau^+)| = 1$. If $\varphi_1(\tau^-) = \varphi_2(\tau^-) = 1$, then (4.17) and (4.18) imply $\varphi_1(\tau^+) = \varphi_2(\tau^+) = -1$. In that case, the robot moves in opposite directions before and after the reflection time, which is absurd. A similar argument rules out $\varphi_1(\tau^-) = \varphi_2(\tau^-) = -1$. Since (4.18) holds, $\varphi_1(\tau^-) = -\varphi_2(\tau^-) = 1$ and $\varphi_1(\tau^-) = -\varphi_2(\tau^-) = -1$ are impossible. Since $|\varphi_i| \leq 1$, the only remaining possibility is $\varphi_1(\tau^+) = -\varphi_2(\tau^-)$ and $\varphi_2(\tau^+) = -\varphi_1(\tau^-)$, which is described below.

1. If $\varphi_1(\tau^-) = -1$ and $|\varphi_2(\tau^-)| < 1$, then $\varphi_2(\tau^+) = 1$, $\varphi_1(\tau^+) = -\varphi_2(\tau^-)$, and $\mu(\frac{\partial g}{\partial x}\cos\theta + \frac{\partial g}{\partial y}\sin\theta) = 1 - \varphi_2(\tau^-)$.

2. If $\varphi_2(\tau^-) = -1$ and $|\varphi_1(\tau^-)| < 1$, then $\varphi_1(\tau^+) = 1$, $\varphi_2(\tau^+) = -\varphi_1(\tau^-)$, and $\mu(\frac{\partial g}{\partial x}\cos\theta + \frac{\partial g}{\partial y}\sin\theta) = 1 - \varphi_1(\tau^-)$.

3. If $\varphi_1(\tau^-) = 1$ and $|\varphi_2(\tau^-)| < 1$, then $\varphi_2(\tau^+) = -1$, $\varphi_1(\tau^+) = -\varphi_2(\tau^-)$, and $\mu(\frac{\partial g}{\partial x}\cos\theta + \frac{\partial g}{\partial y}\sin\theta) = -1 - \varphi_2(\tau^-)$.

4. If $\varphi_2(\tau^-) = 1$ and $|\varphi_1(\tau^-)| < 1$, then $\varphi_1(\tau^+) = -1$, $\varphi_2(\tau^+) = -\varphi_1(\tau^-)$, and $\mu(\frac{\partial g}{\partial x}\cos\theta + \frac{\partial g}{\partial y}\sin\theta) = -1 - \varphi_1(\tau^-)$.

■

Due to symmetries, (Loose-Tight) case is similar to the (Tight-Loose) case in Theorem 4.7. In the (Tight-Tight) case of Theorem 4.7, μ is uniquely computed from the equations in the proof. Hence, $\lambda(\tau^+)$ is uniquely determined by $\lambda(\tau^-)$ at the reflection point.

We are now ready to present the nonholonomic bitangency graph, because we have characterized free and boundary minimum wheel-rotation segments together with their possible intersections. In Section 4.3, we showed that the orientation vector of the robot is tangent to the boundary of obstacle at a junction point over a tight optimal trajectory, and the center of the robot lies on the center line of $S_\pm$ region; see Figure 4.6 for an illustration. We also showed that either the orientation vector of the robot is tangent to the boundary of obstacle at a reflection point or the value of the adjoint after the reflection time is uniquely determined by its value before the reflection time. We use those results to define our nonholonomic bitangency graph.

4.4 Nonholonomic Bitangency Graph

In this section, we define a nonholonomic bitangency graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ among obstacles $P_1, P_2, \ldots, P_n$. The graph is used to answer multiple geodesic queries. Since $\mathbf{G}$ may be infinite in general, it is not always possible to build it explicitly. Therefore for each query, the solution is extracted by a search algorithm in parts of a graph that is obtained by augmenting $\mathbf{G}$ with initial and goal configurations and their connecting edges. Vertices of $\mathbf{G}$ are the junction and reflection points of minimum wheel-rotation trajectories for every pair of initial and goal configurations. More precisely, $v = (x, y, \theta) \in \mathbf{V} \subset \partial\mathcal{C}_{free}$ if and only if $(\cos\theta, \sin\theta)$ is tangent to the boundary of obstacle at (x, y). Thus, there may exist two distinct vertices in $\mathbf{V}$ with the same (x, y) coordinates but with opposite orientations. In a nutshell, there is an edge $(u, v) \in \mathbf{E}$ between $u, v \in \mathbf{V}$ if there is a free extremal, as characterized in Chapter 3, or an obstacle minimum wheel-rotation trajectory, as characterized in Section 4.2, between u and v. We now precisely define edges of $\mathbf{G}$.

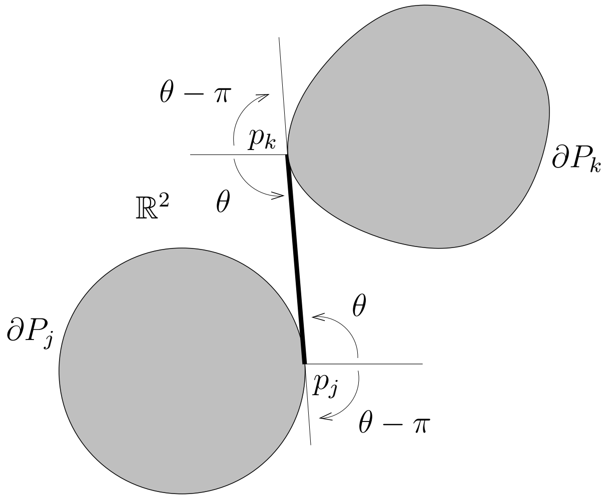


Figure 4.7: Bitangent edges (v_j^1, v_k^1) and (v_j^2, v_k^2) in the nonholonomic bitangency graph $\mathbf{G}$, in which $v_j^1 = (p_j, \theta), v_j^2 = (p_j, \theta - \pi) \in \mathbb{R}^2 \times \mathbb{S}^1$ and $v_k^1 = (p_k, \theta), v_k^2 = (p_k, \theta - \pi) \in \mathbb{R}^2 \times \mathbb{S}^1$.

4.4.1 Tight Edges of G

There is a tight edge between $u, v \in \mathbf{V}$ if there is a tight minimum wheel-rotation trajectory in $\mathcal{C}_{free}$ between u and v. In the following we first present those tight edges whose both end points are junction points.

Let $v_j^1 = (p_j, \theta), v_j^2 = (p_j, \theta - \pi), v_k^1 = (p_k, \theta), v_k^2 = (p_k, \theta - \pi) \in \mathbf{V}$ where $p_j \in \partial P_j$ and $p_k \in \partial P_k$. The superscripts of v^1, v^2 represent two different orientations which are an angle π apart. In that case, $(v_j^1, v_k^1), (v_j^2, v_k^2) \in \mathbf{E}$ if the line segment $p_j - p_k$ is tangent to ∂P_j at p_j and tangent to ∂P_k at p_k and it is completely in $\mathcal{C}_{free}$; see Figure 4.7. Associate the length of $p_j - p_k$ segment to the edges (v_j^1, v_k^1) and (v_j^2, v_k^2). See Figures 4.8 and 4.9 for an example.

Add the edge (v_j^1, v_k^2) (and (v_j^2, v_k^1)) to $\mathbf{E}$ if there is a trajectory $\mathbf{S}_{d_1}\mathbf{L}_{\frac{\pi}{2}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{S}_{d_2}$ or $\mathbf{S}_{d_1}\mathbf{R}_{\frac{\pi}{2}}\mathbf{L}_{\frac{\pi}{2}}\mathbf{S}_{d_2}$ in $\mathcal{C}_{free}$, which starts at v_j^1 (respectively v_j^2) and ends at v_k^2 (respectively v_k^1); see Figures 4.10 and 4.11. Note that the swing parts of such trajectories are in the same direction, i.e. both clockwise or both counter-clockwise. Associate wheel rotation of the trajectory which is $d_1 + d_2 + \pi$ to the edges (v_j^1, v_k^2) and (v_j^2, v_k^1). This construction corresponds to those tight minimum wheel-rotation pieces for which the width of $S_\pm$ region is $2b$.

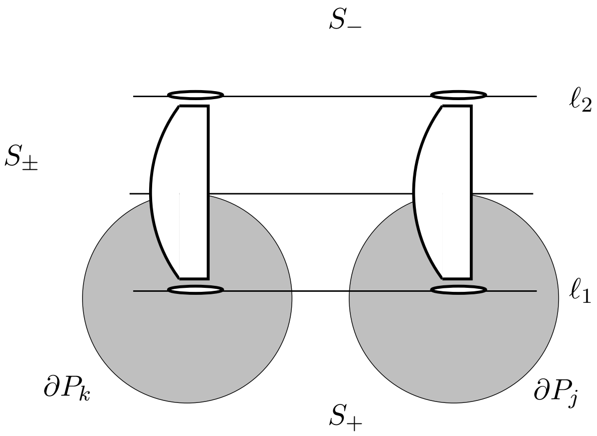


Figure 4.8: Robot motion along a bitangent edge of **G**. The width of $S_\pm$ is $2b$ in this case.

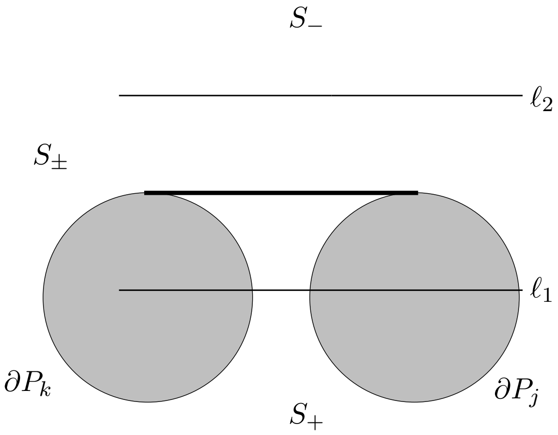


Figure 4.9: The path traversed by the robot center along an edge in Figure 4.8.

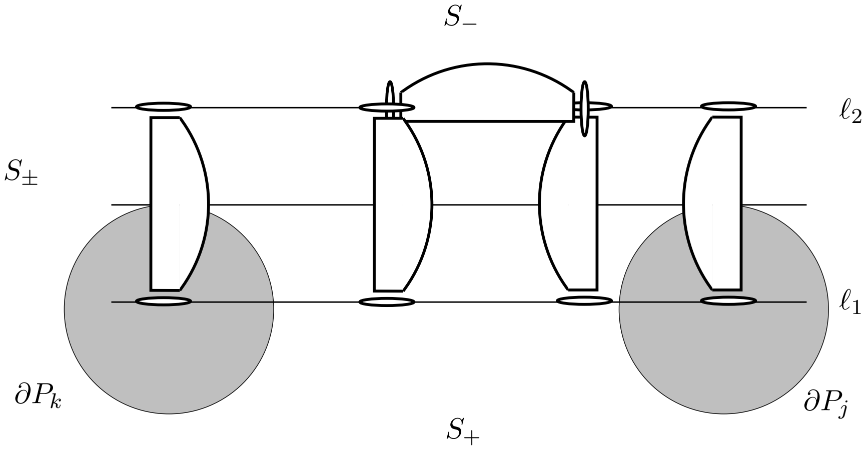


Figure 4.10: A bitangent edge of **G** that contains a 180° flip. The width of $S_\pm$ is $2b$ in this case.

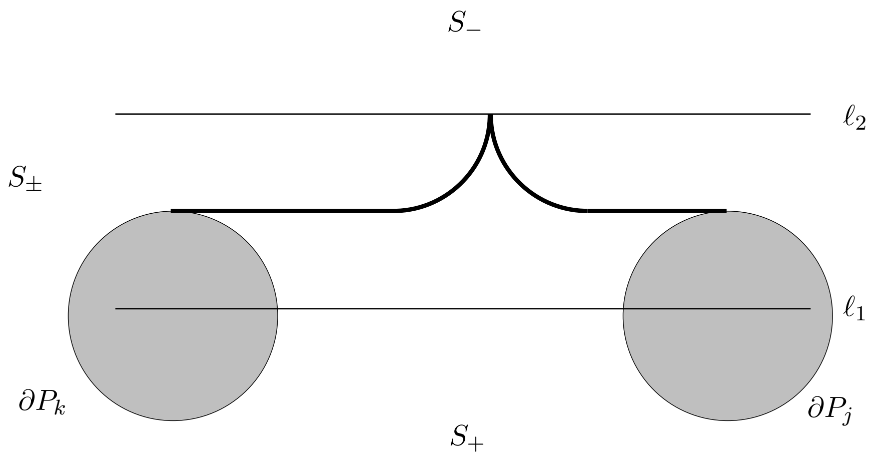


Figure 4.11: The path traversed by the robot center along the edge in Figure 4.10.

Let $v_j = (p_j, \theta_j), v_k = (p_k, \theta_k) \in \mathbf{V}$, in which $p_j \in \partial P_j$ and $p_k \in \partial P_k$. In that case, $(v_j, v_k) \in \mathbf{E}$ if the angle α between the segment $p_j - p_k$ and the tangent on ∂P_j at p_j is equal to the angle between the segment $p_j - p_k$ and the tangent on ∂P_k at p_k, and a path of the following forms: $\mathbf{R}_{\frac{\pi}{2}-\alpha}\mathbf{L}_{\frac{\pi}{2}-\alpha}\mathbf{R}_{\frac{\pi}{2}-\alpha}\mathbf{L}_{\frac{\pi}{2}-\alpha}(\mathbf{R}_{\frac{\pi}{2}-\alpha}\mathbf{L}_{\frac{\pi}{2}-\alpha}\mathbf{R}_{\frac{\pi}{2}-\alpha}\mathbf{L}_{\frac{\pi}{2}-\alpha})^*$, $\mathbf{L}_{\frac{\pi}{2}-\alpha}\mathbf{R}_{\frac{\pi}{2}-\alpha}\mathbf{L}_{\frac{\pi}{2}-\alpha}\mathbf{R}_{\frac{\pi}{2}-\alpha}(\mathbf{L}_{\frac{\pi}{2}-\alpha}\mathbf{R}_{\frac{\pi}{2}-\alpha}\mathbf{L}_{\frac{\pi}{2}-\alpha}\mathbf{R}_{\frac{\pi}{2}-\alpha})^*$, $\mathbf{R}_{\frac{\pi}{2}-\alpha}\mathbf{L}_{\frac{\pi}{2}-\alpha}(\mathbf{R}_{\frac{\pi}{2}-\alpha}\mathbf{L}_{\frac{\pi}{2}-\alpha}\mathbf{R}_{\frac{\pi}{2}-\alpha}\mathbf{L}_{\frac{\pi}{2}-\alpha})^*$, or $\mathbf{L}_{\frac{\pi}{2}-\alpha}\mathbf{R}_{\frac{\pi}{2}-\alpha}(\mathbf{L}_{\frac{\pi}{2}-\alpha}\mathbf{R}_{\frac{\pi}{2}-\alpha}\mathbf{L}_{\frac{\pi}{2}-\alpha}\mathbf{R}_{\frac{\pi}{2}-\alpha})^*$ which takes the robot from v_j to v_k is in $\mathcal{C}_{free}$. Superscript $*$ means zero or more copies of the base expression. In the base expression the first two swings are in the same direction and the remaining two are also in the same direction, i.e. both clockwise or both counter-clockwise. This construction corresponds to those tight minimum wheel-rotation pieces for which the width of $S_\pm$ region is less than $2b$. See Figures 4.12 and 4.13 for an illustration. To the edge (v_j, v_k), associate wheel rotation of such path, which is $N(\pi - 2\alpha)$ where N is the number of swing pairs.

We considered only those tight edges, so far, whose end points are junction points. We here consider those tight edges whose end points are reflection points. Add (v_j, v_k) to $\mathbf{E}$, where $v_j, v_k \in \mathbf{V}$, if there exists a tight extremal, as in Section 3.3.2, that takes the robot from v_j to v_k. See Figures 4.14 and 4.15 for a sample edge whose initial point is a reflection point. In addition, add (v_j, v_k) to $\mathbf{E}$ if there exists a tight extremal, with a number of reflection points in the middle, that takes the robot from v_j to v_k. See Figures 4.16 and 4.17 for a sample tight edge that contains a reflection point in the middle. The reflection

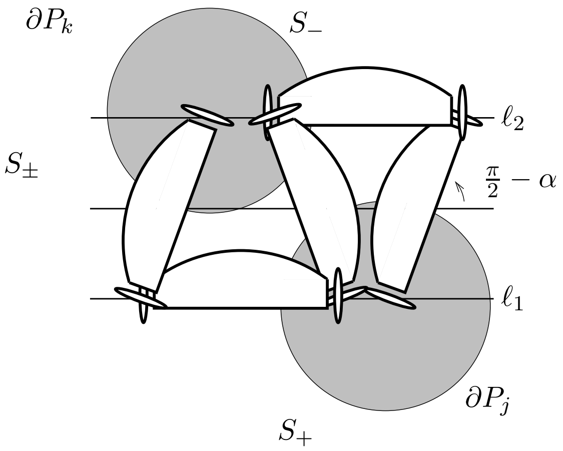


Figure 4.12: Illustration of $\mathbf{R}^-_{\frac{\pi}{2}-\alpha}\mathbf{L}^-_{\frac{\pi}{2}-\alpha}\mathbf{R}^+_{\frac{\pi}{2}-\alpha}\mathbf{L}^+_{\frac{\pi}{2}-\alpha}$ as an edge of $\mathbf{G}$. The width of $S_\pm$ is less than $2b$ in this case.

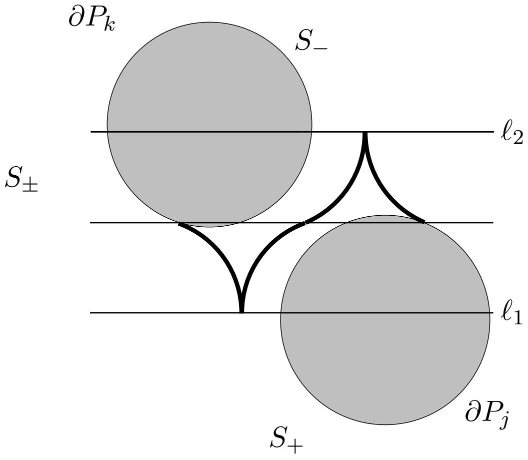


Figure 4.13: The path traversed by the robot center along the edge in Figure 4.12.

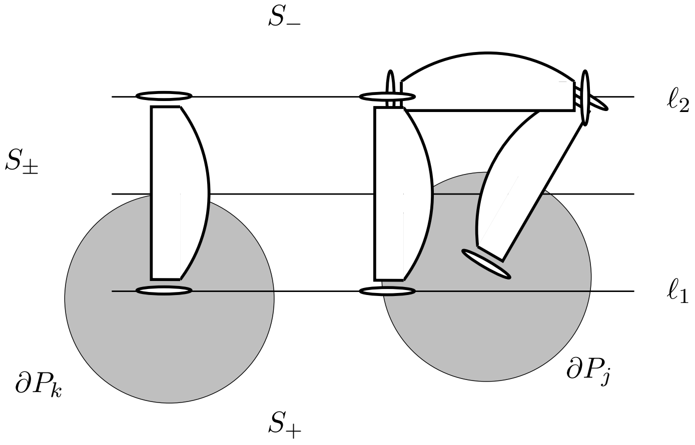


Figure 4.14: A sample tight edge whose initial point is a reflection point.

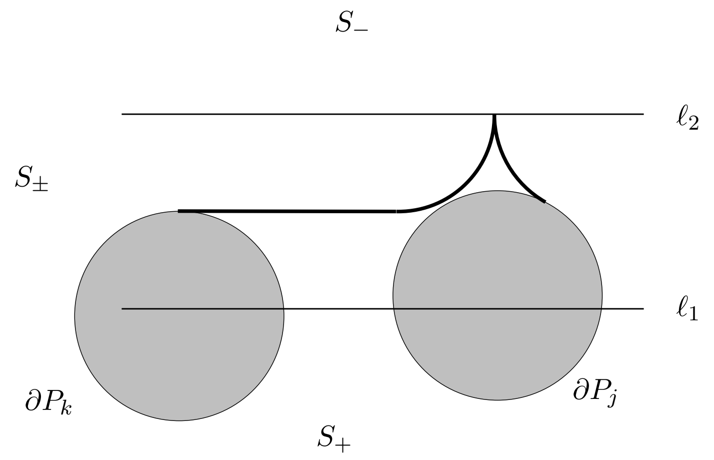


Figure 4.15: The path traversed by the robot center along the edge in Figure 4.14.

points are characterized in the (Tight-Tight) case of Theorem 4.7. In particular, the robot orientation is allowed to be non-tangent to the obstacle boundary at the reflection points. We regard those reflection points as transient points which do not play the role of a vertex in the visibility graph. At those points, the value of the adjoint after the reflection time is uniquely determined by its value before the reflection time; see proof of Theorem 4.7. In this way, the trajectory between v_j and v_k can be uniquely computed.

4.4.2 Loose Edges of G

Lemma 4.3 and Corollary 4.5 show that a loose optimal trajectory remains loose if it does not contain reflection points. Theorem 4.7 shows that an optimal trajectory can switch from loose to tight or from tight to loose at a reflection point if the robot orientation is tangent to the obstacle boundary, and either $\varphi_1 = -\varphi_2 = 1$ or $\varphi_1 = -\varphi_2 = -1$. Let $v_j, v_k \in \mathbf{V}$. If v_j and v_k are both reflection points, adjacent to tight edges, and either $\varphi_1 = -\varphi_2 = 1$ at v_j and v_k, or $\varphi_1 = -\varphi_2 = -1$ at v_j and v_k, then it is possible to add (v_j, v_k) to $\mathbf{E}$. The trajectory connecting v_j to v_k must be a loose extremal of the form given in Section 3.3.3.

4.4.3 Boundary Edges of G

For every pair $v_1 = (p_1, \theta_1), v_2 = (p_2, \theta_2) \in \mathbf{V}$ such that p_1 and p_2 belong to the same obstacle boundary component ∂P_ℓ, add an edge $(v_1, v_2) \in \mathbf{E}$ if the robot can move from v_1

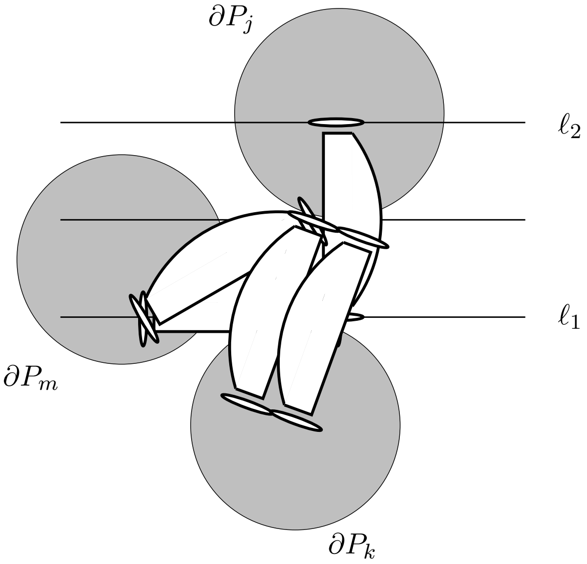


Figure 4.16: A sample tight edge that contains a reflection point in ∂P_m.

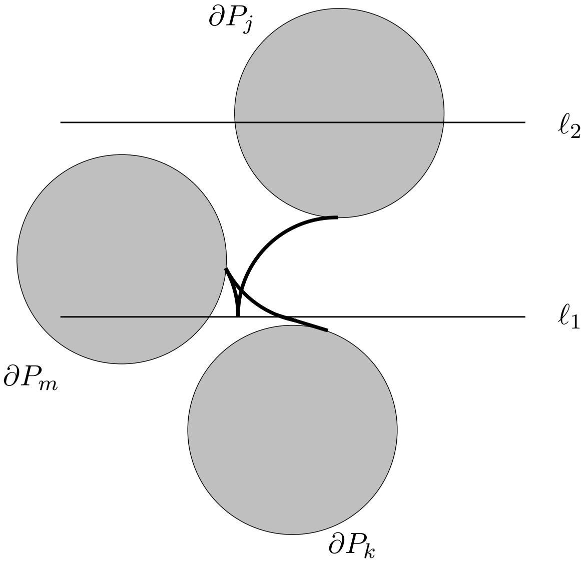


Figure 4.17: The path traversed by the robot center along the edge in Figure 4.16.

to v_2 by following the boundary of obstacle ∂P_ℓ. Associate the length of the path in ∂P_ℓ to this edge.

4.5 Shortest Path Query

In a query, v_{init}, the initial configuration, and v_{goal}, the goal configuration, are given. To compute the solution, we augment the nonholonomic bitangency graph $\mathbf{G}$ with v_{init} and v_{goal} and their connecting edges. To connect v_{init} and v_{goal} to $\mathbf{G}$, it is enough to regard them as elements of $\mathbf{V}$ and consider all applicable tight and loose edges, in Sections 4.4.1 and 4.4.2, that connect them to the graph. Since $\mathbf{G}$ may be infinite in general, it is not always possible to build it explicitly. Therefore for each query, the solution is extracted by a search algorithm in parts or all of the augmented graph.

4.6 Summary

In this chapter, we presented a method to compute minimum wheel-rotation trajectories among convex obstacles. Among obstacles, a minimum wheel-rotation trajectory is composed of a finite number of segments, each of which is either in the interior of the free configuration space or on the boundary of obstacle region. Those segments that are in the interior of the free region were characterized in Chapter 3. A constrained Pontryagin Maximum Principle with our problem-specific arguments helped us to characterize boundary segments in Section 4.2. We used the Pontryagin Jump Condition [141] to rule out non-optimal intersections between free and boundary segments. Those results helped us to introduce a nonholonomic bitangency graph to which the search for the minimum wheel-rotation path is confined. In general, our nonholonomic bitangency graph is a 2-dimensional subset of the 3-dimensional configuration space of the robot. Therefore, further optimization or a continuous search may be required to answer queries. In cases where the graph is 1-dimensional and can be computed, any graph search algorithm, such as Dijkstra's algorithm, is employed to extract the solution.

Chapter 5

Time-Optimal Paths for a Dubins Airplane

We give a characterization of the time-optimal trajectories for our simplified airplane model in this chapter. Our airplane model is a natural extention of the Dubins car [56], and extends it with an additional configuration variable for the altitude. In Section 2.1.2, we precisely defined the model. The time-optimal trajectories specify time-minimizing maneuvers. In addition, they comprise a useful set of motion primitives as it was discussed in Section 1.1, and they play a crucial role in air traffic management systems [95, 168, 174, 182], e.g. in detecting the safety regions. This chapter was a joint work with Steven M. LaValle [43].

5.1 Related Work

Walsh, Montgomery, and Sastry used Pontryagin Maximum Principle to plan optimal paths on matrix Lie groups [174]. Specifically, they plan optimal paths for an airplane in $SE(2)$, $SO(3)$, and $SE(3)$. Their cost function is quadratic in the input. In this chapter, we consider a different problem in which we minimize time for a system in $SE(2) \times \mathbb{R}$. For algorithms for steering on matrix Lie groups, see [150, 173], and for optimal path planning for Unmanned Aerial Vehicles (UAV) with tactical constraints, see [176, 178].

The approach that we use to derive optimal trajectories is similar in spirit to the one used in Chapter 3. In Section 2.2.2, we proved the existence of optimal paths. We use the Pontryagin Maximum Principle as a necessary condition to rule out some non-optimal paths. We distinguish three cases: low, medium, and high goal altitudes of the airplane. Intuitively, if the goal altitude is low, the airplane has to follow the shortest path for the Dubins car with an unsaturated altitude velocity. If the goal altitude is high, the altitude velocity gets saturated and the system has to maneuver until it reaches the goal altitude.

For medium altitudes in between low and high, the time-optimal path is either a locally longest curve for the Dubins car or a path composed of turns and pieces of planar elastica [85] with saturated altitude velocity. Locally longest curves for the Dubins car, which cannot be infinitesimally elongated, play an important role in the airplane time-optimal trajectories for medium altitude. An example of such locally longest curves is a short arc of a circle. Dubins proved that a short arc of a circle is isolated in the space of all bounded curvature plane paths [57]. As a by-product, we characterize locally longest curves in this chapter.

5.2 Pontryagin Maximum Principle

Let the Hamiltonian $H : \mathbb{R}^4 \times \mathcal{C}' \times U \to \mathbb{R}$ be

$$H(\lambda, p, u) = \langle \lambda, \dot{p} \rangle \tag{5.1}$$

in which $\dot{p}$ is given in (2.8). According to the Pontryagin Maximum Principle [141], for every optimal trajectory $p(t)$ defined on $[0, T]$ and associated with control $u(t)$, there exists a constant $\lambda_0 \geq 0$ and an absolutely continuous vector-valued adjoint function $\lambda(t) = (\lambda_1(t), \lambda_2(t), \lambda_3(t), \lambda_4(t))$, that is nonzero if $\lambda_0 = 0$, with the following properties along the optimal trajectory:

$$\begin{aligned} \dot{\lambda} &= -\frac{\partial H}{\partial p}, & (5.2)\\ H(\lambda(t), p(t), u(t)) &= \max_{z \in U} H(\lambda(t), p(t), z), & (5.3)\\ H(\lambda(t), p(t), u(t)) &\equiv \lambda_0. & (5.4) \end{aligned}$$

Definition 5.1. An *extremal* is a trajectory $p(t)$ that satisfies the conditions of the Pontryagin Maximum Principle.

In this section, let $p(t)$ be an extremal associated with the adjoint $\lambda(t)$ and the control

$u(t)$. Equation (5.2) can be solved for λ to obtain

$$\lambda(t) = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_1 y - c_2 x + c_4 \end{pmatrix}, \tag{5.5}$$

in which c_1, c_2, c_3, and c_4 are constants. Along an extremal, (5.3) yields the extremal control law

$$u_z = \operatorname{sgn}(c_3) \text{ if } c_3 \neq 0 \tag{5.6}$$

$$u_z \in [-1, 1] \text{ if } c_3 = 0 \tag{5.7}$$

$$u_\theta = \operatorname{sgn}(c_1 y - c_2 x + c_4) \text{ if } c_1 y - c_2 x + c_4 \neq 0 \tag{5.8}$$

$$u_\theta \in [-1, 1] \text{ if } c_1 y - c_2 x + c_4 = 0. \tag{5.9}$$

If $c_3 = 0$, then (5.6) implies that u_z can have any value within $[-1, 1]$. In this case, the following two propositions show that the projection of $p(t)$ onto the (x, y, θ)-space is an extremal for the Dubins car.

Proposition 5.1. *If $c_3 = 0$ and $\lambda_0 \neq 0$, then the projection of $p(t)$ onto the (x, y, θ)-space is an extremal for the Dubins car, i.e a trajectory of the Dubins car that satisfies the Pontryagin Maximum Principle.*

Proof. Since $c_3 = 0$ and $H = \lambda_0 \neq 0$, the vector $(\lambda_1, \lambda_2, \lambda_4)$ is nonzero. Hence, the projection of $p(t)$ onto (x, y, θ)-space satisfies the Pontryagin Maximum Principle. Thus, it has to be an extremal for the Dubins car. ■

Proposition 5.2. *If both $c_3 = 0$ and $\lambda_0 = 0$, then $p(t)$ has zero duration.*

Proof. In this case, conditions (5.4) and (5.3) imply that $\lambda_1 \cos\theta + \lambda_2 \sin\theta + |\lambda_4| \equiv 0$. Thus, the projection of $p(t)$ onto the (x, y, θ)-space is an abnormal extremal for the Dubins car.

Abnormal extremals for the Dubins car have zero duration. ■

If $c_3 \neq 0$, then the duration of $p(t)$ is $|z_g|$ in which z_g is the final altitude, because $u_z \equiv 1$ or -1, depending on the sign of c_3, by (5.6). It is possible to have $c_1 = c_2 = c_4 = 0$ because $c_3 \neq 0$. In that case, u_θ can have any value within $[-1, 1]$, by (5.6). This means that the projection of $p(t)$ onto the (x, y, θ)-space can be any feasible path for the Dubins car. However, the length of such path must be $|z_g|$. When does there exist a path of given length for the Dubins car? We will study this question in the following section.

5.3 Paths With Given Length for the Dubins Car

We desire to plan a path for the Dubins car with prescribed length. Lemma 5.3 in [57] proves that a short arc of circle (of radius 1) is isolated in the space of all admissible paths for the Dubins car. Intuitively, there are no feasible trajectories for the Dubins car between the end points of the arc with a length slightly more than the length of the arc. Whenever there exists a desired path, we pick the one which minimizes a quadratic cost.

Equations of motion for the Dubins car are

$$\dot{x} = \cos\theta, \tag{5.10}$$

$$\dot{y} = \sin\theta, \tag{5.11}$$

$$\dot{\theta} = u. \tag{5.12}$$

Following [174], we pick the path that minimizes $\int_0^T u^2 dt$ with given length for this system. If there exists such path, it should satisfy the Pontryagin Maximum Principle. Let the Hamiltonian $F : \mathbb{R}^3 \times (\mathbb{R}^2 \times \mathbb{S}^1) \times [-1, 1] \rightarrow \mathbb{R}$ be

$$F(\gamma, (x, y, \theta), u) = \langle \gamma, (\cos\theta, \sin\theta, u) \rangle + \gamma_0 u^2 \tag{5.13}$$

in which γ_0 is a constant. For every desired path $p(t) = (x(t), y(t), \theta(t))$ defined on $[0, T]$ and associated with control $u(t)$, there exists a constant $\gamma_0 \leq 0$ and an absolutely continuous vector-valued adjoint function $\gamma(t) = (\gamma_1(t), \gamma_2(t), \gamma_3(t))$, that is nonzero if $\gamma_0 = 0$, with the following properties along $p(t)$:

$$\dot{\gamma}_1 = -\frac{\partial F}{\partial x}, \tag{5.14}$$
$$\dot{\gamma}_2 = -\frac{\partial F}{\partial y}, \tag{5.15}$$
$$\dot{\gamma}_3 = -\frac{\partial F}{\partial \theta}, \tag{5.16}$$
$$F(\gamma(t), p(t), u(t)) = \max_{z\in[-1,1]} F(\gamma(t), p(t), z), \tag{5.17}$$
$$F(\gamma(t), p(t), u(t)) \equiv K, \tag{5.18}$$

for some constant K. Regular and abnormal extremals, corresponding to $\gamma_0 \neq 0$ and $\gamma_0 = 0$ respectively, are studied in the following two sections.

5.3.1 Regular Extremals

We may now scale F and assume $\gamma_0 = -\frac{1}{2}$. Rewriting the Hamiltonian we get $F = \gamma_1 \cos\theta + \gamma_2 \sin\theta + \gamma_3 u - \frac{u^2}{2}$. Maximization of F in (5.17) implies that

$$u = \begin{cases} -1 & \text{if} \quad \gamma_3 < -1 \\ \gamma_3 & \text{if} \quad -1 \leq \gamma_3 \leq 1 \cdot \\ 1 & \text{if} \quad \gamma_3 > 1 \end{cases} \tag{5.19}$$

Equations (5.14), (5.15), and (5.16) can be solved for γ to obtain

$$\gamma(t) = \begin{pmatrix} e_1 \\ e_2 \\ e_1 y - e_2 x + e_3 \end{pmatrix}, \tag{5.20}$$

in which e_1, e_2, and e_3 are constant. All e_i's cannot be zero, otherwise the extremal is a straight line. Let $\ell : e_1 y - e_2 x + e_3 = 0$, $\ell_+ : e_1 y - e_2 x + e_3 = 1$, and $\ell_- : e_1 y - e_2 x + e_3 = -1$ be three lines in the plane. The control law (5.19) says that $u = \gamma_3$ if the car is moving between ℓ_- and ℓ_+. Otherwise, $u = 1$ or -1 depending on the position of the car with respect to the lines. Figures 5.1 and 5.2 show a few examples of curves that satisfy such control law. These paths are composed of turn with minimum radius, straight line segment, and elastica [85, 174].

5.3.2 Abnormal Extremals

Abnormal extremals correspond to $\gamma_0 = 0$. If $K = 0$ in (5.18), then the extremal is of zero duration. If $K > 0$, then the extremal is a time-extremal for the Dubins car. If $K < 0$, then the extremal is also an extremal of the functional $I(u) = \int_0^T -dt$. We call such extremal a *locally longest curve*, because it can be a local minimum of $I(u)$, or equivalently a local maximum of the length functional.

Further analysis of (5.17) leads to the following control law:

$$u = \text{sgn}(\gamma_3) \text{ if } \gamma_3 \neq 0, \tag{5.21}$$

$$u \in [-1, 1] \text{ if } \gamma_3 = 0. \tag{5.22}$$

Depending on the sign of K, there are two different sets of extremals: time-extremals and locally longest curves.

$K > 0$, time-extremals

In this case, $F = e_1 \cos\theta + e_2 \sin\theta + |e_1 y - e_2 x + e_3| = K > 0$. Moreover, all e_i's cannot be zero. Thus, the extremal satisfies the Pontryagin Maximum Principle with the length cost functional $\int_0^T dt$. Thus, it is composed of turn with minimum radius and straight line segment. The extremal can tangentially join ℓ or diverge from ℓ. Figure 5.3 depicts two examples of time-extremals.

$K < 0$, locally longest curves

The following constraint holds:

$$F = e_1 \cos\theta + e_2 \sin\theta + |e_1 y - e_2 x + e_3| = K < 0. \tag{5.23}$$

In this case, the extremal cannot tangentially join ℓ unless it violates the constraint. Hence, either $u \equiv 0$ or $u(t) = \text{sgn}(e_1 y(t) - e_2 x(t) + e_3)$ and $e_1 \cos\theta + e_2 \sin\theta + |e_1 y - e_2 x + e_3| <$

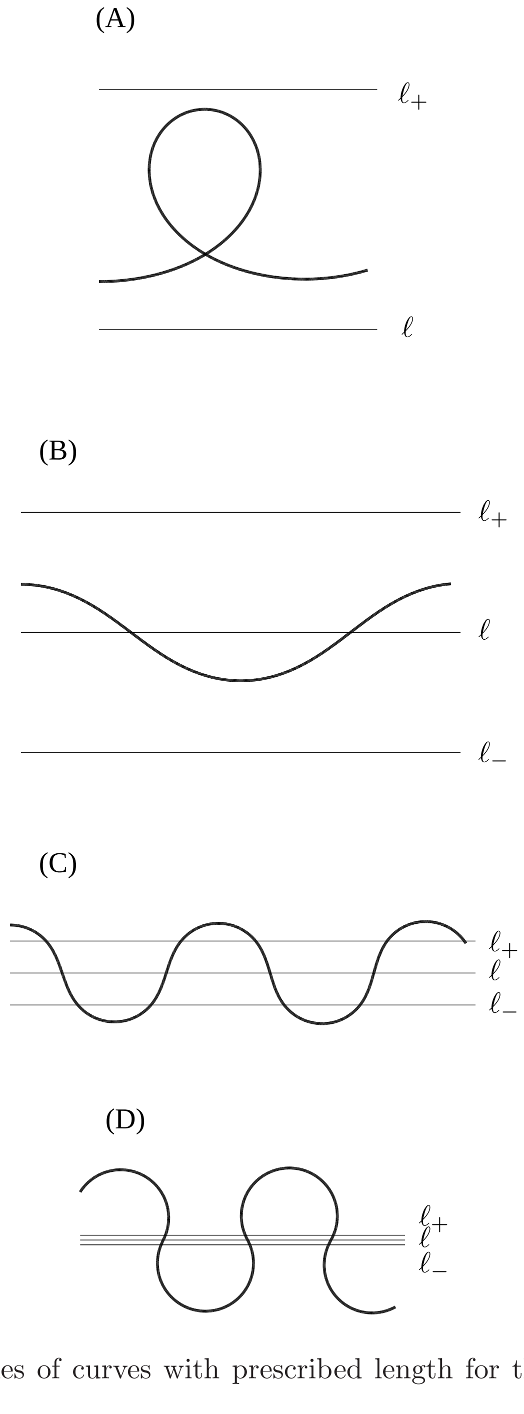


Figure 5.1: Some examples of curves with prescribed length for the Dubins car; see also Figure 5.2.

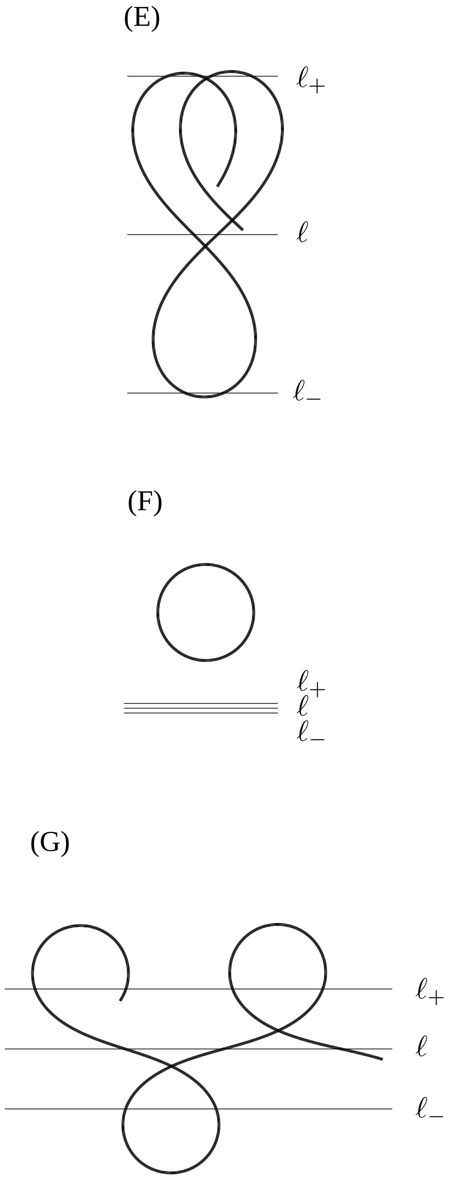


Figure 5.2: Continued from Figure 5.1.

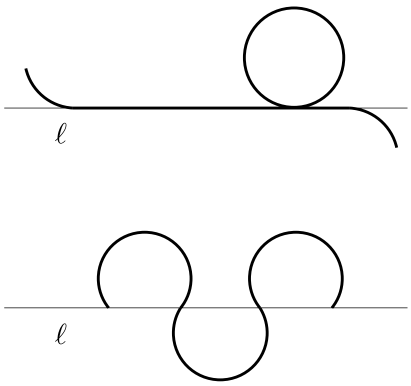


Figure 5.3: Two sample time-extremals for the Dubins car.

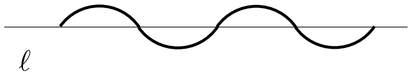


Figure 5.4: Locally longest curves for the Dubins car.

0. More precisely, either the extremal completely lies on ℓ, or it is composed of consecutive arcs of circle of length less than π. In Figure 5.4, the line ℓ and an example of a locally longest curve is shown. Figure 5.5 shows an elongation from $r(t)$, a Dubins shortest path, to a locally longest curve.

5.4 Time-optimal Trajectories for the Airplane

Going back to our original quest, which was to find time-optimal paths for our airplane, recall that the final altitude plays a major role. We distinguish three cases: low, medium, and high goal altitude. In order to precisely define each case we give the following definition.

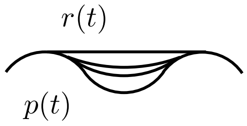


Figure 5.5: Elongation of a Dubins shortest path $r(t)$ to a locally longest curve $p(t)$.

Definition 5.2. Let Δ be the *Dubins distance* of (x_g, y_g, θ_g) from $(0, 0, 0)$. More precisely, let Δ denote the duration, or equivalently the length, of the shortest Dubins curve from $(0, 0, 0)$ to (x_g, y_g, θ_g). We call the final altitude *low* if $|z_g| \leq \Delta$, *medium* if $\Delta < |z_g| < \Delta + 2\pi$, and *high* if $|z_g| \geq \Delta + 2\pi$.

5.4.1 Time-optimal Trajectories for Low Goal Altitude

As we mentioned before, following the shortest Dubins curve with an unsaturated altitude velocity is a time-optimal strategy for low goal altitudes. This case corresponds to $c_3 = 0$ in the Pontryagin Maximum Principle analysis in Section 5.2. Note that the duration of such trajectory is Δ. It is obvious that there exists no trajectory transferring the system faster from the initial configuration to the goal configuration.

Lemma 5.3. *For a low goal altitude, a time-optimal trajectory for the system (2.8) consists of the shortest Dubins curve with altitude velocity* $u_z = \dfrac{z_g}{\Delta}$.

5.4.2 Time-optimal Trajectories for High Goal Altitude

If the goal altitude is high, the system has enough time to follow a helix once it reaches the goal point in the plane and goal orientation. Hence, the shortest Dubins curve followed by a helix all with saturated altitude velocity is a time-optimal strategy in this case. This case corresponds to $c_3 \neq 0$ in Section 5.2. The duration of such trajectory is $|z_g|$. There exists no trajectory taking the system faster from the initial to the goal.

Lemma 5.4. *For a high goal altitude, a time-optimal trajectory for the system (2.8) is composed of two pieces. Along both pieces* $u_z = sgn(z_g)$. *The projection of the first piece onto the* (x, y, θ)*-space is the shortest Dubins curve for* (x_g, y_g, θ_g). *The second piece is a helix. The control is* $u_\theta = \dfrac{2\pi}{|z_g| - \Delta}$ *along the second piece.*

The system first traverses the shortest Dubins curve with saturated altitude velocity along such time-optimal trajectory. It then traverses a helix, i.e. a full circle in the plane with saturated altitude velocity.

5.4.3 Time-optimal Trajectories for Medium Goal Altitude

If there is a path for the Dubins car from the initial configuration to the goal configuration in time $|z_g|$, then the time-optimal trajectory for the system corresponds to $c_3 \neq 0$ in Section 5.2. In this case, the altitude velocity is saturated. If there is no path for the Dubins car from the initial configuration to the goal configuration in time $|z_g|$, then the time-optimal trajectory for the system must correspond to $c_3 = 0$. The altitude velocity is not saturated in this case. Thus, the projection of the time-optimal trajectory onto the (x, y, θ)-space is a Dubins time-extremal. Dubins time-extremals are composed of turn with minimum radius and straight line segments. We presented both cases in Section 5.3.

5.5 Summary

We introduced the Dubins airplane which extends the Dubins car with altitude. We gave a characterization of the time-optimal trajectories for the Dubins airplane. For low and high final altitudes, the time-optimal trajectories respectively consist of the Dubins curve with unsaturated altitude velocity, and the Dubins curve followed by a helix with saturated altitude velocity. For medium altitudes in between low and high, different cases were recognized. The time-optimal trajectory is either a Dubins extremal (not the shortest) with unsaturated altitude velocity or a Dubins path of certain length with saturated altitude velocity. We gave a method to find a Dubins path with prescribed length if it exists. We also gave an analysis of locally longest curves for the Dubins car, i.e. those paths that may not be infinitesimally elongated. Numerical techniques can be used to compute the control synthesis. Analytical solution for the control synthesis remains open for this problem.

Chapter 6

Pareto-Optimal Coordination of Two Translating Polygonal Robots on a Roadmap

Collision-free coordination of multiple bodies is a fundamental problem that has received significant attention over the last decades. We study the problem of planning optimal motions of two polygonal robots, without differential constraints, under translation. Each robot has a reference point that must lie on a given graph, called a roadmap, which is embedded in the plane. The robots have a maximum speed and are capable of instantly switching to any speed between zero and the maximum. Each robot wants to move from its given initial location to its goal location without colliding with the other one. Equivalently, we want to find an optimal path between two points in the coordination space which is the Cartesian product of the roadmap, as a cell complex, with itself. Rather than impose an a priori cost scalarization for choosing the best combined motion, we consider finding motions whose cost vectors are Pareto-optimal. Pareto-optimal coordination strategies are the ones for which there exists no strategy that would be better for both robots.

For clarity of presentation, we first consider the case where the underlying graphs are trees. We present an algorithm that computes the complete set of Pareto-optimal coordination strategies in time $O(mn^2 \log n)$, in which m is the number of paths in the roadmap, and n is the number of coordination space vertices. Second, we present an algorithm that solves the general case. Our algorithm computes an upper bound on the cost of each motion in any Pareto-optimal coordination. Thus, only a finite number of homotopy classes of paths in the coordination space need to be considered. In effect, the new algorithm applies the acyclic algorithm to a finite portion of the universal cover of the roadmap. The algorithm computes solutions in time $O(2^{5\alpha} m^{1+5\alpha} n^2 \log(m^{2\alpha} n))$, in which m is the number of edges in the roadmap, n is the number of coordination space obstacle vertices, and $\alpha = 1 + \lceil (5\ell + r)/b \rceil$ where ℓ is total length of the roadmap and r is total length of coordination space obstacle boundary and b is the length of the shortest edge in the roadmap. This chapter was a joint work with Steven M. LaValle and Jason M. O'Kane [46].

6.1 Related Work

Planning motions of multibody systems such as reconfigurable robots and Autonomous Guided Vehicles (AGV) is an important problem and has been studied in various settings over the last decades [42, 67, 93, 180]. Previous approaches to multiple-robot motion planning are often categorized as *centralized* or *decoupled*. A centralized approach typically constructs a path in a composite configuration space, which is formed by the Cartesian product of the configuration spaces of the individual robots (e.g., [10, 15, 155]). A decoupled approach typically generates paths for each robot independently, and then considers the interactions between the robots (e.g., [5, 60, 136, 137, 138]). In [39, 130, 158], robot paths are independently determined, and a coordination diagram is used to plan a collision-free trajectory along the paths. In [104, 165], an independent roadmap is computed for each robot, and coordination occurs on the Cartesian product of the roadmap path domains. The suitability of one approach over the other is usually determined by the tradeoff between computational complexity associated with a given problem, and the amount of completeness that is lost. In some applications, such as the coordination of AGVs, the roadmap might represent all allowable mobility for each robot.

In this chapter, we study the problem of planning optimal motions of two polygonal robots traveling on a given roadmap. The robots have a maximum speed and are capable of instantly switching to any speed between zero and the maximum. The initial and goal locations are given for each robot; see Figure 6.1. The robots must be disjoint when they travel, and as a result, there are tradeoffs between the robots' completion times. Equivalently, we want to find an optimal path between two points in the coordination space which is the Cartesian product of the roadmap, as a cell complex, with itself. One approach is to consider a scalar cost that combines the completion times. Minimizing the average time robots take to reach their goals [81, 126], and minimizing the time that the last robot takes have been studied before [162]. The problem with scalarization is that it eliminates many

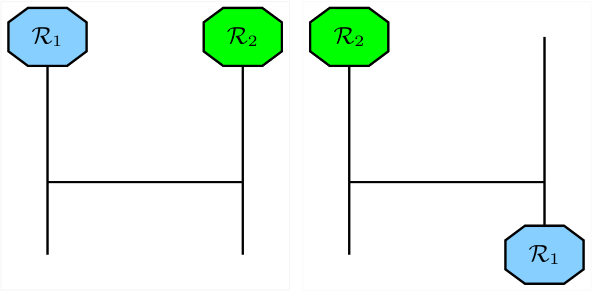


Figure 6.1: A sample coordination problem that can be solved by our methods. [left] The initial configuration. [right] The goal configuration.

interesting coordination strategies, possibly even neglecting optimality for some robots [104]. Rather than impose an a priori scalarization for choosing the best combined motion, we consider finding motions whose cost vectors (cost of robot 1, cost of robot 2) are Pareto-optimal. A sample problem and its Pareto-optimal solutions are illustrated in Figures 6.2, 6.3. Pareto-optimal coordination strategies are the ones for which there exists no strategy that would be better for both robots. The notion of Pareto optimality induces a partial order on the space of coordinations. Pareto-optimal coordinations are the minimal elements of the partial order. For a detailed introduction to Pareto optimality, see [152].

Optimal coordinations according to a scalar cost impose a predetermined preference between the robots, whereas having all Pareto-optimal coordinations beforehand gives the freedom to determine the preference at run-time. The approach can be considered as filtering out all of the motion plans that are not worth considering and presenting the user with a small set of the best alternatives. Within this framework, additional criteria, such as priority or the amount of sacrifice one robot makes, can be applied to automatically select a particular motion plan. If the same tasks are repeated and priorities change, then one only needs to select an alternative minimal plan, as opposed to re-exploring the entire space of motion strategies. It was shown that the number of Pareto-optimal coordinations for n robots on any roadmap is finite [68]; therefore, it is plausible to seek all of them. Figure 6.2 shows a coordination problem on a roadmap with 7 edges and Figure 6.3 illustrates the four Pareto-optimal solutions for the problem.

This work is inspired by previous approaches to multiple robot coordination. O'Donnell

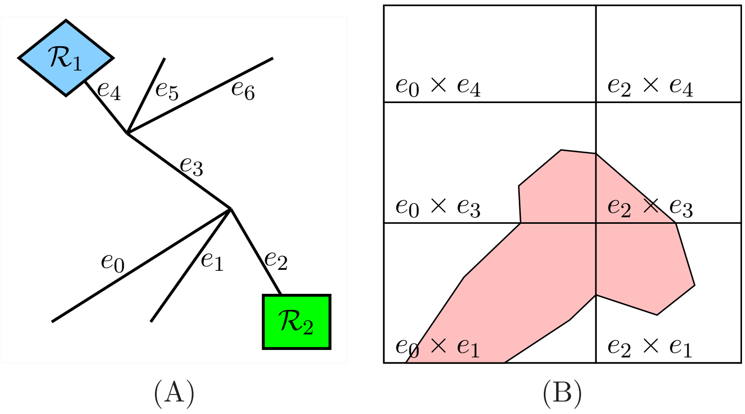


Figure 6.2: (A) A coordination problem on a roadmap with 7 edges. (B) A subset of $\mathcal{G} \times \mathcal{G}$ for this problem.

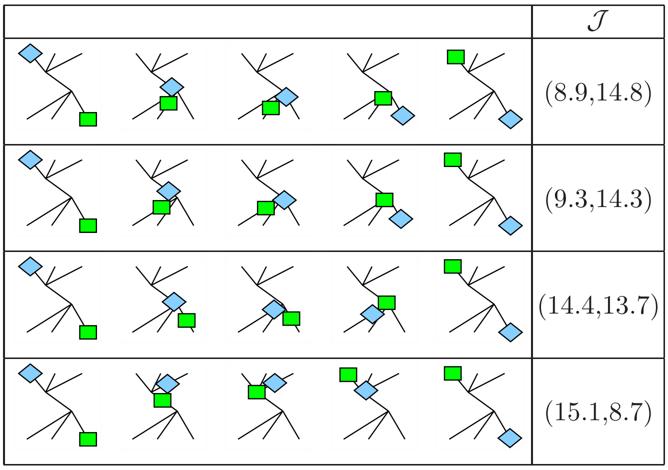


Figure 6.3: The four Pareto-optimal solutions for the problem in Figure 6.2.

and Lozano-Pérez introduced coordination diagrams for planning motions of two robot manipulators [130]. Alt and Godau used similar coordination spaces in a different context to compute the Fréchet distance between two polygonal curves [7]. LaValle and Hutchinson gave the first approach to Pareto-optimal coordination of multiple robots [104]. They presented an approximation algorithm based on dynamic programming in the discretized coordination space. Ghrist et al. gave a characterization of Pareto-optimal coordinations of multiple robots using CAT(0) geometry [69]. They provided an algorithm to shorten a given coordination to a homotopic, possibly Pareto-optimal one.

The coordination space in our case is a two dimensional non-positively curved (NPC) metric space, with nontrivial fundamental group if the roadmap is cyclic. Note that it is not necessarily a manifold. Using shortest path algorithms in the plane such as continuous Dijkstra [79, 124] or visibility graph methods in the universal cover of the coordination space, one can compute the shortest paths. However, those methods are not directly applicable in our problem because the roadmap can be cyclic, and consequently the universal cover can be unbounded. Moreover, an incremental exploration of the unbounded universal cover may never stop, because there are multiple Pareto-optimal coordinations whose maximum length is unknown beforehand. This is simply a fascinating kind of shortest path problem, because the space is unbounded and there are multiple solutions. In addition, paths are partitioned into equivalence classes by $\mathcal{L}^\infty$ equivalence. Our algorithm constructs a bounded portion of the universal cover in which the shortest path algorithm is applied. After the relevant portion of the universal cover is constructed, the rest is similar to the acyclic case.

We give a brief formulation of the problem in Section 6.2. We fix a canonical form for Pareto-optimal coordinations and translate the problem into a search for shortest paths in the coordination space in Section 6.3. We give the algorithm for an acyclic roadmap in Section 6.4. The acyclic algorithm computes the visibility graph, augments it with some extra edges, and finds the shortest paths. We present the general algorithm steps in Section 6.5: 1) compute an upper bound on the cost of each motion in any Pareto-optimal coordination, 2) construct a finite portion of the universal cover of the coordination space, and 3) apply the acyclic algorithm. Finally, Section 6.7 concludes the chapter and mentions

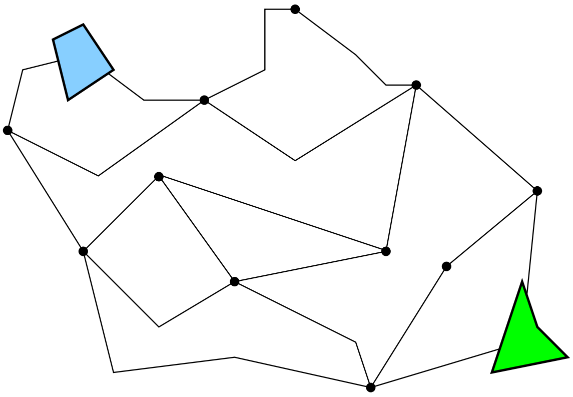


Figure 6.4: Polygonal robots on a piecewise linear roadmap.

interesting open problems.

6.2 Problem Formulation

Let the robots, $\mathcal{R}_1$ and $\mathcal{R}_2$, be polygonal open sets embedded in the plane. They translate along a roadmap $\mathcal{G}$, which is an embedded graph in the plane[1]. Edges of $\mathcal{G}$ are piecewise-linear segments. See Figure 6.4 for an example. The roadmap need not be connected, so effectively each robot can have its own roadmap. Each edge of $\mathcal{G}$ is weighted by its Euclidean length. In this way, $\mathcal{G}$ turns into a metric graph [30]. The robots have a maximum speed and are capable of instantly switching to any speed between zero and the maximum. By scaling the respective metric graphs, we assume without loss of generality that both robots have unit maximum speed. Under this assumption, the distance function $d(x, y)$ gives the minimum amount of time that it takes $\mathcal{R}_i$ to go from x to y on $\mathcal{G}$.

We are given an initial and a goal configuration $q_i^{init}, q_i^{goal} \in \mathcal{G}$ for each robot $\mathcal{R}_i$. A coordination is a continuous path in the coordination space $\mathcal{G} \times \mathcal{G}$, from $q^{init} = (q_1^{init}, q_2^{init})$ to $q^{goal} = (q_1^{goal}, q_2^{goal})$, that avoids collision between the robots and satisfies speed bounds. Figure 6.5 illustrates a graph times a single edge. The coordination space has a similar structure; it is composed of similar rectangular pieces glued together along their respective boundaries. The obstacle region, denoted by $O \subset \mathcal{G} \times \mathcal{G}$, is the set of configurations at which

[1] If we assume that $\mathcal{G}$ is locally embedded in the plane, in which case its edges may intersect, then our algorithm correctly works and our results still hold. For the sake of clarity, we preferred to assume $\mathcal{G}$ is embedded.

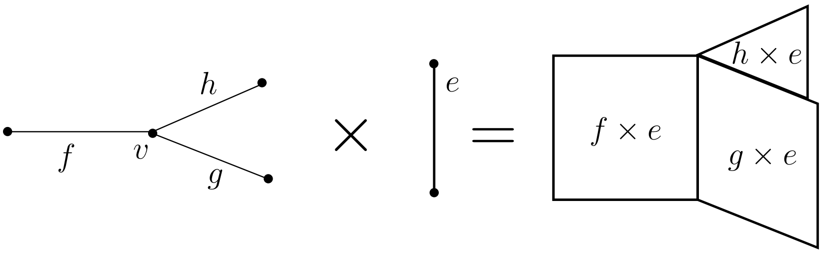


Figure 6.5: A graph times a single edge.

$\mathcal{R}_1$ and $\mathcal{R}_2$ collide. Since the robots are polygonal and roadmap paths are piecewise-linear, the obstacle region is a collection of polygonal, open connected components.

The vector-valued cost $\mathcal{J} = (J_1, J_2)$ separately measures the time that each robot takes to reach its goal and stop. Let $\mathscr{C}$ be the set of all coordinations, i.e. all continuous paths in $\mathcal{G} \times \mathcal{G}$ from q^{init} to q^{goal}. The cost $\mathcal{J} : \mathscr{C} \to [0, \infty)^2$ induces a partial order on the set of all coordinations $\mathscr{C}$. Each minimal element in this partial order is called a Pareto-optimal coordination. The problem is to find all Pareto-optimal coordinations for the two robots.

To specify $\mathcal{J}$, define a metric d_∞ in $\mathcal{G} \times \mathcal{G}$ that gives the minimum amount of time that $\mathcal{R}_1$ and $\mathcal{R}_2$ take to go from (x_1, x_2) to (y_1, y_2) ignoring collisions. It is naturally defined by $d_\infty : ((x_1, x_2), (y_1, y_2)) \mapsto \max(d(x_1, y_1), d(x_2, y_2))$, in which d is the metric that made $\mathcal{G}$ into a metric graph. Let $\mathcal{L}^\infty$ be the functional that gives the length of each continuous path in $\mathcal{G} \times \mathcal{G}$ according to d_∞. For each coordination $\gamma = (\gamma_1, \gamma_2) : [0, 1] \to \mathcal{G} \times \mathcal{G}$, let $t_i = \min\{t \in [0, 1] : \gamma_i([t, 1]) = q_i^{goal}\}$. In that case, $J_i(\gamma) = \mathcal{L}^\infty(\gamma|_{[0,t_i]})$ and $\mathcal{J}(\gamma) = (J_1(\gamma), J_2(\gamma))$.

6.3 Canonical Pareto-optimal Coordinations

Different paths that have the same end points can have equal $\mathcal{L}^\infty$ lengths in the coordination space. Consequently, there are different coordinations with equal cost $\mathcal{J}$. In this section, we fix a canonical form for equivalent Pareto-optimal coordinations based on Euclidean shortest paths. The following proposition precisely presents our canonical form and proves its existence.

Proposition 6.1. *For every Pareto-optimal coordination, there is an equivalent coordina-*

tion that is composed of a finite sequence of Euclidean shortest segments between the vertices of the obstacle region, q^{init}, q^{goal}, and in some cases (x, q_2^{goal}) or (q_1^{goal}, x).

Proof sketch: We first choose Euclidean shortest paths as canonical form for $\mathcal{L}^\infty$-shortest paths in $\mathcal{G} \times \mathcal{G} \backslash O$. Note that a Euclidean shortest path is also $\mathcal{L}^\infty$-shortest. An argument similar to the one in [52], which is essentially based on shortening, shows Euclidean shortest paths in $\mathcal{G} \times \mathcal{G} \backslash O$ comprise Euclidean shortest segments between the vertices of the obstacle region O and the two end points.

We now choose a canonical form for a Pareto-optimal coordination γ. If robot $\mathcal{R}_1$ reaches its goal first under γ, then the final segment of γ is (q_1^{goal}, x) to (q_1^{goal}, q_2^{goal}) for some $x \in \mathcal{G}$. In that case, let $\tilde{\gamma}$ be that part of γ that goes from q^{init} to (q_1^{goal}, x). Likewise, the final segment of γ is (x, q_2^{goal}) to (q_1^{goal}, q_2^{goal}) if robot $\mathcal{R}_2$ reaches its goal first. In that case, let $\tilde{\gamma}$ be that part of γ that goes from q^{init} to (x, q_2^{goal}). If both robots simultaneously reach their goals, then let $\tilde{\gamma} = \gamma$. It is obvious that $\tilde{\gamma}$ is an $\mathcal{L}^\infty$-shortest path; otherwise, γ cannot be Pareto-optimal. Given the canonical form for $\mathcal{L}^\infty$-shortest paths, there is a path equivalent to $\tilde{\gamma}$ that is composed of a finite sequence of Euclidean shortest segments between the vertices of the obstacle region, q^{init}, and the final point of $\tilde{\gamma}$. Eventually, that part of γ that is not in $\tilde{\gamma}$ can be made a Euclidean shortest path. ■

6.4 Algorithm for Acyclic Roadmap

In this section, we present an algorithm for the case in which $\mathcal{G}$ is acyclic. To present the algorithm, we first define a single coordination cell, which is the coordination space of the two robots on two fixed paths.

6.4.1 Coordination Cell

Since $\mathcal{G}$, as a cell complex, consists of 0-dimensional and 1-dimensional cells, $\mathcal{G} \times \mathcal{G}$ is a cube-complex. In fact, $\mathcal{G} \times \mathcal{G}$ consists of a number of 2-dimensional cells appropriately glued to each other along their boundary edges and vertices. Each such 2D cell, $D = e_r \times e_s$, in which $e_r, e_s \subset \mathcal{G}$, can be seen as the coordination space of the two robots on the paths e_r

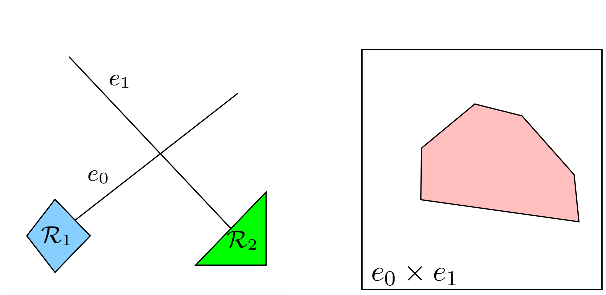


Figure 6.6: A pair of path segments and their coordination cell.

and e_s. In particular, our coordination cell can be seen as $[0, l_r] \times [0, l_s]$, in which $l_k = l(e_k)$ is the length of e_k.

Within each coordination cell, the obstacle region is the set of points corresponding to positions in which the interiors of $\mathcal{R}_1$ and $\mathcal{R}_2$ intersect. The free region is set of points not in the obstacle region. In Figure 6.6, we see an example of a coordination cell and its obstacle region. Notice that our coordination cell is similar to the coordination diagram of [158], but since our robots are polygonal and our paths are piecewise-linear, the obstacle region in our coordination cell is a collection of polygonal connected components. If we confine our attention to a single coordination cell (as we will in Section 6.4.2), a coordination is essentially a piecewise-smooth path from $(0, 0)$ to (l_r, l_s) inside its free region.

To describe the algorithm, we first describe how to compute all Pareto-optimal coordinations in the simpler case of a single coordination cell, then extend the algorithm to the whole $\mathcal{G} \times \mathcal{G}$ which consists of a collection of such coordination cells.

6.4.2 Algorithm for Two Fixed Paths

In this section we describe how to compute all Pareto-optimal coordinations in a single coordination cell, i.e. for the two robots on two fixed paths. As it is stated in Section 6.2, the obstacle (or collision) region of our coordination cell consists of a collection of polygons. Thus, we may use the terms *vertex* and *edge* of the obstacle region. To present the algorithm, we give some statements about the properties of Pareto-optimal coordinations.

As a consequence of Proposition 6.1, it is sufficient to consider only canonical coordinations, i.e. those coordinations that are composed of a sequence of linear segments between the vertices of obstacle region, initial and goal points, and in some cases a point on the boundary of coordination cell. The next lemma explains relevant vertices on the boundary of coordination cell.

Lemma 6.2. *Suppose τ is a canonical Pareto-optimal coordination. Let (t_1, t_2) denote the last vertex of τ that is not on the boundary, i.e. $t_1 \neq l_r$ and $t_2 \neq l_s$. There are three cases:*

(i) If $J_1(\tau) < J_2(\tau)$, then the line segment $(t_1, t_2) - (l_r, t_2 + l_r - t_1)$ is collision free; furthermore, it is exactly a segment of τ.

(ii) If $J_1(\tau) > J_2(\tau)$, then the line segment $(t_1, t_2) - (t_1 + l_s - t_2, l_s)$ is collision free; furthermore, it is exactly a segment of τ.

(iii) There is at most one canonical τ such that $J_1(\tau) = J_2(\tau)$. It is the Euclidean shortest path on the visibility graph of obstacle vertices and endpoints.

Proof. In the first two cases, if the line segment is not collision free, we can always find another coordination which reduces both J_1 and J_2, contradicting the optimality of τ. Furthermore, if the line segment is not part of τ, we can replace it in and find a better coordination. In the third case, τ is the Euclidean shortest path from $(0, 0)$ to (l_r, l_s) in the interior of coordination cell. ■

Corollary 6.3. *The number of Pareto-optimal coordinations of two polygonal robots on two fixed paths is finite.*

Note that in case (i) of Lemma 6.2, $(l_r, t_2 + l_r - t_1)$ is simply the intersection of the line $x_1 = l_r$ and the line with unit slope through (t_1, t_2). Similar remarks can be made for cases (ii) and (iii). Intuitively, we can think of shooting a ray at slope 1 from each obstacle vertex (t_1, t_2) and stopping when that ray hits a point with either $x_1 = l_r$ or $x_2 = l_s$, corresponding respectively to $\mathcal{R}_1$ or $\mathcal{R}_2$ reaching its goal. Lemma 6.2 tells us that every canonical Pareto-optimal coordination ends with such a unit-slope segment.

Now we are ready to present the algorithm in Figure 6.7. The function Obstacle-Polygons computes the obstacle region polygons. The obstacle region is a collection of polygons which can be computed by collision detection algorithm along each pair of linear

```
CoordinateSingleCell($e_r, e_s, \mathcal{R}_1, \mathcal{R}_2$)
$P \leftarrow$ ObstaclePolygons($e_r, e_s, \mathcal{R}_1, \mathcal{R}_2$)
$VG \leftarrow$ VisibilityGraph($P \cup \{(0,0)\}$)
Dijkstra($VG, (0,0), \mathcal{L}^\infty$)

$S \leftarrow \emptyset$
for each vertex $v = (x_1, x_2)$ of each polygon in $P$ do
  if $l_r - x_1 < l_s - x_2$ then
    $q \leftarrow (l_r, x_2 + l_r - x_1)$
  else
    $q \leftarrow (x_1 + l_s - x_2, l_s)$
  end if
  if Free($P, v, q$) and Free($P, q, (l_r, l_s)$) then
    $S \leftarrow S \cup \{$(ShortestPath($(0,0), v$), $q, (l_r, l_s)$)$\}$
  end if
end for

return PruneDominatedSolutions($S$)
```

Figure 6.7: The basic algorithm for two fixed paths. The robots $\mathcal{R}_1$ and $\mathcal{R}_2$ move along e_r and e_s respectively.

path segments. More precisely, we build the Minkowski sum of $\mathcal{R}_2$ on $\mathcal{R}_1$. The intersection points of the roadmap path segments with this polygon give the boundary of obstacle region. The visibility graph of the vertices of obstacle region and endpoints is computed in VisibilityGraph according to the well-known radial sweep algorithm, which was summarized in Section 2.4. The function Free determines whether a line segment is contained in the free region of the coordination cell. This can be performed by simple geometric tests. The optimal path candidates described in Lemma 6.2 are computed by iterating over the vertices of obstacle region. Lastly, the Pareto-optimal solutions are extracted from this set of candidates by simple pairwise comparisons in PruneSolutions. Figure 6.8 shows the visibility graph of a coordination cell, augmented with its full-speed completions, and the two Pareto optimal coordinations extracted from this graph.

Theorem 6.4. *The algorithm* CoordinateSingleCell *in Figure 6.7 correctly computes all Pareto-optimal coordinations of the two robots on two fixed piecewise-linear paths.*

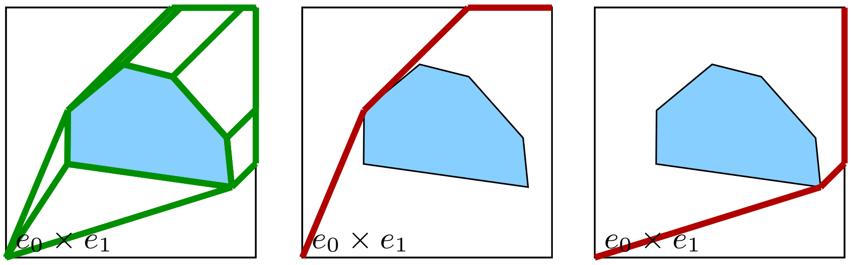


Figure 6.8: [left] The visibility graph of a coordination cell, augmented with its unit-slope completions. [right] The two Pareto optimal coordinations extracted from this graph.

Proof. The result directly follows from Proposition 6.1 and Lemma 6.2. ■

If n denotes the number of obstacle vertices, then VisibilityGraph takes $O(n^2 \log n)$ time. Since each of the other steps can be done in $O(n^2)$ time, the time complexity of CoordinateSingleCell is also $O(n^2 \log n)$.

6.4.3 Acyclic Algorithm Presentation

In this section we extend the coordination cell algorithm in Figure 6.7 to the general case of two robots on an acyclic roadmap $\mathcal{G}$. The theory developed in [71] shows that if $\mathcal{G}$ is acyclic, $\mathcal{G} \times \mathcal{G}$ with Euclidean metric is non-positively curved (NPC), and consequently it has unique Euclidean geodesics. For detailed information on NPC spaces see [30]. This result implies the following proposition.

Proposition 6.5. *Assume $\mathcal{G}$ is an acyclic graph. Equip $\mathcal{G} \times \mathcal{G}$ with the Euclidean metric. Between any two points x, y in the same connected component of $\mathcal{G} \times \mathcal{G}$ there is exactly one geodesic connecting x and y.*

This property makes $\mathcal{G} \times \mathcal{G}$ similar to the plane, because geodesics in $\mathcal{G} \times \mathcal{G}$ play the role of lines in plane. In fact, geodesics inside a coordination cell coincide with the usual Euclidean lines.

Lemma 6.6. *Assume τ is a canonical Pareto-optimal coordination. Let $A = (x_1, x_2)$ be the last vertex of τ such that $x_1 \neq q_1^{goal}$ and $x_2 \neq q_2^{goal}$. Once again, there are three cases:*

(i) If $J_1(\tau) < J_2(\tau)$, then the geodesic segment A to (q_1^{goal}, y) with equal progression for $\mathcal{R}_1$ and $\mathcal{R}_2$ is collision free; furthermore, it is exactly a segment of τ.

(ii) If $J_1(\tau) > J_2(\tau)$, then the geodesic segment A to (y, q_2^{goal}) with equal progression for $\mathcal{R}_1$ and $\mathcal{R}_2$ is collision free; furthermore, it is exactly a segment of τ.

(iii) There is at most one canonical τ such that $J_1(\tau) = J_2(\tau)$. It is the Euclidean shortest path on the generalized visibility graph of obstacle vertices and endpoints.

Proof. In the first two cases, if the geodesic segment is not collision free, we can always find another coordination which reduces both J_1 and J_2, contradicting the optimality of τ. If the geodesic segment is not part of τ, we can replace it in and find a better coordination. In the third case, τ is the shortest path from q^{init} to q^{goal} according to the Euclidean metric. ■

Corollary 6.7. *The number of Pareto-optimal coordinations for two polygonal robots on a piecewise-linear acyclic roadmap is finite.*

A generalization of this result is given in [68].

In CoordinateAcyclic in Figure 6.9, GenVisibilityGraph is a generalization of visibility graph algorithm in Section 2.4. More precisely, we perform a radial sweep in $\mathcal{G} \times \mathcal{G}$. This can be done because the radial geodesics are unique. To sweep about vertex v, sort all the obstacle vertices throughout the cell complex in their geodesic angle order. We extend the standard radial sweep visibility graph algorithm by maintaining a separate balanced binary tree for each 2-cell in $\mathcal{G} \times \mathcal{G}$ intersected by the sweep ray. Edges in each tree remain ordered according to their distance from v. To check whether a geodesic is collision free, we check collision for all the nearest edges given by our tree data structure in those cells that are traversed by the geodesic. The remainder of the algorithm is essentially unchanged from CoordinateSingleCell.

Theorem 6.8. *The algorithm* CoordinateAcyclic *in Figure 6.9 correctly computes all Pareto-optimal coordinations of the two robots on $\mathcal{G}$ from q^{init} to q^{goal}. Moreover, its total*

```
CoordinateAcyclic($\mathcal{G}, \mathcal{R}_1, \mathcal{R}_2, q^{init}, q^{goal}$)
$P \leftarrow \emptyset$
for each pair of edges $e_i, e_j \subset \mathcal{G}$ do
   $P \leftarrow P \cup$ ObstaclePolygons($e_i, e_j, \mathcal{R}_1, \mathcal{R}_2$)
end for
$VG \leftarrow$ GenVisibilityGraph($P \cup \{q^{init}\}$)
Dijkstra($VG, q^{init}, \mathcal{L}^\infty$)

$S \leftarrow \emptyset$
for each vertex $v = (x_1, x_2)$ of each polygon in $P$ do
   if $d(x_1, q_1^{goal}) < d(x_2, q_2^{goal})$ then
      $q \leftarrow (q_1^{goal}, x_2 + d(x_1, q_1^{goal}))$
   else
      $q \leftarrow (x_1 + d(x_2, q_2^{goal}), q_2^{goal})$
   end if
   if Free($P, v, q$) and Free($P, q, q^{goal}$) then
      $S \leftarrow S \cup \{($ShortestPath($q^{init}, v$)$, q, q^{goal})\}$
   end if
end for

return PruneDominatedSolutions($S$)
```

Figure 6.9: The algorithm for finding all Pareto-optimal coordinations of two robots on an acyclic piecewise-linear roadmap.

complexity is $O(mn^2 \log n)$, in which m is the number of edges in $\mathcal{G}$, and n is the number of obstacle vertices.

Proof. Correctness directly follows from Proposition 6.1 and Lemma 6.6. Since each geodesic passes through at most $2m$ cells, in computing the visibility graph, we perform $O(mn^2)$ balanced binary tree operations, each taking $O(\log n)$ time. The visibility graph therefore requires $O(mn^2 \log n)$ time to compute. Both Dijkstra's algorithm and the pruning of S take $O(n^2)$ time. Finally, notice that the number of Pareto-optimal coordinations is at most $n + 2$. Thus, the complexity of algorithm output is $O(n)$. Hence, the total complexity of our algorithm is $O(mn^2 \log n)$. ■

6.5 Algorithm for Cyclic Roadmap

To find canonical Pareto-optimal coordinations for the cyclic case, our algorithm computes Euclidean shortest segments between obstacle vertices, initial and goal configurations, and some points (q_1^{goal}, x) and (x, q_2^{goal}) in the coordination space. The points (q_1^{goal}, x) and (x, q_2^{goal}) that need to be considered are characterized in Lemma 6.6. A point (q_1^{goal}, x) or (x, q_2^{goal}) needs to be considered if there is a collision-free Euclidean shortest segment, with equal progression for $\mathcal{R}_1$ and $\mathcal{R}_2$, from an obstacle vertex or q^{init} to the point (q_1^{goal}, x) or (x, q_2^{goal}). To find the shortest path between two points, we enumerate all homotopy classes and find the shortest path in every class. Fixing the end points in the coordination space, there is only one shortest path in every homotopy class, which holds because the space is non-positively curved [68]. To compute these paths, our algorithm partially builds the universal cover of $\mathcal{G} \times \mathcal{G}$, and finds the shortest path in the universal cover. Using a cost upper bound computed in advance, our algorithm constructs the relevant part of the universal cover. The rest of the algorithm is essentially identical to the acyclic case applied to the universal cover.

6.5.1 Coordination Cost Upper Bound

In a scalar minimization problem, the cost of any feasible solution is an upper bound for the cost of an optimal solution. The key idea here is the same. An upper bound for the cost of any Pareto-optimal coordination can be derived from an arbitrary coordination. The following lemma precisely derives an upper bound on the cost of every motion in any Pareto-optimal coordination.

Lemma 6.9. *Let $\Delta_1, \Delta_2 \subseteq \mathcal{G}$ be such that $\{q_1^{goal}\} \times \Delta_2 = \{q_1^{goal}\} \times \mathcal{G} \backslash O$, and $\Delta_1 \times \{q_2^{goal}\} = \mathcal{G} \times \{q_2^{goal}\} \backslash O$. Let δ_i be the diameter of Δ_i as a metric graph. Let λ be the Euclidean length of an arbitrary coordination γ. Let τ be a Pareto-optimal coordination. In that case, $J_1(\tau), J_2(\tau) \leq \lambda + \delta$, in which $\delta = \max(\delta_1, \delta_2)$.*

Proof. We claim that either $J_1(\tau) \leq \lambda$ or $J_2(\tau) \leq \lambda$. Suppose on the contrary, $J_1(\tau) > \lambda \geq J_1(\gamma)$ and $J_2(\tau) > \lambda \geq J_2(\gamma)$. In that case, γ is a better coordination than τ. That is contradictory to Pareto-optimality of τ. Suppose that $J_1(\tau) \leq \lambda$ and $\mathcal{R}_1$ reaches its goal first. Once $\mathcal{R}_1$ stops at its goal, robot $\mathcal{R}_2$ needs to travel along a roadmap path whose length is at most the diameter of the free portion of the roadmap. The free portion of the roadmap is Δ_2. Hence, $J_2(\tau) \leq J_1(\tau) + \delta_2 \leq \lambda + \delta_2$ because the travel time for both $\mathcal{R}_1$ and $\mathcal{R}_2$ is $J_1(\tau)$ up to the moment $\mathcal{R}_1$ stops, and is at most δ_2 for $\mathcal{R}_2$ afterwards. Thus, $J_1(\tau), J_2(\tau) \leq \lambda + \delta_2$.

Similarly, $J_1(\tau), J_2(\tau) \leq \lambda + \delta_1$ if $J_2(\tau) \leq \lambda$ and $\mathcal{R}_2$ reaches its goal first. Therefore, $J_1(\tau), J_2(\tau) \leq \lambda + \delta$, in which $\delta = \max(\delta_1, \delta_2)$. ■

To compute λ, which is the Euclidean length of an arbitrary coordination γ, we use the dimension reduction method of Aronov et al. [11]. Denote the boundary of obstacle region by ∂O. Define

$$\Upsilon_1 = \{q_1^{init}\} \times \mathcal{G} \backslash O, \tag{6.1}$$

$$\Upsilon_2 = \mathcal{G} \times \{q_2^{init}\} \backslash O, \tag{6.2}$$

$$\Upsilon_3 = \{q_1^{goal}\} \times \mathcal{G} \backslash O, \tag{6.3}$$

$$\Upsilon_4 = \mathcal{G} \times \{q_2^{goal}\} \backslash O, \tag{6.4}$$

and

$$\Sigma = \partial O \cup (\bigcup_{j=1}^{4} \Upsilon_j). \tag{6.5}$$

We call Σ the *skeleton* of $\mathcal{G} \times \mathcal{G} \backslash O$. See Figure 6.10 for a simple example. Note that the skeleton is a one-dimensional object. It is composed of five pieces: $\mathcal{R}_1$ at its initial, $\mathcal{R}_2$ at its initial, $\mathcal{R}_1$ at its goal, $\mathcal{R}_2$ at its goal, and $\mathcal{R}_1$ touching $\mathcal{R}_2$. The following lemma follows from Lemma 1 in [11].

Lemma 6.10 ([11])**.** *There is a collision-free path from q^{init} to q^{goal} in the coordination space if and only if there is a path from q^{init} to q^{goal} in Σ, the skeleton of $\mathcal{G} \times \mathcal{G} \backslash O$.*

Our algorithm constructs Σ by gluing ∂O and Υ_j along their intersection points. We discussed how to compute the obstacle region in Section 6.4.2. To compute Υ_j, first we compute $\mathcal{M} = \mathcal{R}_1 \ominus \mathcal{R}_2$, the Minkowski difference. By intersecting polygon $\mathcal{M}$ po-

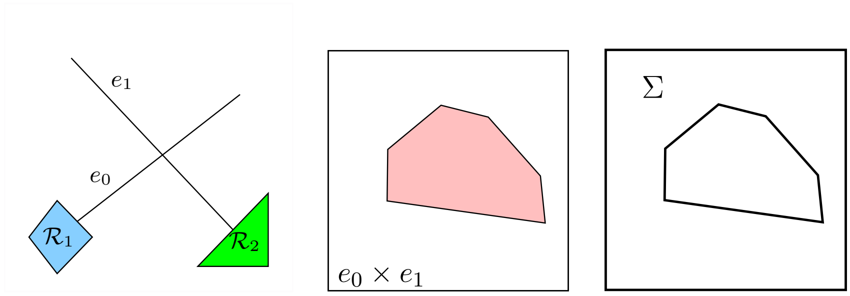


Figure 6.10: A sample problem, a subset of its coordination space, and the skeleton of the depicted part.

sitioned respectively at q_1^{init} and q_1^{goal} with $\mathcal{G}$, we compute $\Gamma_2 = \mathcal{G}\backslash(\{q_1^{init}\} \oplus \mathcal{M})$ and $\Delta_2 = \mathcal{G}\backslash(\{q_1^{goal}\} \oplus \mathcal{M})$. By intersecting $-\mathcal{M}$ positioned respectively at q_2^{init} and q_2^{goal} with $\mathcal{G}$, we compute $\Gamma_1 = \mathcal{G}\backslash(\{q_2^{init}\} \ominus \mathcal{M})$ and $\Delta_1 = \mathcal{G}\backslash(\{q_2^{goal}\} \ominus \mathcal{M})$. It is enough to observe that $\Upsilon_1 = \{q_1^{init}\} \times \Gamma_2$, $\Upsilon_2 = \Gamma_1 \times \{q_2^{init}\}$, $\Upsilon_3 = \{q_1^{goal}\} \times \Delta_2$, and $\Upsilon_4 = \Delta_1 \times \{q_2^{goal}\}$. Dijkstra's algorithm yields γ and the minimum distance of q^{goal} from q^{init} in Σ which is λ. Finally, the diameter, or an overestimate of the diameter, of Δ_i yields δ_i. Recall that the upper bound is $\lambda + \max(\delta_1, \delta_2)$.

6.5.2 Universal Cover of $\mathcal{G} \times \mathcal{G}$

Given the upper bound computed in Section 6.5.1, we only need to consider a finite portion of the universal cover. Here we describe an algorithm to construct it. Let $\mathcal{X}$ be the universal cover of $\mathcal{G}$ as a cell complex. In that case, $\mathcal{X} \times \mathcal{X}$ is the universal cover of $\mathcal{G} \times \mathcal{G}$, and it is enough to build the relevant part of $\mathcal{X}$ to construct the relevant part of $\mathcal{X} \times \mathcal{X}$.

Since $\mathcal{X}$ is composed of disjoint copies of a fundamental domain glued along identified vertices, we describe how to build a fundamental domain, denoted by $\mathcal{X}_0$. Let $\mathcal{T}$ be any spanning tree of $\mathcal{G}$ (a collection of trees if $\mathcal{G}$ is not connected). Let $e_i = (u_i, v_i), i = 1, \ldots, k$ be those edges of $\mathcal{G}$ that are not in $\mathcal{T}$. Obtain $\mathcal{X}_0$, the fundamental domain of $\mathcal{X}$, by adding k new vertices u_i^* and k edges (v_i, u_i^*) to $\mathcal{T}$. Note that the length of (v_i, u_i^*) is the same as that of (u_i, v_i). Cycles of $\mathcal{G}$ are opened into paths in $\mathcal{X}_0$. Vertices u_i^* must be identified with u_i in neighboring copies of the fundamental domain. We call u_i and u_i^* *gluing spots* of $\mathcal{X}_0$, because $\mathcal{X}$ is obtained by iteratively gluing disjoint copies of the fundamental domain

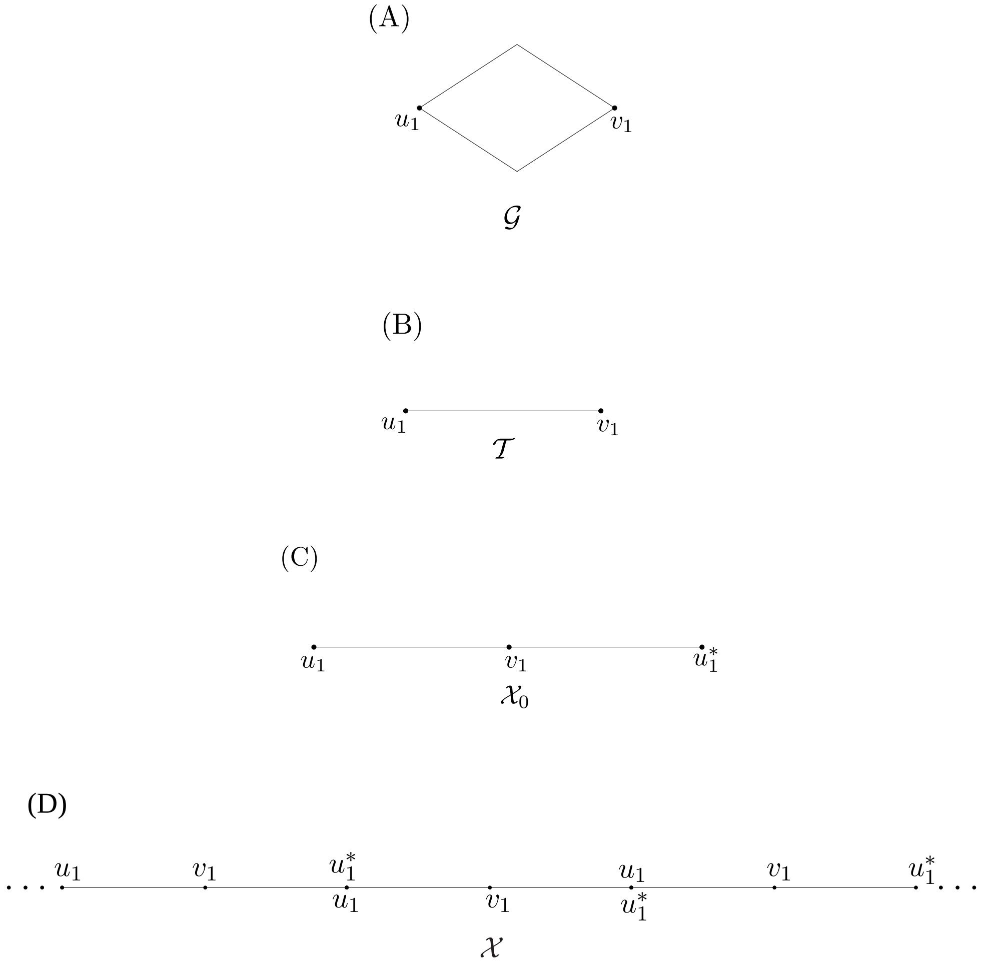


Figure 6.11: (A) 2-cycle roadmap $\mathcal{G}$. (B) an arbitrary spanning tree of $\mathcal{G}$. (C) the fundamental domain of the universal cover of $\mathcal{G}$. (D) the universal cover of $\mathcal{G}$.

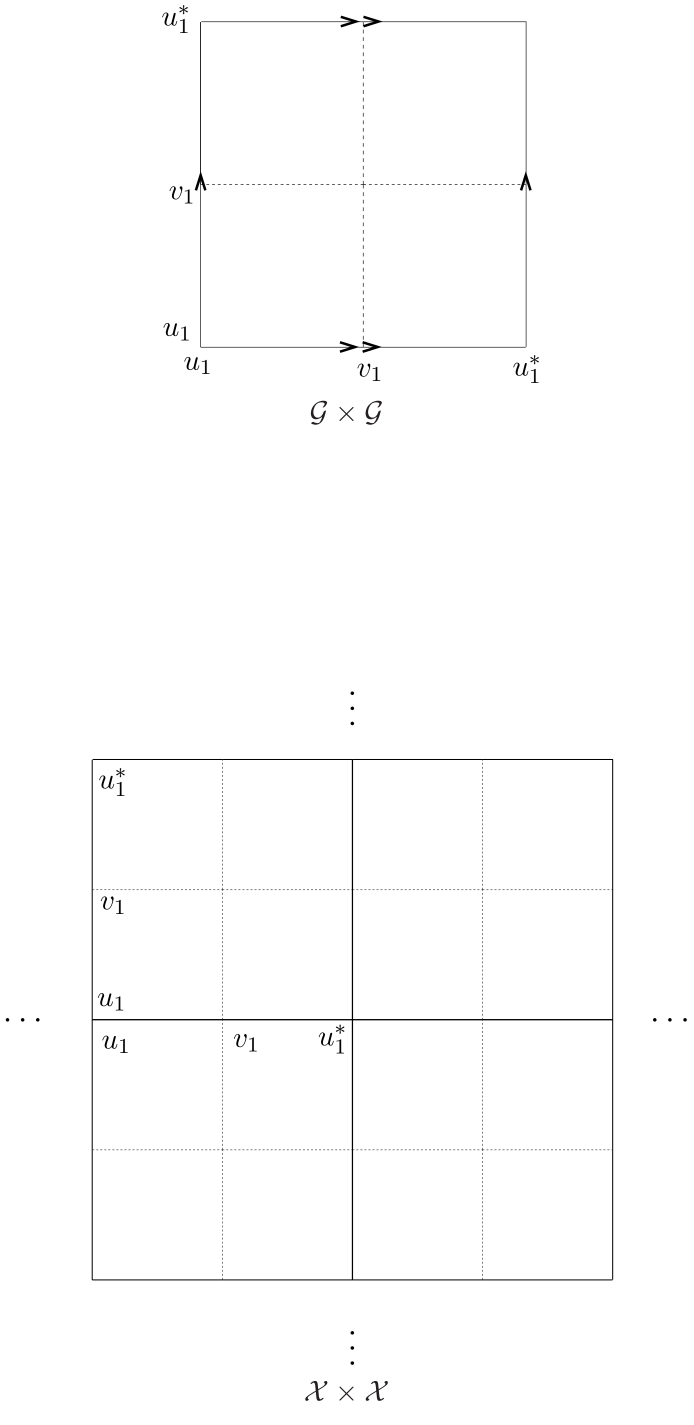


Figure 6.12: The coordination space of the 2-cycle roadmap, in Figure 6.11, and its universal cover. [up] The coordination space $\mathcal{G} \times \mathcal{G}$ which is a flat torus. [down] The universal cover of the coordination space.

to $\mathcal{X}_0$ such that $u_i \in \mathcal{X}_0$ is identified with u_i^* in one copy and $u_i^* \in \mathcal{X}_0$ is identified with u_i in another copy. See Figure 6.11 for a simple example.

Our algorithm builds $\mathcal{X}_0$ first, and initializes $\mathcal{Y} = \mathcal{X}_0$. It inserts u_i and u_i^* onto a list. For every vertex in the list, the algorithm generates a copy of $\mathcal{X}_0$ and glues it to $\mathcal{Y}$ along the relevant vertex. It then inserts the gluing spots of the newly generated copy in the list. It iterates over these steps until $\mathcal{Y}$ covers the relevant part of $\mathcal{X}$. For that purpose, the distance between the vertex and the initial copy of $\mathcal{X}_0$ is computed at each iteration. If that distance is more than the upper bound, then the vertex is neglected and no copies of $\mathcal{X}_0$ is glued. Eventually, the algorithm stops when there are no more vertices in the list.

6.5.3 Applying The Acyclic Algorithm

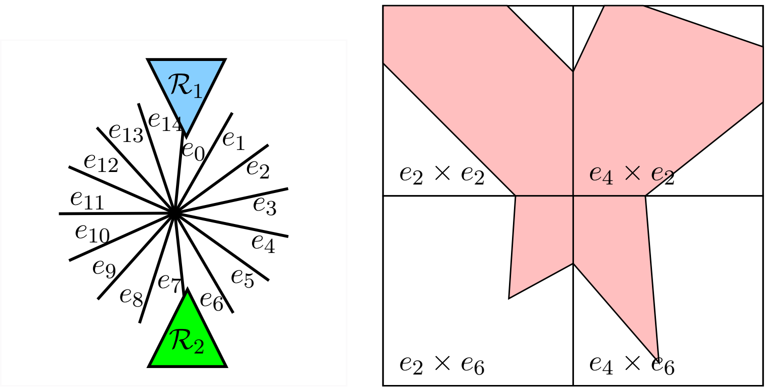
We showed how to compute $\mathcal{Y}$, the relevant portion of the universal cover of $\mathcal{G}$, in Section 6.5.2. Note that $\mathcal{Y} \subset \mathcal{X}$ is contractible. Therefore, it is acyclic and we may now apply our acyclic Pareto-optimal coordination algorithm to it. The acyclic algorithm computes the visibility graph in $\mathcal{Y} \times \mathcal{Y}$ among obstacle vertices and the initial and goal configurations, augments it with some extra edges, and finds the shortest paths. Obstacles are computed once in $\mathcal{G} \times \mathcal{G}$, and they are copied multiple times to obtain obstacles in $\mathcal{Y} \times \mathcal{Y}$. There are several copies of q^{goal} in $\mathcal{Y} \times \mathcal{Y}$ all of which need to be considered in the visibility graph. Any collision-free path from q^{init} to any q^{goal} copy is a coordination. Consequently, there are several copies of visibility graph points (x, q_2^{goal}) and (q_1^{goal}, x) that need to be considered.

6.5.4 Complexity Analysis

Let m denote the number of edges in $\mathcal{G}$ and let n denote total number of obstacle vertices in $\mathcal{G} \times \mathcal{G}$. Let ℓ be the total length of $\mathcal{G}$ and r the total length of obstacle boundary. Let b denote the length of the shortest edge in $\mathcal{G}$. Define $\alpha = 1 + \lceil (5\ell + r)/b \rceil$.

Theorem 6.11. *The time complexity of our cyclic algorithm is* $O(2^{5\alpha} m^{1+5\alpha} n^2 \log(m^{2\alpha} n))$.

Proof. We claim that the upper bound in Section 6.5.1 is not more than $5\ell + r$. Total length of Σ is at most $4\ell + r$. Since λ is the length of a path in Σ, $\lambda \leq 4\ell + r$. Also, δ_i are not more than the total length of the roadmap, so $\delta \leq \ell$. Therefore, $\lambda + \delta \leq 5\ell + r$.



(A) (B)

Figure 6.13: (A) A coordination problem on the star graph S_{16}. (B) A subset of $\mathcal{G} \times \mathcal{G}$ for this problem.

Every copy of the fundamental domain contributes at least b to the distance of a gluing spot from the initial copy of $\mathcal{X}_0$. Since $\mathcal{X}_0$ has no more than $2m$ gluing spots, at most $(2m)^\alpha$ copies of $\mathcal{X}_0$ are used in the construction of $\mathcal{Y}$. Therefore, $\mathcal{Y}$ has at most $2^\alpha m^{1+\alpha}$ edges. The number of obstacle vertices in $\mathcal{Y} \times \mathcal{Y}$ is at most $(2m)^{2\alpha} n$.

Theorem 6.8 proves that the complexity of the acyclic algorithm is $O(mn^2 \log n)$. Since the last step in this algorithm is the acyclic algorithm applied to $\mathcal{Y}$, which has at most $2^\alpha m^{1+\alpha}$ edges and $(2m)^{2\alpha} n$ obstacle vertices, the last step, which is the dominating step, takes

$O(2^{5\alpha} m^{1+5\alpha} n^2 \log(m^{2\alpha} n))$ time. ■

6.6 Examples

Figure 6.2 showed an example coordination problem on a connected roadmap with 7 edges. Each robot is shown in its initial state, and the goal is for the robots to switch places. For this problem $\mathcal{G} \times \mathcal{G}$ contains 31 obstacle polygons totalling 174 obstacle vertices. The complete set of 4 Pareto-optimal coordinations was illustrated in Figure 6.3.

As a second example, consider the star graph S_n with vertex set $\{v_0, \dots, v_{n-1}\}$ and edge set $\{(v_0, v_i) \; : \; 1 \leq i < n\}$. Coordination on this family of graphs is unusual because every cell of $\mathcal{G} \times \mathcal{G}$ has a non-empty obstacle region. In Figure 6.13, $\mathcal{R}_1$ and $\mathcal{R}_2$ navigate on an

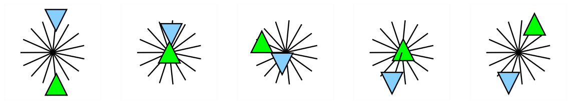
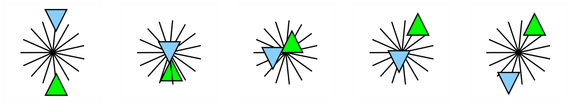
	$\mathcal{J}$
	(7.7,11.2)
	(10.6,7.6)

Figure 6.14: The two Pareto-optimal solutions for the problem in Figure 6.13.

embedding of S_{16}. The obstacle region has 225 obstacles with 933 vertices in total. The two Pareto-optimal solutions are shown in Figure 6.14.

6.7 Summary

We presented an algorithm to compute all Pareto-optimal coordinations of two polygonal robots on a network of piecewise-linear paths in the plane. The robots have a maximum speed and are capable of instantly switching to any speed bounded by the maximum speed. The key insight was an upper bound on the cost of each motion in a Pareto-optimal coordination. For an acyclic roadmap, our algorithm computes the generalized visibility graph, augments it with unit-slope edges, and computes the shortest paths. In the cyclic case, the algorithm applies the acyclic algorithm to a finite portion of the universal cover of the coordination space. We gave the time complexity of the algorithm in Sections 6.4.3 and 6.5.4. It remains open to find an optimal algorithm for this problem in terms of both geometric and combinatorial characteristics of the roadmap.

This method can be applied to find all Pareto-optimal coordinations, provided the configuration space of each robot is $\mathcal{G}$, all paths in $\mathcal{G} \times \mathcal{G}$ are allowed, and the obstacle regions in $\mathcal{G} \times \mathcal{G}$ are polygonal. In cases where the obstacle regions are not polygonal, but we can compute bitangents and the generalized visibility graph, our algorithm is slightly modified to compute all Pareto-optimal coordinations of such robots. For instance, one can consider computing bitangents of the obstacle region in $\mathcal{G} \times \mathcal{G}$ for car-like mobile robots on a network of $\mathcal{SA}$ paths [158]. The generalized visibility graph can then be computed and our algorithm be applied.

For three robots translating on a metric graph $\mathcal{G}$, the coordination space $\mathcal{G} \times \mathcal{G} \times \mathcal{G}$ is a three dimensional cube complex. Finding Pareto-optimal coordinations in this case reduces to finding shortest paths in $\mathcal{G} \times \mathcal{G} \times \mathcal{G}$. It is known that the shortest path problem in $\mathbb{R}^3$ is NP-hard [36]. An open question is whether special cylindrical structure of the obstacles in that case can be exploited to give efficient exact or approximation algorithms.

Chapter 7

Conclusion

We conclude this dissertation by a summary of our main results, open problems, and future directions.

7.1 Summary of Results

First, we characterized minimum wheel-rotation trajectories for the differential drive in the absence of obstacles in Chapter 3. In Section 2.2.1, we proved that minimum wheel-rotation trajectories for the differential drive exist. Therefore, it was viable to apply the Pontryagin Maximum Principle as a necessary condition [141]. By applying the Pontryagin Maximum Principle and developing geometric arguments, we derived necessary optimality conditions which helped to rule out non-optimal trajectories. The remaining trajectories form 28 different maximally optimal trajectories, which are listed in Table 3.1. Minimum wheel-rotation trajectories are composed of three motion primitives: rotation in place, straight line, and swing segments (one wheel stationary and the other rolling). A complete list of words that describe all of 52 minimum wheel-rotation trajectories was given in Table 3.2. We also proved that minimum wheel-rotation for the differential drive is equal to minimum time for the Reeds-Shepp car. Moreover, minimum wheel-rotation paths for the differential drive are exactly minimum time paths for the convexified Reeds-Shepp car. Based on the characterization of minimum wheel-rotation trajectories, a method to further determine the applicable trajectory for every pair of initial and goal configurations was presented in Section 3.7.

Second, we presented a method to compute minimum wheel-rotation trajectories among convex obstacles in Chapter 4. Among obstacles, a minimum wheel-rotation trajectory is composed of a finite number of segments, each of which is either in the interior of the free configuration space or on the boundary of obstacle region. Those segments that are in the interior of the free region were characterized in Chapter 3. A constrained Pontryagin Maximum Principle with our problem-specific arguments helped us to characterize boundary segments in Section 4.2. We used the Pontryagin Jump Condition [141] to rule out non-optimal intersections between free and boundary segments. Those results helped us to introduce a nonholonomic bitangency graph to which the search for the minimum wheel-rotation path is confined. In general, our nonholonomic bitangency graph is a 2-dimensional subset of the 3-dimensional configuration space of the robot. Therefore, a continuous search may be required to answer queries. In cases where the graph is 1-dimensional and can be computed, any graph search algorithm, such as Dijkstra's algorithm, is employed to extract the solution.

Third, we introduced Dubins airplane which extends the Dubins car with altitude. We gave a characterization of the time-optimal trajectories for the Dubins airplane in Chapter 5. For low and high final altitudes, the time-optimal trajectories respectively consist of the Dubins curve with unsaturated altitude velocity, and the Dubins curve followed by a helix with saturated altitude velocity. For medium altitudes in between low and high, different cases were recognized. The time-optimal trajectory is either a Dubins extremal (not the shortest) with unsaturated altitude velocity or a Dubins path of certain length with saturated altitude velocity. We gave a method to find a Dubins path with prescribed length if it exists. We also gave an analysis of locally longest curves for the Dubins car, i.e. those paths that may not be infinitesimally elongated. The control synthesis is computed numerically.

Fourth, in Chapter 6, we presented an algorithm to compute all Pareto-optimal coordinations of two polygonal robots on a network of piecewise-linear paths in the plane. The robots have a maximum speed and are capable of instantly switching to any speed bounded by the maximum speed. The problem was translated to a shortest path problem by fixing a canonical form, based on Euclidean shortest paths, for Pareto-optimal coordinations. For an acyclic roadmap, our algorithm computes the generalized visibility graph, augments it with unit-slope edges, and computes the shortest paths in $O(mn^2 \log n)$ time, in which m is the number of paths in the roadmap, and n is the number of coordination space vertices. In the cyclic case, our algorithm first computes an upper bound on the cost of each motion in a Pareto-optimal coordination. It then applies the acyclic algorithm to a finite portion of the universal cover of the coordination space. We gave the time complexity of the cyclic algorithm in Section 6.5.4. It remains open to find an optimal algorithm for this problem in terms of both geometric and combinatorial characteristics of the roadmap. This method can be applied to find all Pareto-optimal coordinations, provided the configuration space of each robot is the roadmap, all paths in the coordination space are allowed, and the obstacle regions in the coordination space are polygonal. In cases where the obstacle regions are not polygonal, but we can compute bitangents and the generalized visibility graph, our algorithm is slightly modified to compute all Pareto-optimal coordinations of such robots.

7.2 Open Problems

- In spite of partial attempts [49, 160, 163], characterizing time-optimal paths for a bounded velocity car with n trailers and a car with angular acceleration control still remain long-standing open problems.

- Minimum wheel-rotation for the differential drive is equal to minimum time for the Reeds-Shepp car. Moreover, minimum wheel-rotation paths for the differential drive are exactly minimum time paths for the convexified Reeds-Shepp car. We needed complicated optimal control tools to prove that fact which looks simple. It remains open to investigate the case and understand why the two problems yield equal solutions.

- In general, our nonholonomic bitangency graph is infinite. In that case, a continuous optimization is needed to compute minimum wheel-rotation trajectories among obstacles. Further investigation and an efficient algorithm to compute minimum wheel-rotation trajectories remain open.

- Analytical control synthesis for our Dubins airplane remains open.

- For three robots translating on a metric graph $\mathcal{G}$, the coordination space $\mathcal{G} \times \mathcal{G} \times \mathcal{G}$ is a three dimensional cube complex. Finding Pareto-optimal coordinations in this case reduces to finding shortest paths in $\mathcal{G} \times \mathcal{G} \times \mathcal{G}$. It is known that the shortest path problem in $\mathbb{R}^3$ is NP-hard [36]. An open question is whether special cylindrical structure of the obstacles in that case can be exploited to give efficient exact or approximation algorithms.

7.3 Future Directions

There are a number of interesting directions to pursue as future work:

- Investigating different motion primitives for nonholonomic robots is an important area of future work. Motion primitives are useful in general motion planning algorithms, because they provide a set of local plans. For example, rotation in place and straight motion is a sufficient set of primitives for the differential drive, but it does not give the minimum wheel-rotation plan. For various applications and with different planners, it is not clear which set of primitives yields a better motion plan in terms of the quality and computation effort.

- Solving optimal control problems has proven to be challenging, and characterization of the optimal solutions for every problem is a new piece of art. In practical applications, numerical methods are used to solve the problem. Since existing numerical methods have serious limitations, seeking an approximation framework in optimal control seems to be a necessary quest. By an approximation framework we mean a theory that can answer for example the following question: Given an error toleration bound ϵ, what are the necessary and sufficient conditions for a given trajectory to have a cost within ϵ distance of the optimal cost? How can we compute such trajectory? To the best of our knowledge, existing optimal control approximation schemes are mainly based on discretizing the solution. This is an approach mainly inspired by numerical methods. However, our question is more general: how can we analytically characterize approximately optimal trajectories?

- *Chattering* or infinite control switchings within a finite time interval, also called Fuller phenomenon, occurs in some practical problems such as time-optimal paths for a car with angular acceleration control [163]. For future investigation, one approach is to identify chattering pieces in an optimal trajectory and replace them with approximately optimal pieces. Another approach is to constrain the number of control discontinuities. One can also confine the class of controls to piecewise polynomial functions.

References

[1] P. K. Agarwal, T. Biedl, S. Lazard, S. Robbins, S. Suri, and S. Whitesides. Curvature-constrained shortest paths in a convex polygon. In *Proc. ACM Symposium on Computational Geometry*, pages 392–401, 1998.

[2] P. K. Agarwal, P. Raghavan, and H.Tamaki. Motion planning for a steering constrained robot through moderate obstacles. In *Proc. ACM Symposium on Computational Geometry*, pages 343–352, 1995.

[3] P. K. Agarwal and H. Wang. Approximation algorithms for curvature-constrained shortest paths. *SIAM J. Computing*, 30(6):1739–1772, 2001.

[4] H.-K. Ahn, O. Cheong, J. Matoušek, and A. Vigneron. Reachability by paths of bounded curvature in convex polygons. In *Proc. ACM Symposium on Computational Geometry*, pages 251–259, 2000.

[5] S. Akella and S. Hutchinson. Coordinating the motions of multiple robots with specified trajectories. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 624–631, 2002.

[6] J. V. Albro and L. E. Bobrow. Optimal motion primitives for a 5 dof experimental hopper. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 3630– 3635, 2001.

[7] H. Alt and M. Godau. Computing the Fréchet distance between two polygonal curves. *Int. J. Comput. Geometry Appl.*, 5:75–91, 1995.

[8] N. M. Amato, O. B. Bayazit, L. K. Dale, C. Jones, and D. Vallejo. OBPRM: An obstacle-based PRM for 3D workspaces. In *Proceedings of the Workshop on Algorithmic Foundations of Robotics*, pages 155–168, 1998.

[9] N.M. Amato, O.B. Bayazit, L.K. Dale, C. Jones, and D. Vallejo. Choosing good distance metrics and local planners for probabilistic roadmap methods. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 630–637, 1998.

[10] M. D. Ardema and J. M. Skowronski. Dynamic game applied to coordination control of two arm robotic system. In R. P. Hämäläinen and H. K. Ehtamo, editors, *Differential Games - Developments in Modelling and Computation*, pages 118–130. Springer-Verlag, Berlin, 1991.

[11] B. Aronov, M. de Berg, A.F. van der Stappen, P. Švestka, and J. Vleugels. Motion planning for multiple robots. *Discrete and Computational Geometry*, 22(4):505–525, 1999.

[12] Devin J. Balkcom, Paritosh A. Kavathekar, and Matthew T. Mason. The minimum-time trajectories for an omni-directional vehicle. In *Proc. Sixth International Workshop on the Algorithmic Foundations of Robotics*, 2006.

[13] Devin J. Balkcom and Matthew. T. Mason. Time optimal trajectories for bounded velocity differential drive vehicles. *Int. J. Robot. Res.*, 21(3):199–218, March 2002.

[14] J. Barraquand and J.-C. Latombe. A Monte-Carlo algorithm for path planning with many degrees of freedom. In *IEEE Int. Conf. Robot. & Autom.*, pages 1712–1717, 1990.

[15] J. Barraquand and J.-C. Latombe. Robot motion planning: A distributed representation approach. *Int. J. Robot. Res.*, 10(6):628–649, December 1991.

[16] S. Basu, R. Pollack, and M.-F. Roy. Computing roadmaps of semi-algebraic sets on a variety. *J. American Math. Soc.*, 13(1):55–82, 2000.

[17] S. Basu, R. Pollack, and M.-F. Roy. *Algorithms in Real Algebraic Geometry.* Springer, 2006.

[18] R. E. Bellman. *Dynamic Programming.* Princeton University Press, Princeton, NJ, 1957.

[19] C. Belta, A. Bicchi, M. Egerstedt, E. Frazzoli, E. Klavins, and G. J. Pappas. Symbolic planning and control of robot motion: State of the art and grand challenges. *IEEE Robotics and Automation Magazine*, 14(1):61–70, 2007.

[20] S. Bereg and D. Kirkpatrick. Curvature-bounded traversals of narrow corridors. In *Proc. ACM Symposium on Computational Geometry*, pages 278–287, 2005.

[21] D. P. Bertsekas. *Dynamic Programming: Deterministic and Stochastic Models.* Prentice-Hall, Englewood Cliffs, NJ, 1987.

[22] A. Bhatia and E. Frazzoli. Incremental search methods for reachability analysis of continuous and hybrid systems. In *Lecture Notes in Computer Science, 2993, Hybrid Systems: Computation and Control*, pages 67–78. Springer-Verlag, 2004.

[23] A.M. Bloch, P. Crouch, and T.S. Ratiu. Sub-riemannian optimal control problems. *Fields Inst. Comm.*, 3:35–48, 1994.

[24] R. Bohlin and L. Kavraki. Path planning using Lazy PRM. In *Proceedings IEEE International Conference on Robotics and Automation*, 2000.

[25] J.-D. Boissonnat, A. Cérézo, and J. Leblond. Shortest paths of bounded curvature in the plane. *J. Intelligent and Robotic Systems*, 11:5–20, 1994.

[26] J.-D. Boissonnat, S. Ghosh, T. Kavitha, and S. Lazard. An algorithm for computing a convex and simple path of bounded curvature in a simple polygon. *Algorithmica*, 34(2):109–156, 2002.

[27] J.-D. Boissonnat and S. Lazard. A polynomial-time algorithm for computing a shortest path of bounded curvature amidst moderate obstacles. In *Proc. ACM Symposium on Computational Geometry*, pages 242–251, 1996.

[28] V. Boor, N. H. Overmars, and A. F. van der Stappen. The gaussian sampling strategy for probabilistic roadmap planners. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 1018–1023, 1999.

[29] M. S. Branicky and M. M. Curtiss. Nonlinear and hybrid control via RRTs. In *Proc. Intl. Symp. on Math. Theory of Networks and Systems*, 2002.

[30] M. Bridson and A. Haefliger. *Metric Spaces of Nonpositive Curvature.* Springer-Verlag, Berlin, 1999.

[31] R. A. Brooks and T. Lozano-Pérez. A subdivision algorithm in configuration space for findpath with rotation. *IEEE Trans. Syst., Man, Cybern.*, 15(2):224–233, 1985.

[32] J. Bruce and M. Veloso. Real-time randomized path planning for robot navigation. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2002.

[33] A. E. Bryson and Y.-C. Ho. *Applied Optimal Control.* Hemisphere Publishing Corp., New York, NY, 1975.

[34] B. Burns and O. Brock. Sampling-based motion planning using predictive models. In *Proceedings IEEE International Conference on Robotics and Automation*, 2005.

[35] B. Burns and O. Brock. Toward optimal configuration space sampling. In *Proc. Robotics: Science and Systems*, 2005.

[36] J. Canny and J. Reif. New lower bound techniques for robot motion planning problems. In *Proc. IEEE Conf. on Foundations of Computer Science*, pages 49–60, 1987.

[37] J. F. Canny. *The Complexity of Robot Motion Planning.* MIT Press, Cambridge, MA, 1988.

[38] Lamberto Cesari. *Optimization Theory and Applications: problems with ordinary differetial equations.* Springer-Verlag, New York, NY, 1983.

[39] C. Chang, M. J. Chung, and B. H. Lee. Collision avoidance of two robot manipulators by minimum delay time. *IEEE Trans. Syst., Man, Cybern.*, 24(3):517–522, 1994.

[40] P.C. Chen and Y.K. Hwang. Sandros: A motion planner with performance proportional to task difficulty. In *Proc. of IEEE Int. Conf. Robotics and Automation*, pages 2346–2353, Nice, France, 1992.

[41] P. Cheng and S. M. LaValle. Resolution complete rapidly-exploring random trees. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 267–272, 2002.

[42] G. Chirikjian, A. Pamecha, and I. Ebert-Uphoff. Evaluating efficiency of self-reconfiguration in a class of modular robots. *Journal of Robotic Systems*, 13(5):717–338, 1996.

[43] H. Chitsaz and S.M. LaValle. On time-optimal paths for the dubins airplane. In *IEEE Conf. Decision & Control*, to be submitted, 2007.

[44] H. Chitsaz, S.M. LaValle, D.J. Balkcom, and M.T. Mason. An explicit characterization of minimum wheel-rotation paths for differential-drives. In *Proceedings 12th IEEE International Conference on Methods and Models in Automation and Robotics*, 2006.

[45] H. Chitsaz, S.M. LaValle, D.J. Balkcom, and M.T. Mason. Minimum wheel-rotation paths for differential-drive mobile robots. In *Proceedings IEEE International Conference on Robotics and Automation*, 2006.

[46] H. Chitsaz, J.M. O'Kane, and S.M. LaValle. Exact pareto-optimal coordination of two translating polygonal robots on an acyclic roadmap. In *Proceedings IEEE International Conference on Robotics and Automation*, 2004.

[47] Joonsoo Choi, Jürgen Sellen, and Chee-Keng Yap. Approximate euclidean shortest path in 3-space. In *Annual ACM Symposium on Computational Geometry*, pages 41–48, New York, NY, USA, 1994. ACM.

[48] P. Choudhury and K. Lynch. Trajectory planning for second-order underactuated mechanical systems in presence of obstacles. In *Proc. International Workshop on the Algorithmic Foundations of Robotics*, 2002.

[49] M. Chyba and S. Sekhavat. Time optimal paths for a mobile robot with one trailer. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, volume 3, pages 1669–1674, 1999.

[50] G. E. Collins. *Lecture Notes in Computer Science, 33.* Springer-Verlag, Berlin, 1975.

[51] D. C. Conner, H. Kress-Gazit, H. Choset, A. Rizzi, and G. J. Pappas. Valet parking without a valet. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2007.

[52] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry: Algorithms and Applications.* Springer, Berlin, 2000.

[53] G. Desaulniers. On shortest paths for a car-like robot maneuvering around obstacles. *Robotics and Autonomous Systems*, 17:139–148(10), May 1996.

[54] G. Desaulniers, F. Soumis, and J.-C. Laurent. A shortest path algorithm for a car-like robot in a polygonal environment. *Int. J. Robot. Res.*, 17(5):512–530, 1998.

[55] B. R. Donald. A search algorithm for motion planning with six degrees of freedom. *Artif. Intell.*, 31:295–353, 1987.

[56] 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*, 79:497–516, 1957.

[57] L. E. Dubins. On plane curves with curvature. *Pacific J. Math.*, 11(2):471481, 1961.

[58] H. Edelsbrunner. *Algorithms in Combinatorial Geometry.* Springer-Verlag, Berlin, 1987.

[59] I. Emiris. *Sparse Elimination and Applications in Kinematics.* PhD thesis, UC Berkeley, Berkeley, CA, 1994.

[60] M. Erdmann and T. Lozano-Perez. On multiple moving objects. In *IEEE Int. Conf. Robot. & Autom.*, pages 1419–1424, 1986.

[61] D. Ferguson and A. Stentz. Replanning with RRTs. In *Proceedings IEEE International Conference on Robotics and Automation*, 2006.

[62] S. Fortune and G. Wilfong. Planning constrained motion. *Annals of Mathematics and AI*, 3(1):21–82, 1991.

[63] E. Frazzoli, M. A. Dahleh, and E. Feron. Real-time motion planning for agile autonomous vehicles. *AIAA Journal of Guidance and Control*, 25(1):116–129, 2002.

[64] E. Frazzoli, M. A. Dahleh, and E. Feron. Maneuver-based motion planning for nonlinear systems with symmetries. *IEEE Trans. on Robotics*, 21(6):1077–1091, 2005.

[65] I. M. Gelfand and S. V. Fomin. *Calculus of variations.* Dover Publications, 2000.

[66] S. K. Ghosh and D. M. Mount. An output-sensitive algorithm for computing visibility graphs. *SIAM J. Computing*, 20:888–910, 1991.

[67] R. Ghrist. Shape complexes for metamorphic robot systems. In *Proc. Workshop on the Algorithmic Foundations of Robotics*, December 2002.

[68] R. Ghrist and S. M. LaValle. Nonpositive curvature and pareto-optimal coordination of robots. *SIAM Journal of Control and Optimization*, 45:1697–1713, 2006.

[69] R. Ghrist, J. M. O'Kane, and S. M. LaValle. Computing Pareto Optimal Coordinations on Roadmaps. *The International Journal of Robotics Research*, 24(11):997–1010, 2005.

[70] D. Grigoriev and A. Slissenko. Polytime algorithm for the shortest path in a homotopy class amidst semi-algebraic obstacles in the plane. In *ISSAC '98: Proceedings of the 1998 international symposium on Symbolic and algebraic computation*, pages 17–24. ACM Press, 1998.

[71] M. Gromov. Hyperbolic groups. In S.M. Gersten, editor, *Essays in Group Theory*, pages 75–264. Springer-Verlag, 1987.

[72] L. J. Guibas and J. Hershberger. Optimal shortest path queries in a simple polygon. *Journal of Computer and System Sciences*, 39(2):126–152, 1989.

[73] K. Hauser, T. Bretl, K. Harada, and J. C. Latombe. Using motion primitives in probabilistic sample-based planning for humanoid robots. In *Workshop on the Algorithmic Foundations of Robotics (WAFR)*, 2006.

[74] T. Hebert. *Navigation of an autonomous vehicle using a combined electrostatic potential field/fuzzy inference approach.* PhD thesis, Univ. Southwestern Louisiana, 1998.

[75] J. Heintz, T. Krick, A. Slissenko, and P. Solernó. Search for shortest path around semialgebraic obstacles in the plane. *J. Math Sciences*, 70(4):1944–1949, 1994.

[76] G. Heinzinger, P. Jacobs, J. Canny, and B. Paden. Time-optimal trajectories for a robotic manipulator: A provably good approximation algorithm. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 150–155, Cincinnati, OH, 1990.

[77] J. Hershberger. A new data structure for shortest path queries in a simple polygon. *Information Processing Letters*, 38(5):225–281, 1991.

[78] J. Hershberger and J. Snoeyink. Computing minimum length paths of a given homotopy class. *Computational geometry*, 4:63–97, 1994.

[79] J. Hershberger and S. Suri. An optimal algorithm for Euclidean shortest paths in the plane. *SIAM J. Computing*, 28:2215–2256, 1999.

[80] D. Hsu, J.-C. Latombe, and R. Motwani. Path planning in expansive configuration spaces. *Int. J. Comput. Geom. & Appl.*, 4:495–512, 1999.

[81] H. Hu, M. Brady, and P. Probert. Coping with uncertainty in control and planning for a mobile robot. In *IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, pages 1025–1030, Osaka, Japan, November 1991.

[82] P. Jacobs and J. Canny. Planning smooth paths for mobile robots. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 2–7, 1989.

[83] L. Jaillet, A. Yershova, S. M. LaValle, and T. Simeon. Adaptive tuning of the sampling domain for dynamic-domain RRTs. In *Proceedings IEEE International Conference on Intelligent Robots and Systems*, 2005.

[84] O. Junge, J. Marsden, and S. Ober-Blobaum. Discrete mechanics and optimal control. In *Proceedings of the 16th IFAC World Congress*, 2005.

[85] V. Jurdjevic. Optimal control problems on Lie groups: Crossroads between geometry and mechanics. In B. Jakubczyk and W. Respondek, editors, *Geometry of Feedback and Optimal Control*. Marcel-Decker, 1992.

[86] S. Kagami, J. Kuffner, K. Nishiwaki, K. Okada, and M. Inaba. Humanoid arm motion planning using stereo vision and RRT search. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2003.

[87] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Trans. Robot. & Autom.*, 12(4):566–580, June 1996.

[88] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *Int. J. Robot. Res.*, 5(1):90–98, 1986.

[89] J. Kim and J. P. Ostrowski. Motion planning of aerial robot using rapidly-exploring random trees with dynamic constraints. In *Proceedings IEEE International Conference on Robotics and Automation*, 2003.

[90] Y. Koga, K. Kondo, J. Kuffner, and J.-C. Latombe. Planning motions with intentions. *Computer Graphics (SIGGRAPH'94)*, pages 395–408, 1994.

[91] K. Kondo. Motion planning with six degrees of freedom by multistrategic bidirectional heuristic free-space enumeration. *IEEE Transactions on Robotics and Automation*, 7(3):267–277, 1991.

[92] W.-S. Koon and J.E. Marsden. Optimal control for holonomic and nonholonomic mechanical systems with symmetry and lagrangian reduction. *SIAM J. Control & Optimization*, 35(3):901–929, 1997.

[93] K. Kotay, D. Rus, M. Vora, and C. McGray. The self-reconfiguring robotic molecule: Design and control algorithms. In P.K. Agarwal, L. Kavraki, and M. Mason, editors, *Robotics: The Algorithmic Perspective.* AK Peters, Natick, MA, 1998.

[94] B. H. Krogh. A generalized potential field approach to obstacle avoidance control. In *Proceedings of SME Conference on Robotics Research*, August 1984.

[95] J. Krozel, T. Mueller, and G. Hunter. Free flight conflict detection and resolution analysis. In *AIAA Guidance, Navigation and Control Conference AIAA-1996-3763*, 1996.

[96] H. Kurniawati and D. Hsu. Workspace-based connectivity oracle: an adaptive sampling strategy for prm planning. In *Proc. International Workshop on the Algorithmic Foundations of Robotics*, 2006.

[97] A. Ladd and L. Kavraki. Fast exploration for robots with dynamics. In *Proc. International Workshop on the Algorithmic Foundations of Robotics*, 2004.

[98] J.-C. Latombe. A fast path planner for a car-like indoor mobile robot. In *Proc. Am. Assoc. Artif. Intell.*, pages 659–665, 1991.

[99] J.-C. Latombe. *Robot Motion Planning.* Kluwer Academic Publishers, Boston, MA, 1991.

[100] J.-P. Laumond. Finding collision-free smooth trajectories for a non-holonomic mobile robot. In *Proc. Int. Joint Conf. on Artif. Intell.*, pages 1120–1123, 1987.

[101] J.-P. Laumond, editor. *Robot Motion Planning and Control.* Springer, 1998.

[102] S. M. LaValle. Rapidly-exploring random trees: A new tool for path planning. TR 98-11, Computer Science Dept., Iowa State University, Oct. 1998.

[103] S. M. LaValle. *Planning Algorithms.* Cambridge University Press, Cambridge, U.K., 2006. Also available at http://planning.cs.uiuc.edu/.

[104] S. M. LaValle and S. A. Hutchinson. Optimal motion planning for multiple robots having independent goals. *IEEE Trans. on Robotics and Automation*, 14(6):912–925, December 1998.

[105] S. M. LaValle and J. J. Kuffner. Rapidly-exploring random trees: Progress and prospects. In B. R. Donald, K. M. Lynch, and D. Rus, editors, *Algorithmic and Computational Robotics: New Directions*, pages 293–308. A K Peters, Wellesley, MA, 2001.

[106] Der-Tsai Lee. *Proximity and reachability in the plane.* PhD thesis, University of Illinois, Urbana, IL, 1978. Report R-831, Doc no. 1978-2224.

[107] J. Lengyel, M. Reichert, B. R. Donald, and D. P. Greenberg. Real-time robot motion planning using rasterizing computer graphics hardware. *Computer Graphics*, 24(4):327–335, 1990.

[108] P. Leven and S. Hutchinson. Real-time motion planning in changing environments. In *Proc. International Symposium on Robotics Research*, 2000.

[109] T.-Y. Li and Y.-C. Shie. An incremental learning approach to motion planning with roadmap management. In *Proceedings IEEE International Conference on Robotics and Automation*, 2002.

[110] S. R. Lindemann and S. M. LaValle. Incrementally reducing dispersion by increasing Voronoi bias in RRTs. In *Proceedings IEEE International Conference on Robotics and Automation*, 2004.

[111] S. R. Lindemann and S. M. LaValle. Steps toward derandomizing RRTs. In *IEEE Fourth International Workshop on Robot Motion and Control*, 2004.

[112] F. Lingelbach. Path planning using probabilistic cell decomposition. In *Proceedings IEEE International Conference on Robotics and Automation*, 2004.

[113] S. G. Loizou, D. V. Dimarogonas, and K. J. Kyriakopolous. Decentralized feedback stabilization of multiple nonholonomic agents. In *Proceedings IEEE International Conference on Robotics and Automation*, 2004.

[114] S. G. Loizou and K. J. Kyriakopolous. Closed loop navigation for multiple nonholonomic vehicles. In *Proceedings IEEE International Conference on Robotics and Automation*, 2003.

[115] C. López and E. Martínez. Sub-finslerian metric associated to an optimal control system. *SIAM J. Control & Optimization*, 39(3):798–811, 2000.

[116] T. Lozano-Pérez. Spatial planning: A configuration space approach. *IEEE Trans. on Comput.*, C-32(2):108–120, 1983.

[117] T. Lozano-Pérez and M. A. Wesley. An algorithm for planning collision-free paths among polyhedral obstacles. *Communications of the ACM*, 22(10):560–570, 1979.

[118] David G. Luenberger. *Optimization by Vector Space Methods.* Wiley-Interscience, 1997.

[119] David Lutterkort and Jörg Peters. Smooth paths in a polygonal channel. In *Annual ACM Symposium on Computational Geometry.* ACM, 1999.

[120] J. Marsden and M. West. Discrete mechanics and variational integrators. *Acta Num.*, 10:357–514, 2001.

[121] E. Mazer, J. M. Ahuactzin, and P. Bessière. The Ariadne's clew algorithm. *J. Artificial Intell. Res.*, 9:295–316, November 1998.

[122] T. Mehta and M. Egerstedt. An optimal control approach to mode generation in hybrid systems. *Nonlinear Analysis: Theory, Methods and Applications*, 65(5):963–983, 2006.

[123] J. S. B. Mitchell. *Planning Shortest Paths.* PhD thesis, Stanford University, 1986.

[124] J. S. B. Mitchell. Shortest paths among obstacles in the plane. *Int. J. Comput. Geom. & Appl.*, 6(3):309–332, 1996.

[125] J. S. B. Mitchell. Shortest paths and networks. In J. E. Goodman and J. O'Rourke, editors, *Handbook of Discrete and Computational Geometry, 2nd Ed.*, pages 607–641. Chapman and Hall/CRC Press, New York, 2004.

[126] J. Miura and Y. Shirai. Planning of vision and motion for a mobile robot using a probabilistic model of uncertainty. In *IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, pages 403–408, Osaka, Japan, May 1991.

[127] R. Montgomery. *A tour of subriemannian geometries, their geodesics and applications*, volume 91 of *Math. Surveys and Monographs.* AMS, 2002.

[128] P. Moutarlier, B. Mirtich, and J. Canny. Shortest paths for a car-like robot to manifolds in configuration space. *Int. J. Robot. Res.*, 15(1), 1996.

[129] N. J. Nilsson. A mobile automaton: An application of artificial intelligence techniques. In *1st International Joint Conference on Artificial Intelligence*, pages 509–520, 1969.

[130] P. A. O'Donnell and T. Lozano-Pérez. Deadlock-free and collision-free coordination of two robot manipulators. In *IEEE Int. Conf. Robot. & Autom.*, pages 484–489, 1989.

[131] C. Ó'Dúnlaing and C. K. Yap. A retraction method for planning the motion of a disc. *Journal of Algorithms*, 6:104–111, 1985.

[132] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces.* Springer, 2003.

[133] C. H. Papadimitriou. An algorithm for shortest-path planning in three dimensions. *Information Processing Letters*, 20(5):259–263, 1985.

[134] W. Pasillas-Lépine and W. Respondek. Conversion of the kinematics of the n-trailer system into Kumpera-Ruiz normal form and motion planning through the singular locus. In *IEEE Conf. Decision & Control*, pages 2914–2919, 1999.

[135] W. Pasillas-Lépine and W. Respondek. On the geometry of Goursat structures. *ArXiv Mathematics e-prints*, math/9911101, November 1999.

[136] J. Peng and S. Akella. Coordinating multiple robots with kinodynamic constraints along specified paths. In *Proc. Fifth International Workshop on the Algorithmic Foundations of Robotics*, 2002.

[137] J. Peng and S. Akella. Coordinating multiple double integrator robots on a roadmap: Convexity and global optimality. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 2762–2769, Barcelona, Spain, April 2005.

[138] J. Peng and S. Akella. Coordinating multiple robots with kinodynamic constraints along specified paths. *Int. J. Robot. Res.*, 24(4):295–310, April 2005.

[139] C. Pisula, K. Hoff, M. Lin, and D. Manoch. Randomized path planning for a rigid body based on hardware accelerated Voronoi sampling. In *Proc. Workshop on Algorithmic Foundation of Robotics*, 2000.

[140] E. Plaku, K. E. Bekris, B. Y. Chen, A. M. Ladd, and L. E. Kavraki. Sampling-based roadmap of trees for parallel motion planning. *IEEE Transactions on Robotics and Automation*, 21(4):597–608, 2005.

[141] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko. *The Mathematical Theory of Optimal Processes*. John Wiley, 1962.

[142] J. A. Reeds and L. A. Shepp. Optimal paths for a car that goes both forwards and backwards. *Pacific J. Math.*, 145(2):367–393, 1990.

[143] J. Reif and H. Wang. The complexity of the two dimensional curvature-constrained shortest-path problem. In *Proc. Third International Workshop on the Algorithmic Foundations of Robotics*, pages 49–57, 1998.

[144] J. H. Reif. Complexity of the mover's problem and generalizations. In *Proc. of IEEE Symp. on Foundat. of Comp. Sci.*, pages 421–427, 1979.

[145] David B. Reister and Francois G. Pin. Time-optimal trajectories for mobile robots with two independently driven wheels. *International Journal of Robotics Research*, 13(1):38–54, February 1994.

[146] M. Renaud and J.Y. Fourquet. Minimum time motion of a mobile robot with two independent acceleration-driven wheels. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 2608–2613, 1997.

[147] E. Rimon and D. E. Koditschek. Exact robot navigation using artificial potential fields. *IEEE Trans. Robot. & Autom.*, 8(5):501–518, October 1992.

[148] S. Rodriguez, X. Tang, J.-M. Lien, and N. M. Amato. Obrrt: an obstacle-based rapidly-exploring random tree. In *Proceedings IEEE International Conference on Robotics and Automation*, 2006.

[149] P. Rouchon, M. Fliess, J. Levine, and P. Martin. Flatness, motion planning and trailer systems. In *IEEE Conf. Decision & Control*, 1993.

[150] A. Sarti, G. C. Walsh, and S. Sastry. Steering left-invariant control systems on matrix Lie groups. In *IEEE Conf. Decision & Control*, pages 3117–3121, 1993.

[151] A.V. Sarychev and H. Nijmeijer. Extremal controls for chained systems. *Journal of Dynamical and Control Systems*, 2(4):503–527, 1996.

[152] Y. Sawaragi, H. Nakayama, and T. Tanino. *Theory of Multiobjective Optimization*. Academic Press, New York, NY, 1985.

[153] J. T. Schwartz and M. Sharir. On the piano movers' problem: I. The case of a two-dimensional rigid polygonal body moving amidst polygonal barriers. *Communications on Pure and Applied Mathematics*, 36:345–398, 1983.

[154] J. T. Schwartz and M. Sharir. On the piano movers' problem: II. General techniqies for computing topological properties of algebraic manifolds. *Communications on Pure and Applied Mathematics*, 36:345–398, 1983.

[155] J. T. Schwartz and M. Sharir. On the piano movers' problem: III. Coordinating the motion of several independent bodies. *Int. J. Robot. Res.*, 2(3):97–140, 1983.

[156] J.A. Sethian. *Level Set Methods and Fast Marching Methods.* Cambridge University Press, 1999.

[157] T. Simeon, J.-P. Laumond., and C. Nissoux. Visibility based probabilistic roadmaps for motion planning. *Advanced Robotics Journal*, 14(6), 2000.

[158] T. Simeon, S. Leroy, and J.-P. Laumond. Path coordination for multiple mobile robots: a resolution complete algorithm. *IEEE Trans. Robot. & Autom.*, 18(1), February 2002.

[159] O. Sørdalen. Conversion of the kinematics of a car with n trailers into chained form. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 382–387, 1993.

[160] P. Souères and J.-D. Boissonnat. Optimal trajectories for nonholonomic mobile robots. In J.-P. Laumond, editor, *Robot Motion Planning and Control*, pages 93–170. Springer, 1998.

[161] P. Souères and J. P. Laumond. Shortest paths synthesis for a car-like robot. In *IEEE Transactions on Automatic Control*, pages 672–688, 1996.

[162] S.-H. Suh and K. G. Shin. A variational dynamic programming approach to robot-path planning with a distance-safety criterion. *IEEE Trans. Robot. & Autom.*, 4(3):334–349, June 1988.

[163] Héctor Sussmann. The markov-dubins problem with angular acceleration control. In *Proccedings of the 36th IEEE Conference on Decision and Control, San Diego, CA*, pages 2639–2643. IEEE Publications, 1997.

[164] Héctor Sussmann and Guoqing Tang. Shortest paths for the Reeds-Shepp car: A worked out example of the use of geometric techniques in nonlinear optimal control. Technical Report SYNCON 91-10, Dept. of Mathematics, Rutgers University, 1991.

[165] P. Svestka and M. H. Overmars. Coordinated motion planning for multiple car-like robots using probabilistic roadmaps. In *IEEE Int. Conf. Robot. & Autom.*, pages 1631–1636, 1995.

[166] H. G. Tanner, S. G. Loizou, and K. J. Kyriakopolous. Nonholonomic navigation and control of cooperating mobile manipulators. *IEEE Transactions on Robotics and Automation*, 19(1):53–64, 2003.

[167] S. Thomas, G. Song, and N. Amato. Protein folding by motion planning. *Physical Biology*, 2:148–155, 2005.

[168] C. Tomlin, G. J. Pappas, and S. Sastry. Conflict resolution for air traffic management: A study in multiagent hybrid systems. *IEEE Trans. on Automatic Control*, 43(4), 1998.

[169] C. Urmson and R. Simmons. Approaches for heuristically biasing RRT growth. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2003.

[170] M. Vendittelli, J.P. Laumond, and C. Nissoux. Obstacle distance for car-like robots. *IEEE Transactions on Robotics and Automation*, 15(4):678–691, 1999.

[171] M. Vendittelli, J.P. Laumond, and P. Souères. Shortest paths to obstacles for a polygonal car-like robot. In *IEEE Conf. Decision & Control*, 1999.

[172] R. Volpe and P. Khosla. Manipulator control with superquadratic artificial potential functions: Theory and experiments. *IEEE Transactions on Systems, Man, and Cybernetics*, 20(6):1423–1436, 1990.

[173] G. Walsh, A. Sarti, and S. Shankar Sastry. Algorithms for steering on the group of rotations. Technical Report UCB/ERL M93/44, EECS Department, University of California, Berkeley, 1993.

[174] G. C. Walsh, R. Montgomery, and S. Sastry. Optimal path planning on matrix Lie groups. In *IEEE Conf. Decision & Control*, volume 2, pages 1258–1263, 1994.

[175] S. A. Wilmarth, N. M. Amato, and P. F. Stiller. MAPRM: A probabilistic roadmap planner with sampling on the medial axis of the free space. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 1024–1031, 1999.

[176] H. Wong, V. Kapila, and R. Vaidyanathan. UAV optimal path planning using c-c-c class paths for target touring. In *IEEE Conf. Decision & Control*, pages 1105–1110, 2004.

[177] J. Yakey, S. M. LaValle, and L. E. Kavraki. Randomized path planning for linkages with closed kinematic chains. *IEEE Transactions on Robotics and Automation*, 17(6):951–958, December 2001.

[178] G. Yang and V. Kapila. Optimal path planning for unmanned air vehicles with kinematic and tactical constraints. In *IEEE Conf. Decision & Control*, pages 1301–1306, 2002.

[179] A. Yershova, L. Jaillet, T. Simeon, and S. M. LaValle. Dynamic-domain RRTs: Efficient exploration by controlling the sampling domain. In *Proceedings IEEE International Conference on Robotics and Automation*, 2005.

[180] M. Yim. *Locomotion with a Unit-Modular Reconfigurable Robot.* PhD thesis, Stanford Univ., December 1994. Stanford Technical Report STAN-CS-94-1536.

[181] M.I. Zelikin and V.F. Borisov. *Theory of Chattering Control.* Birkhäuser, Boston, NJ, 1994.

[182] Y. Zhao and R. L. Schultz. Deterministic resolution of two aircraft conflict in free flight. In *AIAA Guidance, Navigation, and Control Conference AIAA-1997-3547*, 1997.

Author's Biography

Dr. Hamid Reza Chitsaz was born in Tehran, Iran on September 20, 1978. From 1989 to 1996, he attended Allameh Helli school in Tehran, which is affiliated with the National Organization for Development of Exceptional Talents (NODET). Dr. Chitsaz graduated from Allameh Helli high school and entered Sharif University of Technology, Tehran, Iran in 1996. He received the Bachelors degree in computer engineering and the Bachelors degree in pure mathematics both from Sharif University of Technology in 2001. Dr. Chitsaz then moved to Champaign, Illinois to pursue graduate study in computer science - robotics. He earned a Master of Science degree in mathematics from University of Illinois-Urbana Champaign in 2006. Following the completion of his Ph.D., Dr. Chitsaz will begin work for the Department of Computer Science of Simon Fraser University, Canada, as a postdoctoral fellow doing research on computational biology and bioinformatics.