Ph.D. DISSERTATION

A Velocity-Based Local Navigation Approach to Collision Avoidance of Elliptic Robots

타원 로봇의 충돌 회피를 위한 속도 기반의 지역 경로 계획 방법

BY

JAE-DO JEON

FEBRUARY 2017

DEPARTMENT OF ELECTRICAL ENGINEERING AND
COMPUTER SCIENCE
COLLEGE OF ENGINEERING
SEOUL NATIONAL UNIVERSITY

Abstract

Collision-free motion planning has been hierarchically decomposed into two parts: global and local planners. While the former generates the shortest path to the goal from global environmental information, the latter modifies the path from the global one by considering unexpected dynamic obstacles and motion constraints of mobile robots. In the local navigation problem, robots and obstacles have been approximated by simple geometric objects in order to decrease the computation time. They have been generally enclosed by circles due to its simplicity in collision detection. However, this approximation becomes overly conservative if the objects are elongated, which leads the robots to travel longer paths than necessary to avoid collisions.

This dissertation presents a velocity-based approach to address the local navigation problem of anisotropic mobile robots bounded by ellipses. Compared with the other geometries, Löwner ellipse, the minimum area bounding ellipse, provides more compact representation for robots and obstacles in a 2D plane, but the collision detection between them is more complicated. Hence, it is first investigated under what conditions a collision between two ellipses occurs. To this end, the configuration space framework and an algebraic approach are introduced. In the former method, it is found that an elliptic robot can be regarded as a circular robot with radius equal to its minor radius by adequately controlling its orientation. In the latter method, the interior-disjoint condition between two ellipses is characterized by four inequalities.

Next, a velocity-based approach is suggested on the basis of the collision detection so that an elliptic robot moves to its goal without collisions with obstacles. The proposed algorithm is decomposed into two phases: linear and angular motion planning. In the first phase, the ellipse-based velocity obstacle (EBVO) is defined as the set of linear velocities of a robot that would cause a collision within a finite time horizon. Furthermore, strategies for determining a new linear velocity with the EBVO are explained. In the second phase, the angular velocity is selected with which the robot can circumvent the obstacle blocking the path to the goal with the minimum deviation.

Finally, the obstacle avoidance method was extended for multi-robot collision avoidance on the basis on the concept of reciprocity. The concept of hybrid reciprocal velocity obstacles is adopted in the part of linear motion planning, and the collision-free reciprocal rotation angles are calculated in the part of angular motion planning on the assumption that if one robot rotates, then the other robot may rotate equally or equally opposite.

The proposed algorithm was validated in simulations for various scenarios in terms of travel time and distance. It was shown that it outperformed the methods that enclosed robots and obstacles by circles, by ellipses without rotation, and by polygons with rotation. In addition, it was shown that the computation time of the proposed method was much smaller than the sampling time, which means that it is fast enough for real-time applications.

Keywords: local motion planning, ellipse, velocity obstacles, reciprocal collision avoidance, multi-robot systems

Student Number: 2011-20924

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

1.1 Background of the Problem

Collision-free motion planning has been one of the fundamental problems in robotics, which is hierarchically decomposed into two parts: high-level and low-level planners [7]. The higher level planner generates the shortest path to the goal from global environmental information, whereas the local level planner modifies the path from the higher level one by considering unexpected dynamic obstacles and motion constraints of mobile robots. This dissertation deals with the latter, where a robot selects its new linear and angular velocities in each sense-plan-act cycle.

There have been three representative approaches to solve the local navigation problem: cellular automata, social forces, and velocity-based method. First of all, cellular automata are discrete-time dynamical models, where the workspace of robots is discretized into a cell grid. In this model, a robot must occupy at most one cell, and a cell have to contain no more than a single robot in order to prevent collisions [82]. Approaches based on the cellular automata have been employed to analyze the traffic flow [29], to reproduce the movement of pedestrians [14, 15, 16], and to plan the motion of multiple robots in a decentralized way [57, 107]. Nevertheless, these approaches exhibited unrealistic solution because movements of robots were limited due to the discretization, as mentioned in [67, 126].

Next, the concept of social forces was first introduced in [55] to describe individual pedestrian behavior. Since social forces were virtual forces that represented social interaction in nature, people were motivated to move as if the forces were directly exerted on their body. For instance, a pedestrian felt a driving force toward its goal and repulsive forces from other pedestrians and obstacles, accelerated by the resultant force according to Newton's second law. To comprehend the behavior of pedestrians more exactly, compression and friction forces model [54] and centrifugal force model [26, 127] were added. However, this model ran into some problems when applied to planning collision-free motions for robots. One was that mobile robots using a social force model suffered a near miss in even uncrowded situations. Notwithstanding robots preferred to adjust their paths to a minimum to avert collisions, they did not take any actions to avoid collisions until they got very close [58, 79]. Another issue was the overlapping-oscillations duality. In social forces model, if the strength of the repulsive forces was reduced, robots were overlapped with one another. Otherwise, they performed an oscillating movement and were forced to move the opposite direction from their goal. These problems were difficult to solve without compromising the simplicity of the model, as referred to in [25].

In contrast, velocity-based approaches have not had problems associated with the confinement of robots' motion and the overlapping between robots. This approach has utilized the robot's velocity in order to determine potential collisions on the basis of the concept of the velocity obstacle (VO) [41], which maps the information of the workspace to the velocity space. The VO is defined as a set of velocities of the robot that would induce a collision within a finite time horizon, so that the robot can avoid other robots and obstacles by selecting its new velocity outside of the VO.

However, robots and obstacles are approximated by simple geometric objects in the velocity-based approach since the contours of them are quite complicated in general. There are three criteria in choosing a bounding box: tightness, computational ease, and complexity of describing the shape. The first criterion is that the area of the bounding box should be as low as possible. If the tightness is not satisfied, a robot has to travel longer paths than necessary to avoid obstacles and is not allowed to pass an alley even though it is possible in practice. The second criterion is that the collision avoidance conditions should be calculated simply and fast. Because the collision avoidance has to be guaranteed while a robot performs complicated tasks, the algorithm must utilize just a little part of the computational resources. The final criterion is to reduce the amount of information used to store a computer model [47]. With compact representation of the shape, a robot can not only improve the running time for planning a collision-free path but also model the shape of a newly detected obstacle from equipped sensors without the risk of overfitting.

In general, both circles and polygons have been typical candidates for enclosing objects due to its simplicity in collision detection in the 2D plane. For example, a collision between two circles (or polygons) is equivalent to that between a point and an inflated circle (or polygon) in the configuration space (C-space) introduced in [77]. Then it is easy to test the inclusion of the point in the configuration space obstacle (C-obstacle). However, there is a trade-off between the tightness and the complexity. While it is specified by just three variables (x-position, y-position, and radius), the bounding circle becomes overly conservative if an object is elongated. To remedy this problem, some authors approximated the object by a group of several circles in [18, 41, 108]. In case of the bounding polygon, the convex hull of the object may satisfy the tightness, but the complexity increases as much as twice of the number of the vertices.

In this respect, approximating an object with the minimum area bounding ellipse, known as the Löwner ellipse [62], compromises between these two criteria. It has been stated in [19, 23] that far fewer ellipses than circles are required to enclose a given object with the same degree of the tightness. Also, an ellipse is characterized by five variables (x-position, y-position, major radius, minor radius, and orientation), which is lower than that of the simplest polygon—an triangle. However, collision detection between ellipses is more difficult than those between circles and between polygons [31]. Therefore, this dissertation deals with the collision detection problem between two ellipses and exploits it to plan the motion of elliptic robots based on the velocity-based approach.

Meanwhile, there have been few researches about the local collision avoidance of anisotropic robots with rotation, whereas the velocity-based methods have mostly been studied on circular robots. A circular robot does not control its orientation because it is rotationally-invariant. On the other hand, it is essential for an anisotropic robot to consider its angular motion because its collision-free motion depends on its orientation. Hence, it is important to find out the optimal orientation that enables the robot to avoid collision through the shortest path possible.

The previous work was about the polygonal robot. [64] suggested the formation velocity obstacles (FVO), where the orientation of a formation was supposed to be parallel to the direction of motion. In [45], the concept of reciprocally-rotating velocity obstacle (RRVO) was introduced to solve the problem of the deadlock, emerged when polygonal robots tried to avoid collisions without rotation. This approach found the range of reachable orientations by discretizing the set of rotations of the robot and employed a brute-force search strategy in order to find the optimal orientation. Moreover, the collision avoidance between elliptic agents was suggested by approximating an ellipse with multiple piecewise lines in [12]. This method used precomputed Minkowski sum approximations for practical real-time applications, but instead required a lot of memory. However, there was a drawback that real-time operation was difficult when the robot encountered unexpected obstacles.

For the real-time collision avoidance, analytic solutions of the motion planning of elliptic robots have to be researched. Therefore, this dissertation addresses the local collision avoidance of elliptic robots so that they reach their goal efficiently with rotation without collisions.

1.2 Statement of the Problem

The problem of real-time local collision avoidance for elliptic robots is addressed in this dissertation, where each of them makes a detour to reach its goal without collisions with other robots and obstacles.

Consider elliptic robots $\mathcal{R}_i$, $i = 1, \cdots, N_{\mathcal{R}}$ moving in a planar workspace $\mathcal{W}$. Each robot $\mathcal{R}_i$ has major and minor radii of $M_{\mathcal{R}_i}$ and $m_{\mathcal{R}_i}$. Also, its configuration consists of the position $\mathbf{p}_{\mathcal{R}_i} \in \mathbb{R}^2$ and orientation $\theta_{\mathcal{R}_i} \in (-\pi, \pi]$ since it has a holonomic and omni-directional movable ability. The position $\mathbf{p}_{\mathcal{R}_i}$ is the center of the ellipse, and the orientation $\theta_{\mathcal{R}_i}$ is the angle between the major axis and the x-axis. The shape of robot $\mathcal{R}_i$ is represented by a symmetric positive definite matrix $\mathbf{S}_{\mathcal{R}_i} \in \mathbb{R}^{2\times 2}$ whose eigenvalues are squares of the semi-axes and eigenvectors indicate the principal axes, defined by

$$\mathbf{S}_{\mathcal{R}_i}\left(\theta_{\mathcal{R}_i}\right) = \mathbf{R}_{\theta_{\mathcal{R}_i}} \begin{bmatrix} M_{\mathcal{R}_i}^2 & 0 \\ 0 & m_{\mathcal{R}_i}^2 \end{bmatrix} \mathbf{R}_{\theta_{\mathcal{R}_i}}^T, \tag{1.1}$$

where $\mathbf{R}_\theta$ is the rotation matrix corresponding to a counter-clockwise rotation of angle θ. Let f be a function given by

$$f\left(\mathbf{x}; \mathbf{S}, \mathbf{p}\right) = \left(\mathbf{x} - \mathbf{p}\right)^T \mathbf{S}^{-1}\left(\mathbf{x} - \mathbf{p}\right) - 1. \tag{1.2}$$

Then the footprint $\mathcal{F}_{\mathcal{R}_i}$, the occupied region of its workspace, is defined by

$$\mathcal{F}_{\mathcal{R}_i}(\theta_{\mathcal{R}_i}) = \left\{\mathbf{x} \in \mathcal{W} \,|\, f\left(\mathbf{x}; \mathbf{S}_{\mathcal{R}_i}(\theta_{\mathcal{R}_i}), \mathbf{p}_{\mathcal{R}_i}\right) \leq 0\right\}. \tag{1.3}$$

In addition, each robot $\mathcal{R}_i$ moves with its linear velocity $\mathbf{v}_{\mathcal{R}_i}$ and angular velocity $w_{\mathcal{R}_i}$. These two factors satisfy its dynamic constraints such as the maximum linear speed $v_{\mathcal{R}_i}^{max}$, angular speed $w_{\mathcal{R}_i}^{max}$, linear acceleration $a_{\mathcal{R}_i}^{max}$, and angular acceleration $\alpha_{\mathcal{R}_i}^{max}$. Furthermore, there is no communication between robots, so a robot have to observe other robots with equipped sensors.

In view of the fact that sensor data is returned at a certain sampling period

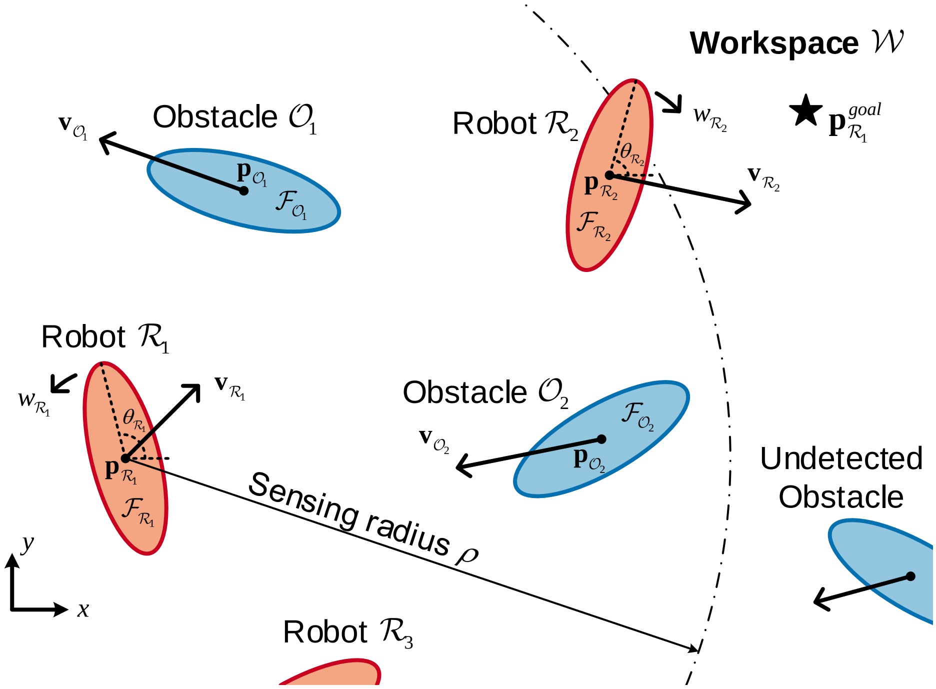

Workspace $\mathcal{W}$
Obstacle $\mathcal{O}_1$
$\mathbf{v}_{\mathcal{O}_1}$
$\mathbf{p}_{\mathcal{O}_1}$
$\mathcal{F}_{\mathcal{O}_1}$
Robot $\mathcal{R}_2$
$w_{\mathcal{R}_2}$
$\theta_{\mathcal{R}_2}$
$\mathbf{p}_{\mathcal{R}_2}$
$\mathcal{F}_{\mathcal{R}_2}$
$\mathbf{v}_{\mathcal{R}_2}$
$\mathbf{p}_{\mathcal{R}_1}^{goal}$
Robot $\mathcal{R}_1$
$w_{\mathcal{R}_1}$
$\mathbf{v}_{\mathcal{R}_1}$
$\theta_{\mathcal{R}_1}$
$\mathbf{p}_{\mathcal{R}_1}$
$\mathcal{F}_{\mathcal{R}_1}$
Obstacle $\mathcal{O}_2$
$\mathcal{F}_{\mathcal{O}_2}$
$\mathbf{v}_{\mathcal{O}_2}$
$\mathbf{p}_{\mathcal{O}_2}$
Sensing radius ρ
Undetected Obstacle
y
x
Robot $\mathcal{R}_3$

Figure 1.1 Elliptic robots and obstacles in the workspace $\mathcal{W}$. Robot $\mathcal{R}_1$ can sense the other robots and the obstacles within the range of ρ: $\mathcal{R}_2$, $\mathcal{R}_3$, $\mathcal{O}_1$, and $\mathcal{O}_2$. The objective of the robot is to arrive at $\mathbf{p}_{\mathcal{R}_1}^{goal}$ without collisions.

Δt, a discrete-time robot model is employed. At each time instant, each robot follows a sense-plan-act sequence.

In the first step, a robot detects other robots and obstacles with an omni-directional range sensor with a detection range up to ρ, as shown in Fig. 1.1. The detection region D_i of robot $\mathcal{R}_i$ is defined by

$$D_i = \{\mathbf{x} \in \mathcal{W} \,|\, \|\mathbf{x} - \mathbf{x}_{\mathcal{R}_i}\| \leq \rho\}\,. \tag{1.4}$$

Let $\mathcal{NR}_i$ denote the sets of robots detected by robot $\mathcal{R}_i$. For any robot $\mathcal{R}_j \in$

$\mathcal{NR}_i$, it is satisfied that

$$\mathcal{F}_{\mathcal{R}_j} \cap D_i \neq \emptyset. \tag{1.5}$$

Because each robot is assumed to identify all other robots, robot $\mathcal{R}_i$ knows the shape of robot $\mathcal{R}_j \in \mathcal{NR}_i$. Therefore, robot $\mathcal{R}_i$ can measure or extract the information of the position $\mathbf{p}_{\mathcal{R}_j}$, orientation $\theta_{\mathcal{R}_j}$, linear velocity $\mathbf{v}_{\mathcal{R}_j}$, and angular velocity $w_{\mathcal{R}_j}$ of robot $\mathcal{R}_j$.

On the other hand, each robot does not have any information of obstacles in advance. If the range sensor detects something other than robots, it is regarded as an obstacle. Without loss of generality, suppose that there are obstacles $\mathcal{O}_j$, $j = 1, \cdots, N_{\mathcal{O}}$ detected by robot $\mathcal{R}_i$. Because it is difficult to measure the angular velocity without the exact knowledge of the obstacles' shape, the obstacles are assumed to move in straight lines without rotation. After some obstacle $\mathcal{O}_j$ was first detected by robot $\mathcal{R}_i$, it has gathered the sensor data about the outline of the obstacle. From those data, the robot approximates the shape of obstacle $\mathcal{O}_j$ as the minimum area bounding ellipse of the accumulated points, represented by a symmetric positive semidefinite matrix $\mathbf{S}_{\mathcal{O}_j}$. If the obstacle is rotating, it would be bounded by a circle whose radius is the maximum distance from the center to its boundary. In addition, both the position $\mathbf{p}_{\mathcal{O}_j}$ and the linear velocity $\mathbf{v}_{\mathcal{O}_j}$ are measured by robot $\mathcal{R}_i$, and its footprint $\mathcal{F}_{\mathcal{O}_j}$ is defined by

$$\mathcal{F}_{\mathcal{O}_j} = \left\{ \mathbf{x} \in \mathcal{W} \,\middle|\, f\left(\mathbf{x}; \mathbf{S}_{\mathcal{O}_j}, \mathbf{p}_{\mathcal{O}_j}\right) \leq 0 \right\}. \tag{1.6}$$

In the next step, each robot plans a trajectory toward its goal from the information received by the equipped sensor. At each time instant, robot $\mathcal{R}_i$ determines a control input including its new linear velocity $\mathbf{v}_{\mathcal{R}_i}^{new}$ and angular velocity $w_{\mathcal{R}_i}^{new}$ for the next sampling period that guarantee no collision with other robots and obstacles within a time horizon $\tau \geq 0$. Moreover, these new velocities are taken within the dynamic constraints in accordance with its preferred linear velocity $\mathbf{v}_{\mathcal{R}_i}^{pref}$ and angular velocities $w_{\mathcal{R}_i}^{pref}$, respectively. If there were not other robots and obstacles in the workspace, a robot would move with its preferred linear velocity without rotation to reach the goal as soon as possible. When the robot's goal located at $\mathbf{p}_{\mathcal{R}_i}^{goal}$ is given by an external global planner, $\mathbf{v}_{\mathcal{R}_i}^{pref}$ has a magnitude of the robot's preferred linear speed $v_{\mathcal{R}_i}^{pref}$ and is directed toward $\mathbf{p}_{\mathcal{R}_i}^{goal}$. On the other hand, the preferred angular velocity enables the robot to avoid obstacles more efficiently on the basis of its geometric shape. If there is no other robot and obstacle, the robot does not need to turn around, that is $w_{\mathcal{R}_i}^{pref} = 0$. Otherwise, the robot needs rotating with the preferred angular velocity so that the robot circumvents others with the minimum possible deviation from the shortest path.

Finally, each robot $\mathcal{R}_i$ executes the determined command and changes its velocities to $\mathbf{v}_{\mathcal{R}_i}^{new}$ and $w_{\mathcal{R}_i}^{new}$.

Based on the above descriptions, the problem to be solved in this dissertation is defined as follows.

Problem 1.1 (Collision Avoidance of Elliptic Robots) *For any elliptic robot $\mathcal{R}_i$, compute the new linear velocity $\mathbf{v}_{\mathcal{R}_i}^{new}$ and angular velocity $w_{\mathcal{R}_i}^{new}$ in order to generate a trajectory to the goal that circumvents other robots and obstacles with the minimum possible deviation from*

- *the measurements of other robots and obstacles in its detection area D_i,*
- *the robot's goal position $\mathbf{p}_{\mathcal{R}_i}^{goal}$.*

1.3 Contributions

The main contributions of this dissertation into three parts.

First, we investigate two methods for the collision detection between two ellipses in Chapter 3: a configuration space framework and an algebraic approach. First of all, the configuration space obstacle (C-obstacle) of an elliptic obstacle with respect to an elliptic robot is defined, and its region is identified by finding the locus of its boundary points. It is also shown that an elliptic robot can be regarded as a circular robot with radius equal to its minor radius by adequately controlling its orientation. This facilitates an efficient framework for collision avoidance considering the geometric shape of the robot. However, it is difficult to immediately decide whether the collision occurs with the C-obstacle. To overcome this drawback, the interior-disjoint of two ellipses is analyzed algebraically. This enables an elliptic robot to determine whether its nonlinear motion causes a collision in the near future. [1]

Second, we present a velocity-based local collision avoidance method for an elliptic robot in dynamic environments with moving obstacles in Chapter 4. This method consists of two parts: linear and angular motion planning. In the first part, the ellipse-based velocity obstacle (EBVO), a set of linear velocities of an elliptic robot that would induce a collsion with obstacles within a finite horizon, is derived. Next, a strategy for selecting the new linear velocity closest to the preferred linear velocity and outside the approximated EBVO is presented. In the second part, the time to contact is calculated when the robot maintains its new linear velocity, and the collision-free rotation angles are calculated in the

[1] This content is reproduced by permission of the Institution of Engineering & Technology [59].

time to contact. Next, the preferred angular velocities are calculated that enable the robot to detour obstacles with the minimum deviation from the straight path to its goal. Finally, a strategy for selecting the new angular velocity, which is similar with that of the first part, is presented. The evaluation and comparison of the proposed method are presented in Chapter 6. [2]

Lastly, we extends the above method in multi-robot systems in Chapter 5. In order to account for reciprocity between robots, the concept of hybrid reciprocal velocity obstacles is adopted in the part of linear motion planning. In addition, the collision-free reciprocal rotation angles are calculated on the assumption that if one robot rotates by $\Delta\theta$, then the other robot may rotate by $\Delta\theta$ equally or $-\Delta\theta$ equally opposite. Likewise, the evaluation and comparison of the proposed method are presented in Chapter 6.

1.4 Organization

The rest of dissertation is organized as follows. Chapter 2 reviews relevant literature in the areas of bounding ellipsoid, collision detection between ellipsoids, and velocity-based local navigation. Chapter 3 presents two frameworks for collision detection between two ellipses. Chapter 4 addresses the problem of the local collision avoidance problem for a holonomic elliptic robot in dynamic environments with multiple elliptic obstacles. Chapter 5 modifies the approach proposed in Chapter 4 for multi-robot collision avoidance. Chapter 6 discusses the implementation of the algorithms with simulations. We conclude in Chapter 7 with suggestions for future work.

[2] This content is reproduced by permission of the Institution of Engineering & Technology [59] and Springer [73].

Chapter 2

Literature Review

This chapter outlines literature on bounding ellipsoid (or ellipse), collision detection between ellipsoids (or ellipses), and velocity-based local navigation, which are related with this dissertation.

2.1 Bounding Ellipsoid

The minimum-volume enclosing ellipsoid (or the minimum-area enclosing ellipse) of a given point set has been called the Löwner ellipsoid (or ellipse), which is named after Karel (or Charles) Löwner who proved the uniqueness of that ellipsoid, according to [10] and [17]. Afterwards, the proof of the uniqueness in the general case was first presented in [62]. Here, the uniqueness of the maximal volume ellipsoid inscribed to a convex body, called the John ellipsoid, was also proved. The both extremal volume ellipsoids have been called Löwner-John ellipsoids. Additionally, the uniquenesses of the minimal enclosing ellipsoids for various different size functions was demonstrated in [97].

The classical problem of finding the Löwner ellipsoid for a given set of points has been studied for a long time. An exact algorithm was suggested in [89], which ran in $O\left(n^2\right)$ in terms of the number n of points. Afterward, a number of numerical algorithms whose computational complexity is linear in the number n have been suggested. These algorithms were classified as first-order algorithms [65, 69, 100, 111, 112, 113], second-order interior-point algorithms [27, 106], and combination of the both [65]. For small dimensions, the problem was able to be solved faster by employing randomized [121] or deterministic [20] algorithm.

More than one ellipsoid has been used to approximate an object of arbitrary shape. An ellipsoid decomposition method was suggested for the robust transmission of a geometric object in [13], which was applied to the human body modeling and animation in [56]. Also, [101] represented a mesh surface with multiple ellipsoids by utilizing the Lloyd method on metric spaces endowed with Euclidean radial distance, surface normals, and curvatures. To calculate the union of tight bounding volumes and improve the efficiency, a variational method was proposed to compute the optimal segmentation in [78].

Meanwhile, the Löwner-John ellipsoids were employed to measure the distance between a robot and its surrounding environment. A conservative estimate method was suggested in [94], where the distance between two convex polyhedra was approximated with that between the associated Löwner ellipsoids. After that, it was improved in [63, 98] by computing the upper and lower bounds with the both Löwner and John ellipsoids.

2.2 Collision Detection between Ellipsoids

In applications of the bounding ellipsoids, it has been important to detect a collision between two ellipsoids. For this purpose, conventional methods [37, 75, 123] could be used to find the intersection of two quadric surfaces. If there were no point of the intersection between them, either these ellipsoids were separated or one was contained in the other. Nevertheless, these methods were inefficient to figure out the positional relationship, such as separation, external touching, and overlapping, because they were designed to determine the structure and parameterize the intersection curve, as mentioned in [120].

The overlapping condition between ellipsoids has been studied in computational physics in geomechanics in order to model ellipsoidal particles. There has been three different algorithms to determine the contact point of two ellipsoids (or ellipses). First, the contact point was defined in [95] as the mid-point of the line connecting the intersection points of the two ellipses. However, this method was not only difficult to be extended to the three-dimensional case, but also poorly conditioned and inaccurate. Second, more robust algorithms were presented on the basis of a geometric potential concept in [109, 110], where they located two specific points, not the intersection points, by minimizing the geometric potential functions of each ellipsoids. Nevertheless, these methods had an imperfection that the normal vectors in the specific points might not be parallel to each other. In view of the fact, the third algorithm exploiting a common normal concept was proposed in [76, 83].

Some researchers have focused on calculating the distance between two ellipsoids. [28] suggested an iteration scheme based on the convexity of the ellipsoid surface in the field of astronautics, where the problem of computing the closest distance between a point and an ellipsoid was repeatedly solved. Meanwhile, [34] explained the procedure of finding the minimum distance between ellipses by utilizing the Lagrange multipliers, which was a system of four equations in four variables. This methods were improved to a system of two equations in two variables in [74, 105]. However, the distance computation has been a more difficult problem than collision detection because a positive distance implies no collision between the two.

Several studies have been proposed to deal with just the problem of detecting a collision between two ellipsoids (or ellipses). [125] suggested an iterative procedure based on a concept similar to the bisection method. He suggested and proved a theorem that determined whether two ellipsoids intersected at a point only or not, and then confirmed whether inflated ellipsoids could intersect at a point only for various scale factors. Meanwhile, [35] transformed two ellipses to a circle centered at the origin and an axis-aligned ellipse using an affine mapping. Next, the relationship between the ellipse and the circle was determined by computing the extreme points on the ellipses that were closest and farthest from the origin.

The collision detection problem also studied in computational physics for molecule simulation. [87] designed a contact function for two arbitrary ellipsoids representing molecules. As a result, the numerical value of the function was less than one if they overlapped, and it was greater than one if they did not. Afterward, this function was generalized and simplified in [88] with clarifying the relation of the ellipsoid contact to the Gaussian overlap potentials. Also, it was extended to predict the time of collision between two moving ellipsoids (or ellipses) in [32, 33].

Other studies have connected the problem of collision detection to the literature on algebra. A simple algebraic condition for the separation between two ellipsoids was first established in [120]. They defined the characteristic polynomial of two ellipsoids and clarified that the two were separated if and only if the polynomial had two distinct positive real roots. [22] extended the result of [120] to take into account the time parameter and then performed the collision detection using the zero-set of the bivariate characteristic equation of two moving ellipsoids. This method was modified to devise more accurate and efficient algorithm in [21]. On the other hand, [23] analyzed the separation of two ellipses in a similar way to [120], but they showed that the problem of detection collisions of the two moving ellipses was reduced to a problem of detecting the zero of a univariate function due to the simplicity of ellipses. Furthermore, the concept of a separating plane was adopted in [119] to enhance the efficiency of collision detection. They utilized the fact that once a plane separating two ellipsoids was found, there could be no collision between the ellipsoids until one of them collided with the separating plane.

By using Sturm-Habicht sequences defined in [49], the conditions of the coefficients could be determined that the characteristic equation has exactly two positive real roots. [36] and [51] reflected this result to the cases of ellipses and ellipsoids, respectively. As a consequence, the collision conditions were summarized into four and six polynomials in each of the cases, and [61] additionally distinguished the two conditions of external touching and overlapping from the result of [51].

2.3 Velocity-based Local Navigation

Velocity-based collision avoidance methods have been widely adopted in real-world scenarios for tasks such as warehousing [39], autonomous cars [93] and ships [70], navigating robotic wheelchairs through crowded public environment [90, 91, 92], animation display with multiple ground robots [4, 5] and aerial robots [6], and crowd tracking [11, 66].

The concept of a velocity obstacle (VO) was first presented in [40], which was a set of all the velocities of a robot that would induce a collision within a finite time horizon. This region was calculated by representing the information of static and dynamic obstacles to the robot's velocity space. With this representation, the robot did not collide with any obstacles if it moved only at the velocity outside this region. Hence, an avoidance maneuver that selected its new velocity outside the VO was proposed and analyzed in [41]. However, the similar method has been published under different names as mentioned in [122]: a maneuvering board approach in [114], a collision cone in [18], and a forbidden velocity map in [30].

Whereas the above methods assumed that trajectories of obstacles were globally linear, the nonlinear velocity obstacle (NLVO) [99] was suggested to account for general as well as for linear trajectories. The NLVO of an obstacle was a warped cone that was its time-scaled map along the trajectory. When the exact trajectory of a moving obstacle was not available, it was estimated from the current linear and angular velocity. This method was demonstrated for realistic traffic scenarios on an expressway junction and a parking lot in [72]. Furthermore, the probabilistic velocity obstacle approach [43] was proposed to address uncertainty in the future trajectory of obstacles.

Meanwhile, the determination of a proper time horizon has been essential in the velocity-obstacle approach. Setting the time horizon too high would be too prohibitive because maneuvers that cause a collision at a distant time would be considered dangerous. On the other hand, setting it too small would fall a robot into an inevitable collision state [42] that is a state where a collision with an obstacle eventually occurs regardless of the robot's future trajectory. In view of the fact, [44] proposed a method that determined the obstacle specific safety time horizon by considering size of the obstacle, its velocity, and the robot's dynamic constraints.

For a non-holonomic mobile robot, [84] represented the constraints of the robot in the space of the angular and linear velocities. They assumed that the robot could move following straight or circular paths and its velocities were constant during a sampling period. This research was extended in [85] by imposing compound trajectories to be followed by the robot in order to maintain a continuous curvature. In a similar way, [60] also represented the forbidden velocity region in the wheel velocity space of a differential drive robot. Moreover, [124] accounted for the constraints of a car-like robot and formulated the velocity obstacle in terms of the set of the controls.

Heretofore, there was only one robot in the environment and all the rest were obstacles. However, such an approach was not sufficient when the robot tried to avoid collisions with other robots as reported in [38]. If a robot treated other robots as passively moving obstacles, undesirable oscillations were generated in the motion of robots. That was because other robots also perceived it and adapted its own motion accordingly, which violated the assumption of the concept of the velocity obstacle.

Many studies have been conducted on the local collision avoidance in multi-robot systems. They assumed that all the robots utilized the same decision making algorithm [46], which enabled robots to predict the motion of other robots in the near future without explicit communication. In this regard, the first extension of the VO for multi-robot systems was the common velocity obstacle map in [1], which was defined in the 4-dimensional space of all combinations of the velocities of two robots. However, it was unclear how this notion was extended for multiple robots or how well it scaled to more complicated environments [118].

Next, the reflective navigation method, called recursive probabilistic velocity obstacles, was proposed in [68]. In this method, the first robot determined its own velocity based on the expected motion of the second robot, and vice versa. This recursion continued until some termination condition was satisfied. Nevertheless, the convergence of the applied iterative method was difficult to be guaranteed since the chosen velocities might oscillate between odd and even level of recursion.

In addition, [118] showed the reason why the original VO generated oscillatory motions when used in navigation in multi-robot systems in the chicken scenario. They also suggested the concept of a reciprocal velocity obstacle (RVO) to remedy the oscillation problem without explicit communication, where each robot took half of the responsibility for collision avoidance and assumed that the other robot took the other half. However, robots frequently failed to collaborate for collision avoidance due to disagreement on which side to pass each other. In this case, they ended up in a reciprocal dance [38, 48], which causes collisions between them.

To address the reciprocal dance, the hybrid reciprocal velocity (HRVO) was presented in [102, 104], in which the RVO was expanded on the side that the robots should not pass. As a result, if a robot attempted to graze on the inappropriate side of another robot, it had to take the full responsibility for collision avoidance. Otherwise, the robot shared the responsibility in half. Additionally, a strategy for selection of a new collision-free velocity, the ClearPath algorithm, was described in [53].

However, the HRVO was known not to guarantee collision avoidance between multiple robots sufficiently in general even though it struggled to prevent the reciprocal dance. That was because two robots could select their new velocity on the opposite side due to influences of other robots. To overcome this drawback, the concept of the optimal reciprocal collision avoidance (ORCA) was presented in [116], which provided a sufficient condition for collision avoidance in multi-robot systems.

The ORCA was first suggested to apply robots with simple holonomic dynamics, but more recently, it has been extended to robots with differential-drive [3, 103], car-like [2], double-integrator [71, 115], arbitrary-degree integrator [96], and general linear dynamics [8]. However, these approaches had a major limitation that all robots were assumed to have exactly the same dynamics and control laws, which caused they did not exploited in non-homogeneous or non-linear systems. To overcome this problem, [9] not only unified all the previous approaches in terms of control obstacles, but also extended them to be utilized in general non-linear or non-homogeneous systems.

Chapter 3

Collision Detection

3.1 Introduction

In order to plan collision-free paths for elliptic robots, it is necessary to know under what conditions a collision occurs. A fundamental tool to detect the collision is the configuration space framework [77]. According to [24], the configuration of a robot is a complete specification of the position of every point in its footprint. In addition, the configuration space (C-space) of the robot is defined as the space of all the possible configurations. In the process of transformation from the workspace to the C-space of a particular robot, the robot is deflated to a point, whereas the others are inflated. The inflated ones are called configuration space obstacles (C-obstacles), defined as the set of the robot's configurations at which a collision occurs. As a result, the problem of the collision detection turns into that of determining the inclusion of a point in the C-obstacles.

In general, the C-obstacle region is identified by finding the boundary, where a robot and an obstacle touch but do not overlap. If a robot and an obstacle are disks, the boundary of the C-obstacle is a circle whose radius is the sum of the radii of both the robot and the obstacle. It is straightforward to determine whether the point robot is contained in the inflated disk by a second-order inequality. If they are polygons, the C-obstacle is derived by the Star algorithm [24], so that the region is represented by the intersection of a finite number of half spaces. Hence, a point located in the C-obstacle satisfies the corresponding linear inequalities. In this manner, the boundary of the C-obstacle of an elliptic obstacle to an elliptic robot is derived in Section 3.3.

However, the collision detection between two ellipses with the configuration space framework has been known to be computationally expensive. That is because the boundary of the C-obstacle has no simple geometric shape and the region is not bounded by a finite number of linear inequalities. To remedy this problem, a simple algebraic condition for checking the separation of two ellipses was presented in [23, 36]. They changed the problem of the collision detection to the one of detecting real zeros of the characteristic polynomial of the two ellipses. Whether the collision occurred for given position and orientation of the robot or not was determined according to the pattern of the roots of the polynomial. Nevertheless, it is necessary to clarify the condition for the interior-disjoint of two ellipses because the two robots do not collide when they are separated or externally touched.

Therefore, this chapter introduces both the configuration space framework and the algebraic approach since they are complementary to each other. The overall shape of the collision condition is uncovered through deriving the C-obstacle, and the collision is immediately detected by the algebraic condition. These two concepts facilitate the collision avoidance for elliptic robots in the following chapters.

This chapter is organized as follows. The problem of the collision detection is first formulated in Section 3.2. Next, the C-obstacle of an elliptic obstacle to an elliptic robot is derived to find the configurations at which the robot and the obstacle collide with each other in Section 3.3. Section 3.4 introduces an algebraic condition that two ellipses are interior-disjoint. Finally, the approaches to the collision detection are summarized in Section 3.5.

3.2 Problem Formulation

Without loss of generality, it is assumed that there are an elliptic robot $\mathcal{R}$ and an elliptic obstacle $\mathcal{O}$, which can be another robot in multi-robot systems. The robot $\mathcal{R}$ with major and minor radii of $M_{\mathcal{R}}$ and $m_{\mathcal{R}}$ is currently located at $\mathbf{p}_{\mathcal{R}}$ with an orientation of $\theta_{\mathcal{R}}$. Also, the obstacle $\mathcal{O}$ has a current position $\mathbf{p}_{\mathcal{O}}$ and footprint characterized by a matrix $\mathbf{S}_{\mathcal{O}}$.

The objective of this chapter is to find the set of the robot's configurations at which it intersects the obstacle $\mathcal{O}$ and to determine whether the two ellipses are overlapped.

3.3 Configuration Space Obstacle

For the collision detection, the robot $\mathcal{R}$ and the obstacle $\mathcal{O}$ are mapped from workspace $\mathcal{W}$ to the robot's configuration space $\mathcal{Q}$, which is a subset of $\mathbb{R}^2 \times \mathrm{SO}(2)$ because ellipses are anisotropic.

Then the C-obstacle of the obstacle $\mathcal{O}$ to the robot $\mathcal{R}$ is defined as follows:

Definition 3.1 (Configuration Space Obstacle) *The configuration space obstacle $QO_{\mathcal{R}|\mathcal{O}}$ of an elliptic obstacle $\mathcal{O}$ to an elliptic robot $\mathcal{R}$ is the set of the configurations at which the robot $\mathcal{R}$ intersects the obstacle $\mathcal{O}$, which has the form*

$$QO_{\mathcal{R}|\mathcal{O}} = \left\{ \begin{bmatrix} \mathbf{q}_\mathbf{x} \\ q_\theta \end{bmatrix} \in \mathcal{Q} \,\middle|\, \left(\{\mathbf{q}_\mathbf{x}\} \oplus \mathcal{F}_\mathcal{R}\left(q_\theta\right)\right) \cap \mathcal{F}_\mathcal{O} \neq \emptyset \right\}, \tag{3.1}$$

where $\oplus$ is the Minkowski sum operator defined as

$$\mathcal{A} \oplus \mathcal{B} = \{\mathbf{x} + \mathbf{y} | \mathbf{x} \in \mathcal{A}, \mathbf{y} \in \mathcal{B}\}. \tag{3.2}$$

For a fixed orientation θ of the robot $\mathcal{R}$, the curve $\partial QO_{\mathcal{R}|\mathcal{O}}\left(\theta\right)$ obtained as the intersection of the C-obstacle $QO_{\mathcal{R}|\mathcal{O}}$ with a plane $q_\theta = \theta$ is derived as follows.

Theorem 3.2 *Consider a robot $\mathcal{R}$ and an obstacle $\mathcal{O}$ in a planar workspace $\mathcal{W}$. If the orientation of the robot is fixed at θ, the C-obstacle $QO_{\mathcal{R}|\mathcal{O}}$ is cut out by a plane $q_\theta = \theta$. Let $QO_{\mathcal{R}|\mathcal{O}}\left(\theta\right)$ denote the slice of the C-obstacle $QO_{\mathcal{R}|\mathcal{O}}$ at the orientation of θ. Then the boundary of the slice, $\partial QO_{\mathcal{R}|\mathcal{O}}\left(\theta\right)$ is the locus of all points $\mathbf{q} = \begin{bmatrix} \mathbf{q}_\mathbf{x}^T & \theta \end{bmatrix}^T$ that satisfy the equation*

$$\mathbf{q}_\mathbf{x} = \mathbf{p}_\mathcal{O} - \mathbf{p}_\mathcal{R} + \mathbf{S}_\mathcal{O}^{\frac{1}{2}}\mathbf{u} + \frac{\mathbf{S}_\mathcal{R}\left(\theta\right)\mathbf{S}_\mathcal{O}^{-\frac{1}{2}}\mathbf{u}}{\left\|\mathbf{S}_\mathcal{R}\left(\theta\right)^{\frac{1}{2}}\mathbf{S}_\mathcal{O}^{-\frac{1}{2}}\mathbf{u}\right\|}, \tag{3.3}$$

where $\mathbf{u}$ is a parameter such that $\|\mathbf{u}\| = 1$.

Proof. Pick a point $\mathbf{q}_0 = \begin{bmatrix} \mathbf{q}_{0,\mathbf{x}}^T & \theta \end{bmatrix}^T \in \partial QO_{\mathcal{R}|\mathcal{O}}\left(\theta\right)$. Then two ellipses $\{\mathbf{q}_{0,\mathbf{x}}\} \oplus \mathcal{F}_\mathcal{R}\left(\theta\right)$ and $\mathcal{F}_\mathcal{O}$ touch each other externally from (3.1), as shown in Fig. 3.1. The tangential point $\mathbf{x}_c$ satisfies

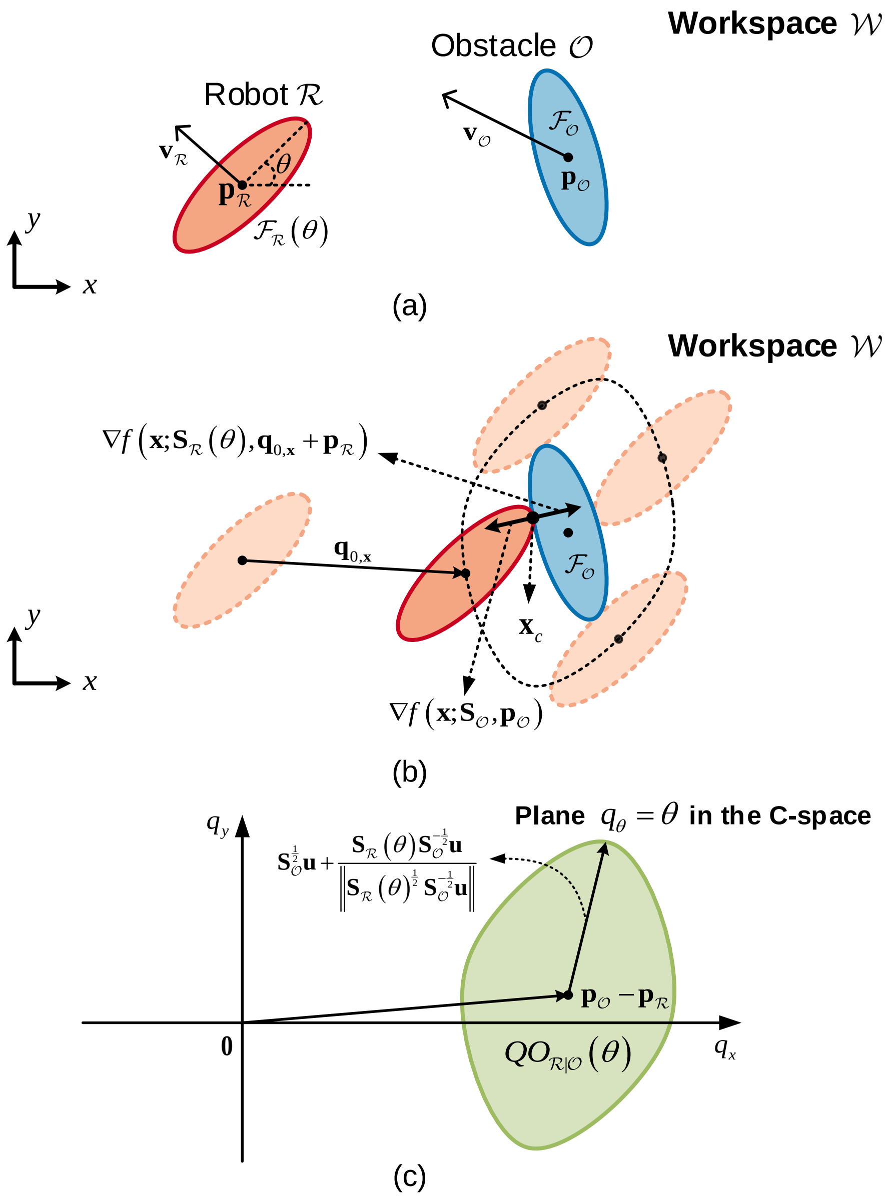

Workspace $\mathcal{W}$
Obstacle $\mathcal{O}$
Robot $\mathcal{R}$
$\mathbf{v}_{\mathcal{R}}$
θ
$\mathbf{p}_{\mathcal{R}}$
$\mathcal{F}_{\mathcal{R}}(\theta)$
$\mathbf{v}_{\mathcal{O}}$
$\mathcal{F}_{\mathcal{O}}$
$\mathbf{p}_{\mathcal{O}}$
y
x
(a)
Workspace $\mathcal{W}$
$\nabla f\left(\mathbf{x};\mathbf{S}_{\mathcal{R}}(\theta),\mathbf{q}_{0,\mathbf{x}}+\mathbf{p}_{\mathcal{R}}\right)$
$\mathbf{q}_{0,\mathbf{x}}$
$\mathcal{F}_{\mathcal{O}}$
$\mathbf{x}_c$
$\nabla f\left(\mathbf{x};\mathbf{S}_{\mathcal{O}},\mathbf{p}_{\mathcal{O}}\right)$
y
x
(b)
Plane $q_\theta=\theta$ in the C-space
q_y
$\mathbf{S}_{\mathcal{O}}^{\frac{1}{2}}\mathbf{u}+\dfrac{\mathbf{S}_{\mathcal{R}}(\theta)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}}{\left\|\mathbf{S}_{\mathcal{R}}(\theta)^{\frac{1}{2}}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}\right\|}$
$\mathbf{p}_{\mathcal{O}}-\mathbf{p}_{\mathcal{R}}$
$\mathbf{0}$
$QO_{\mathcal{R}|\mathcal{O}}(\theta)$
q_x
(c)

Figure 3.1 The construction process of the C-obstacle: (a) a robot $\mathcal{R}$ and an obstacle $\mathcal{O}$ in a workspace $\mathcal{W}$; (b) a contact point $\mathbf{x}_c$ between two ellipses, $\{\mathbf{q}_{0,\mathbf{x}}\} \oplus \mathcal{F}_{\mathcal{R}}(\theta)$ and $\mathcal{F}_{\mathcal{O}}$, and its trace obtained by sliding the orange ellipse around the blue ellipse; (c) the slice of the C-obstacle $QO_{\mathcal{R}|\mathcal{O}}(\theta)$ of the obstacle $\mathcal{O}$ with respect to the robot $\mathcal{R}$.

- $\mathbf{x}_c \in \partial\left(\{\mathbf{q}_{0,\mathbf{x}}\} \oplus \mathcal{F}_{\mathcal{R}}(\theta)\right)$;

- $\mathbf{x}_c \in \partial\mathcal{F}_{\mathcal{O}}$;

- for some $k > 0$, $\nabla f\left(\mathbf{x}; \mathbf{S}_{\mathcal{R}}(\theta), \mathbf{q}_{0,\mathbf{x}} + \mathbf{p}_{\mathcal{R}}\right) = -k\nabla f\left(\mathbf{x}; \mathbf{S}_{\mathcal{O}}, \mathbf{p}_{\mathcal{O}}\right)$.

From (1.1) and (1.2), the first condition means

$$\mathbf{x}_c = \mathbf{q}_{0,\mathbf{x}} + \mathbf{p}_{\mathcal{R}} + \mathbf{S}_{\mathcal{R}}(\theta)^{\frac{1}{2}}\,\tilde{\mathbf{u}}, \tag{3.4}$$

where $\tilde{\mathbf{u}} \in \mathbb{R}^2$ such that $\|\tilde{\mathbf{u}}\| = 1$. Likewise, the second one implies

$$\mathbf{x}_c = \mathbf{p}_{\mathcal{O}} + \mathbf{S}_{\mathcal{O}}^{\frac{1}{2}}\mathbf{u}, \tag{3.5}$$

where $\mathbf{u} \in \mathbb{R}^2$ such that $\|\mathbf{u}\| = 1$. Then

$$\mathbf{q}_{0,\mathbf{x}} = \mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}} + \mathbf{S}_{\mathcal{O}}^{\frac{1}{2}}\mathbf{u} - \mathbf{S}_{\mathcal{R}}(\theta)^{\frac{1}{2}}\,\tilde{\mathbf{u}}. \tag{3.6}$$

From the last condition, for some $k > 0$,

$$\mathbf{S}_{\mathcal{R}}(\theta)^{-\frac{1}{2}}\,\tilde{\mathbf{u}} = -k\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}, \tag{3.7}$$

which can be written as $\tilde{\mathbf{u}} = -k\mathbf{S}_{\mathcal{R}}(\theta)^{\frac{1}{2}}\,\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}$. From $\|\tilde{\mathbf{u}}\| = 1$, we obtain $k = \left\|\mathbf{S}_{\mathcal{R}}(\theta)^{\frac{1}{2}}\,\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}\right\|^{-1}$. Hence,

$$\tilde{\mathbf{u}} = -\frac{\mathbf{S}_{\mathcal{R}}(\theta)^{\frac{1}{2}}\,\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}}{\left\|\mathbf{S}_{\mathcal{R}}(\theta)^{\frac{1}{2}}\,\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}\right\|}. \tag{3.8}$$

Finally, (3.3) follows from (3.6) and (3.8). □

The first two terms in (3.3) represent the relative position of the obstacle with respect to the robot, and the last two terms describe its shape parameterized by a unit vector $\mathbf{u}$ and the robot's orientation θ, as shown in Fig. 3.1.

Furthermore, the boundary surface of the C-obstacle $QO_{\mathcal{R}|\mathcal{O}}$ is plotted in Fig. 3.2 for the parameters $\mathbf{u}$ and θ by assuming $\mathbf{p}_{\mathcal{R}} = \mathbf{p}_{\mathcal{O}}$ in order to concentrate on its shape. Viewed from the front, it takes after a twisted rod or a screw, of which the pitch between crests is π since ellipses are symmetric with respect to their major and minor axes. On the other hand, viewed from the above, it looks like an elongated circle in the direction along the major axis of the obstacle $\mathcal{O}$.

Theorem 3.2 finds out the forbidden positions by assuming a fixed orientation of the robot. Hereafter, the prohibited orientation is analyzed for a fixed position of the robot. To this end, $QO_{\mathcal{R}|\mathcal{O}}$ is first projected onto the q_x-q_y plane, which is equivalent to see the C-obstacle from the above as shown in Fig. 3.2(a).

Lemma 3.3 *Consider the C-obstacle $QO_{\mathcal{R}|\mathcal{O}}$ whose boundary is represented by* (3.3). *Then*

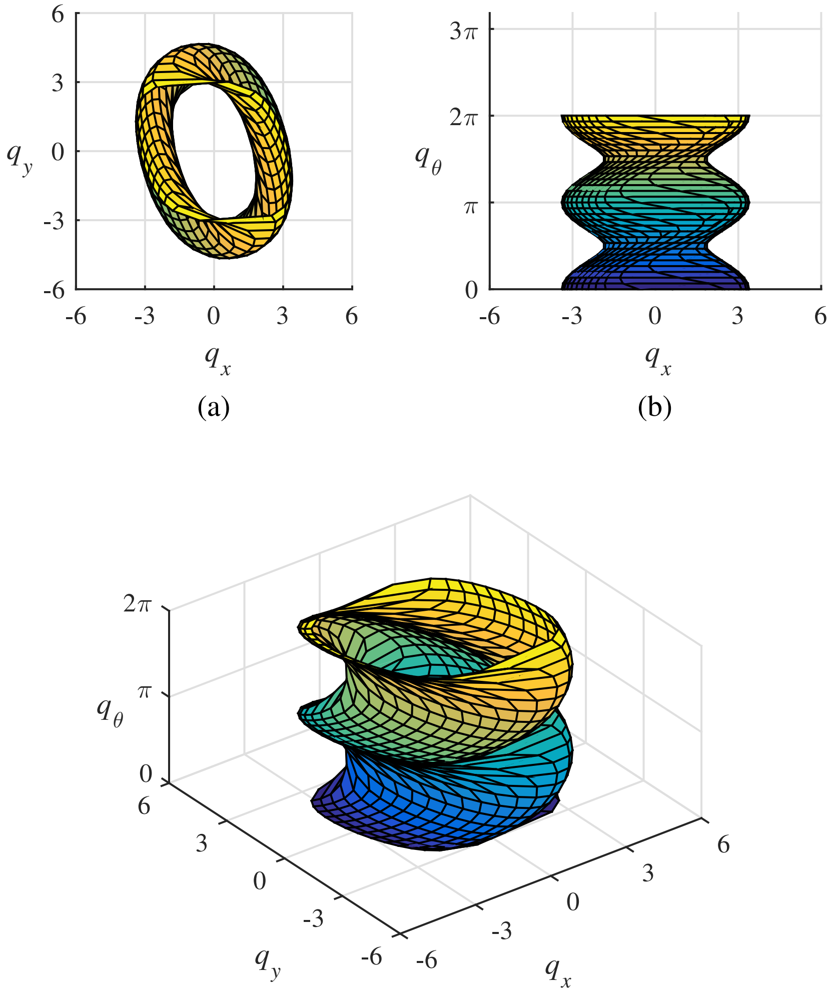
$$\bigcap_{\theta} QO_{\mathcal{R}|\mathcal{O}}(\theta) = QO_{D(\mathbf{p}_{\mathcal{R}}, m_{\mathcal{R}})|\mathcal{O}}, \tag{3.9}$$

$$\bigcup_{\theta} QO_{\mathcal{R}|\mathcal{O}}(\theta) = QO_{D(\mathbf{p}_{\mathcal{R}}, M_{\mathcal{R}})|\mathcal{O}}, \tag{3.10}$$

where $D(\mathbf{p}, r)$ is a disk with radius r located at $\mathbf{p}$ such that

$$D(\mathbf{p}, r) = \left\{\mathbf{x} \in \mathbb{R}^2 \,|\, \|\mathbf{x} - \mathbf{p}\| \le r\right\}. \tag{3.11}$$

Because a disk is rotationally invariant, $QO_{D(\cdot,\cdot)|\mathcal{O}} \in \mathbb{R}^2$.



(a)

(b)

(c)

Figure 3.2 The C-obstacle $QO_{\mathcal{R}|\mathcal{O}}$ in the C-space $\mathcal{Q}$ given by $M_{\mathcal{R}} = 2.3757$, $m_{\mathcal{R}} = 0.82$, $\mathbf{S}_{\mathcal{O}} = [\begin{smallmatrix} 0.9325 & 0.4028 \\ 0.4028 & 2.2632 \end{smallmatrix}]$, and $\mathbf{p}_{\mathcal{R}} = \mathbf{p}_{\mathcal{O}}$: (a) a top view; (b) a front view; (c) a view of azimuth –37.5° and elevation 30°.

Proof. From (3.1) and (3.2), the C-obstacle of an elliptic obstacle $\mathcal{O}$ to an elliptic robot $\mathcal{R}$ with fixed orientation θ is written as follows.

$$QO_{\mathcal{R}|\mathcal{O}}(\theta) = \left\{\mathbf{x} - \mathbf{y} \in \mathbb{R}^2 \left| \mathbf{x} \in \mathcal{F}_{\mathcal{O}}, \mathbf{y} \in \mathcal{F}_{\mathcal{R}}(\theta) \right.\right\}. \tag{3.12}$$

Hence, left-hand side of (3.9) is expressed as

$$\begin{aligned}\bigcap_{\theta} QO_{\mathcal{R}|\mathcal{O}}(\theta) &= \bigcap_{\theta} \left\{\mathbf{x} - \mathbf{y} \in \mathbb{R}^2 \left| \mathbf{x} \in \mathcal{F}_{\mathcal{O}}, \mathbf{y} \in \mathcal{F}_{\mathcal{R}}(\theta) \right.\right\} \\ &= \left\{\mathbf{x} - \mathbf{y} \in \mathbb{R}^2 \left| \mathbf{x} \in \mathcal{F}_{\mathcal{O}}, \mathbf{y} \in \bigcap_{\theta} \mathcal{F}_{\mathcal{R}}(\theta) \right.\right\} \\ &= \left\{\mathbf{x} - \mathbf{y} \in \mathbb{R}^2 \left| \mathbf{x} \in \mathcal{F}_{\mathcal{O}}, \mathbf{y} \in D(\mathbf{p}_{\mathcal{R}}, m_{\mathcal{R}}) \right.\right\} \\ &= QO_{D(\mathbf{p}_{\mathcal{R}}, m_{\mathcal{R}})|\mathcal{O}}, \end{aligned} \tag{3.13}$$

which proves (3.9).

Likewise, (3.10) is written as follows.

$$\begin{aligned}\bigcup_{\theta} QO_{\mathcal{R}|\mathcal{O}}(\theta) &= \bigcup_{\theta} \left\{\mathbf{x} - \mathbf{y} \in \mathbb{R}^2 \left| \mathbf{x} \in \mathcal{F}_{\mathcal{O}}, \mathbf{y} \in \mathcal{F}_{\mathcal{R}}(\theta) \right.\right\} \\ &= \left\{\mathbf{x} - \mathbf{y} \in \mathbb{R}^2 \left| \mathbf{x} \in \mathcal{F}_{\mathcal{O}}, \mathbf{y} \in \bigcup_{\theta} \mathcal{F}_{\mathcal{R}}(\theta) \right.\right\} \\ &= \left\{\mathbf{x} - \mathbf{y} \in \mathbb{R}^2 \left| \mathbf{x} \in \mathcal{F}_{\mathcal{O}}, \mathbf{y} \in D(\mathbf{p}_{\mathcal{R}}, M_{\mathcal{R}}) \right.\right\} \\ &= QO_{D(\mathbf{p}_{\mathcal{R}}, M_{\mathcal{R}})|\mathcal{O}}. \end{aligned} \tag{3.14}$$

This completes the proof. □

Lemma 3.3 furnishes convenient criteria for determining whether the rotation of the robot $\mathcal{R}$ causes a collision with the obstacle $\mathcal{O}$, described in Fig. 3.3.

- If the relative position of the obstacle with respect to the robot is in $QO_{D(\mathbf{p}_{\mathcal{R}},M_{\mathcal{R}})|\mathcal{O}}$, that is $\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}} \notin QO_{D(\mathbf{p}_{\mathcal{R}},M_{\mathcal{R}})|\mathcal{O}}$, the robot $\mathcal{R}$ with rotation does not collide with the obstacle $\mathcal{O}$.

- If the relative position is not in $QO_{D(\mathbf{p}_{\mathcal{R}},m_{\mathcal{R}})|\mathcal{O}}$ but in $QO_{D(\mathbf{p}_{\mathcal{R}},M_{\mathcal{R}})|\mathcal{O}}$, that is $\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}} \in QO_{D(\mathbf{p}_{\mathcal{R}},M_{\mathcal{R}})|\mathcal{O}} \setminus QO_{D(\mathbf{p}_{\mathcal{R}},m_{\mathcal{R}})|\mathcal{O}}$, the robot $\mathcal{R}$ can be overlapped with the obstacle $\mathcal{O}$ depending on its orientation.

- If the relative position is located in $QO_{D(\mathbf{p}_{\mathcal{R}},m_{\mathcal{R}})|\mathcal{O}}$, that is $\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}} \in QO_{D(\mathbf{p}_{\mathcal{R}},m_{\mathcal{R}})|\mathcal{O}}$, the robot $\mathcal{R}$ is overlapped with the obstacle $\mathcal{O}$ with all the orientation.

While it is unnecessary to consider the robot's orientation in the first and third cases, the orientation decides whether the collision occurs in the second case. In other words, the second case can be written as the following corollary:

Corollary 3.4 *If* $\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}} \in QO_{D(\mathbf{p}_{\mathcal{R}},M_{\mathcal{R}})|\mathcal{O}} \setminus QO_{D(\mathbf{p}_{\mathcal{R}},m_{\mathcal{R}})|\mathcal{O}}$*, there exists* θ *such that* $\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}} \notin QO_{\mathcal{R}|\mathcal{O}}(\theta)$.

To find the collision-free orientations for a given the position of the robot when the relative position is contained in $QO_{D(\mathbf{p}_{\mathcal{R}},M_{\mathcal{R}})|\mathcal{O}} \setminus QO_{D(\mathbf{p}_{\mathcal{R}},m_{\mathcal{R}})|\mathcal{O}}$, the inclusion of a point in the C-obstacle must be determined immediately. However, it is difficult to do with only the information of the boundary from Theorem 3.2, this will be discussed in detail in the next section.

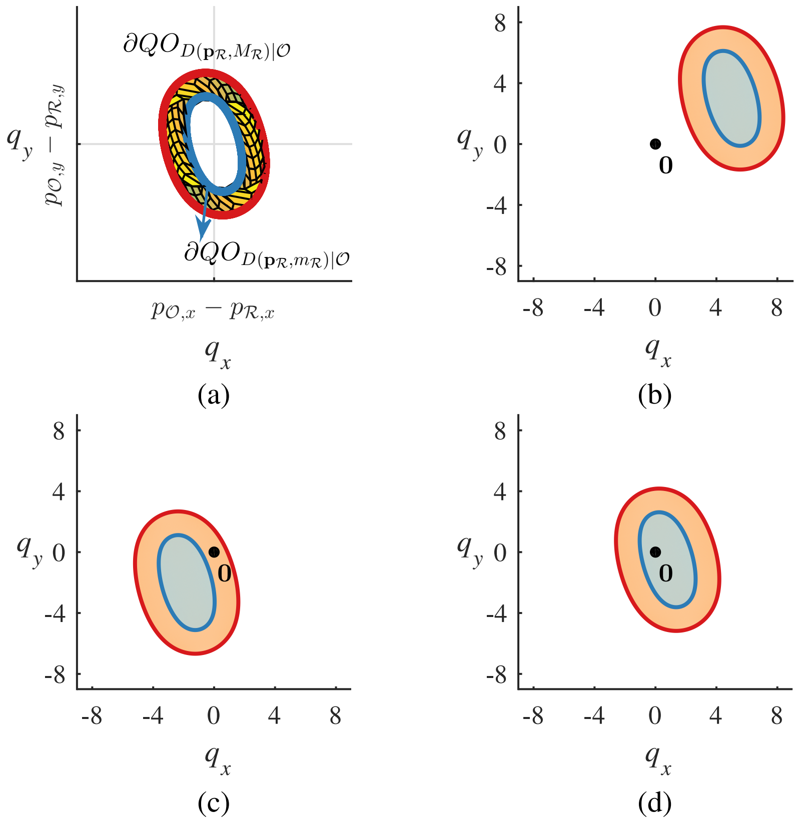

$\partial QO_{D(\mathbf{p}_{\mathcal{R}},M_{\mathcal{R}})|\mathcal{O}}$
$\partial QO_{D(\mathbf{p}_{\mathcal{R}},m_{\mathcal{R}})|\mathcal{O}}$
q_y
$p_{\mathcal{O},y} - p_{\mathcal{R},y}$
$p_{\mathcal{O},x} - p_{\mathcal{R},x}$
q_x
(a)
8
4
0
-4
-8
q_y
q_x
0
(b)
(c)
(d)

Figure 3.3 The proposed criteria for determining the overlap between the obstacle $\mathcal{O}$ and the robot $\mathcal{R}$ with rotation when $\mathbf{p}_{\mathcal{R}} = \mathbf{p}_{\mathcal{O}}$: (a) the boundaries of the union and intersection of the slice of the C-obstacle, $QO_{\mathcal{R}|\mathcal{O}}(\theta)$ for all the robot's orientation; (b) the robot is not overlapped with the obstacle for all the orientation; (c) the robot can be overlapped with the obstacle for some orientation; (d) the robot is overlapped with the obstacle for all the orientation. The red region represents $QO_{D(\mathbf{p}_{\mathcal{R}},M_{\mathcal{R}})|\mathcal{O}} \setminus QO_{D(\mathbf{p}_{\mathcal{R}},m_{\mathcal{R}})|\mathcal{O}}$, and the blue region represents $QO_{D(\mathbf{p}_{\mathcal{R}},m_{\mathcal{R}})|\mathcal{O}}$.

3.4 Algebraic Condition for the Interior-disjoint of Two Ellipses

In this section, the homogeneous coordinate is adopted in order to derive an algebraic condition for the interior-disjoint of a robot $\mathcal{R}$ and an obstacle $\mathcal{O}$. When an ellipse is located at $\mathbf{p}$ and has the shape matrix $\mathbf{S}$, its footprint $\mathcal{F}$ can be written as

$$\mathcal{F} = \left\{ \mathbf{x} \in \mathbb{R}^2 \middle| \left[\mathbf{x}^T \quad 1\right] \mathbf{M} \left[\mathbf{x}^T \quad 1\right]^T \leq 0 \right\}, \tag{3.15}$$

where $\mathbf{M}$ is a 3×3 symmetric coefficient matrix such that

$$\mathbf{M} = \begin{bmatrix} \mathbf{S}^{-1} & -\mathbf{S}^{-1}\mathbf{p} \\ -\mathbf{p}^T\mathbf{S}^{-1} & \mathbf{p}^T\mathbf{S}^{-1}\mathbf{p} - 1 \end{bmatrix}. \tag{3.16}$$

With respect to a local frame attached to $\mathcal{R}$, the robot $\mathcal{R}$ is at the origin and the obstacle $\mathcal{O}$ is located at $\mathbf{p}_{\mathcal{R}|\mathcal{O}} = \mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}}$. Then the coefficient matrix of the robot is represented as

$$\mathbf{M}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right) = \begin{bmatrix} \mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)^{-1} & \mathbf{0} \\ \mathbf{0} & -1 \end{bmatrix}, \tag{3.17}$$

and that of the obstacle is expressed as

$$\mathbf{M}_{\mathcal{O}} = \begin{bmatrix} \mathbf{S}_{\mathcal{O}}^{-1} & -\mathbf{S}_{\mathcal{O}}^{-1}\mathbf{p}_{\mathcal{R}|\mathcal{O}} \\ -\mathbf{p}_{\mathcal{R}|\mathcal{O}}^T\mathbf{S}_{\mathcal{O}}^{-1} & \mathbf{p}_{\mathcal{R}|\mathcal{O}}^T\mathbf{S}_{\mathcal{O}}^{-1}\mathbf{p}_{\mathcal{R}|\mathcal{O}} - 1 \end{bmatrix}. \tag{3.18}$$

As mentioned in [23, 36], the collision-free condition of two ellipses $\mathcal{F}_{\mathcal{R}}$ and $\mathcal{F}_{\mathcal{O}}$ is associated with the characteristic polynomial, defined as follows:

Definition 3.5 (Characteristic Polynomial of Two Ellipses) *The characteristic polynomial of two ellipses $\mathcal{F}_{\mathcal{R}}$ and $\mathcal{F}_{\mathcal{O}}$ is defined as*

$$g\left(\xi\right) = \det\left(\xi \mathbf{M}_{\mathcal{R}} + \mathbf{M}_{\mathcal{O}}\right). \tag{3.19}$$

Since $\mathbf{M}_{\mathcal{R}}, \mathbf{M}_{\mathcal{O}} \in \mathbb{R}^{3\times 3}$, the polynomial in (3.19) has a degree of 3 in ξ. Hence, it can be rewritten as

$$g\left(\xi\right) = a_3\xi^3 + a_2\xi^2 + a_1\xi + a_0. \tag{3.20}$$

Let $\bar{\mathbf{S}}_{\mathcal{R}} = \mathbf{S}_{\mathcal{R}}\left(0\right)$. Put (3.17) and (3.18) into (3.19), then

$$a_3 = -\det \bar{\mathbf{S}}_{\mathcal{R}}^{-1}, \tag{3.21}$$

$$a_2 = a_{22}\cos 2\theta_{\mathcal{R}} + a_{21}\sin 2\theta_{\mathcal{R}} + a_{20}, \tag{3.22}$$

$$a_1 = a_{12}\cos 2\theta_{\mathcal{R}} + a_{11}\sin 2\theta_{\mathcal{R}} + a_{10}, \tag{3.23}$$

$$a_0 = -\det \mathbf{S}_{\mathcal{O}}^{-1}, \tag{3.24}$$

where

$$a_{22} = \operatorname{tr}\left(\bar{\mathbf{S}}_{\mathcal{R}}^{-1}\mathbf{S}_{\mathcal{O}}^{-1}\right) - \frac{1}{2}\operatorname{tr}\bar{\mathbf{S}}_{\mathcal{R}}^{-1}\operatorname{tr}\mathbf{S}_{\mathcal{O}}^{-1}, \tag{3.25}$$

$$a_{21} = \operatorname{tr}\left(\mathbf{R}_{\frac{\pi}{2}}\bar{\mathbf{S}}_{\mathcal{R}}^{-1}\mathbf{S}_{\mathcal{O}}^{-1}\right), \tag{3.26}$$

$$a_{20} = \left(\mathbf{p}_{\mathcal{R}|\mathcal{O}}^T\mathbf{S}_{\mathcal{O}}^{-1}\mathbf{p}_{\mathcal{R}|\mathcal{O}} - 1\right)\det\bar{\mathbf{S}}_{\mathcal{R}}^{-1} - \frac{1}{2}\operatorname{tr}\bar{\mathbf{S}}_{\mathcal{R}}^{-1}\operatorname{tr}\mathbf{S}_{\mathcal{O}}^{-1}, \tag{3.27}$$

$$\begin{aligned} a_{12} = {} & \mathbf{p}_{\mathcal{R}|\mathcal{O}}^T\left(\bar{\mathbf{S}}_{\mathcal{R}}^{-1} - \frac{1}{2}\mathbf{I}_2\operatorname{tr}\bar{\mathbf{S}}_{\mathcal{R}}^{-1}\right)\mathbf{p}_{\mathcal{R}|\mathcal{O}}\det\mathbf{S}_{\mathcal{O}}^{-1} \\ & + \operatorname{tr}\left(\bar{\mathbf{S}}_{\mathcal{R}}^{-1}\mathbf{S}_{\mathcal{O}}^{-1}\right) - \frac{1}{2}\operatorname{tr}\bar{\mathbf{S}}_{\mathcal{R}}^{-1}\operatorname{tr}\mathbf{S}_{\mathcal{O}}^{-1}, \end{aligned} \tag{3.28}$$

$$a_{11} = \mathbf{p}_{\mathcal{R}|\mathcal{O}}^T \mathbf{R}_{\frac{\pi}{2}} \left(\bar{\mathbf{S}}_{\mathcal{R}}^{-1} - \frac{1}{2} \mathbf{I}_2 \mathrm{tr}\, \bar{\mathbf{S}}_{\mathcal{R}}^{-1} \right) \mathbf{p}_{\mathcal{R}|\mathcal{O}} \det \mathbf{S}_{\mathcal{O}}^{-1} + \mathrm{tr} \left(\mathbf{R}_{\frac{\pi}{2}} \bar{\mathbf{S}}_{\mathcal{R}}^{-1} \mathbf{S}_{\mathcal{O}}^{-1} \right), \quad (3.29)$$

$$a_{10} = \frac{1}{2} \mathrm{tr}\, \bar{\mathbf{S}}_{\mathcal{R}}^{-1} \left(\mathbf{p}_{\mathcal{R}|\mathcal{O}}^T \mathbf{p}_{\mathcal{R}|\mathcal{O}} \det \mathbf{S}_{\mathcal{O}}^{-1} - \mathrm{tr}\, \mathbf{S}_{\mathcal{O}}^{-1} \right) - \det \mathbf{S}_{\mathcal{O}}^{-1}. \quad (3.30)$$

Lemma 3.6 *Consider two ellipses $\mathcal{F}_{\mathcal{R}}$ and $\mathcal{F}_{\mathcal{O}}$ defined by* (3.15)*,* (3.17)*, and* (3.18)*. Then the characteristic polynomial of the two ellipses has a negative real root.*

Proof. This proof is based on [23, 36]. The characteristic polynomial is $g(\xi) = a_3\xi^3 + a_2\xi^2 + a_1\xi + a_0$ from (3.20). Since $a_3 = -\det \bar{\mathbf{S}}_{\mathcal{R}}^{-1} < 0$, $g(\infty) < 0$ and $g(-\infty) > 0$. In addition, $g(0) = a_0 = -\det \mathbf{S}_{\mathcal{O}}^{-1} < 0$. Therefore, $g(\xi) = 0$ has at least one negative real root. □

Lemma 3.7 *Consider two ellipses $\mathcal{F}_{\mathcal{R}}$ and $\mathcal{F}_{\mathcal{O}}$ defined by* (3.15)*,* (3.17)*, and* (3.18)*. Then the characteristic polynomial of the two ellipses has two positive and one negative real roots if and only if they are interior-disjoint.*

Proof. This proof is based on [23]. Because $a_0/a_3 > 0$, the product of the three roots of the characteristic polynomial is negative. Since one of the roots is negative from Lemma 3.6, the other two roots are positive if the polynomial has a positive real roots. Suppose that

$$\xi = \frac{1-\zeta}{\zeta}, \quad (3.31)$$

which maps $\zeta \in [0, 1]$ into $\xi \in [0, \infty)$. Hence, the characteristic polynomial $g(\xi) = \det(\xi \mathbf{M}_{\mathcal{R}} + \mathbf{M}_{\mathcal{O}})$ is transformed into

$$h(\zeta) = \det\left((1-\zeta)\mathbf{M}_{\mathcal{R}} + \zeta \mathbf{M}_{\mathcal{O}}\right), \quad (3.32)$$

which means $g(\xi) = 0$ has a finite positive root if and only if $h(\zeta) = 0$ has a real root in $(0,1)$. Accordingly, it will be shown that $h(\zeta) = 0$ has a real root in $(0,1)$ if and only if the two ellipses are interior-disjoint.

Let us prove the "if" part first by contradiction on the assumption that $h(\zeta) = 0$ has no real root in $(0,1)$. Since $h(\zeta)$ is a continuous function of ζ and is negative at $\zeta = 0$ and $\zeta = 1$, we see that $h(\zeta) < 0$ for $\zeta \in [0,1]$. That is why

$$\begin{aligned}\bar{\mathbf{M}}(\zeta) &= (1-\zeta)\,\mathbf{M}_{\mathcal{R}} + \zeta\mathbf{M}_{\mathcal{O}} \\ &= \begin{bmatrix} (1-\zeta)\,\mathbf{S}_{\mathcal{R}}(\theta_{\mathcal{R}})^{-1} + \zeta\mathbf{S}_{\mathcal{O}}^{-1} & -\zeta\mathbf{S}_{\mathcal{O}}^{-1}\mathbf{p}_{\mathcal{R}|\mathcal{O}} \\ -\zeta\mathbf{p}_{\mathcal{R}|\mathcal{O}}^{T}\mathbf{S}_{\mathcal{O}}^{-1} & \zeta\mathbf{p}_{\mathcal{R}|\mathcal{O}}^{T}\mathbf{S}_{\mathcal{O}}^{-1}\mathbf{p}_{\mathcal{R}|\mathcal{O}} - 1 \end{bmatrix}.\end{aligned} \tag{3.33}$$

is invertible for all $\zeta \in [0,1]$. Moreover, because the set of symmetric positive definite matrices is closed under addition and non-negative scaling, the block matrix expressed as $(1-\zeta)\,\mathbf{S}_{\mathcal{R}}(\theta_{\mathcal{R}})^{-1} + \zeta\mathbf{S}_{\mathcal{O}}^{-1}$ is still symmetric positive definite. Hence, the region

$$\mathcal{E}(\zeta) = \left\{\mathbf{x} \in \mathbb{R}^2 \,\middle|\, \begin{bmatrix}\mathbf{x}^T & 1\end{bmatrix} \bar{\mathbf{M}}(\zeta) \begin{bmatrix}\mathbf{x}^T & 1\end{bmatrix}^T \leq 0\right\} \tag{3.34}$$

is an ellipse with the center located at $\bar{\mathbf{p}}(\zeta)$ such that $\bar{\mathbf{P}}(\zeta) = \begin{bmatrix} w\bar{\mathbf{p}}(\zeta)^T & w\end{bmatrix} = \begin{bmatrix}0 & 0 & 1\end{bmatrix} \bar{\mathbf{M}}(\zeta)^{-1}$ for a nonzero real number w and all $\zeta \in [0,1]$. Let us denote

$$\gamma(\zeta) = \bar{\mathbf{P}}(\zeta)^T \mathbf{M}_{\mathcal{R}} \bar{\mathbf{P}}(\zeta). \tag{3.35}$$

Since $\bar{\mathbf{p}}(0)$ is the center of $\mathcal{F}_{\mathcal{R}}$, $\gamma(0) = \bar{\mathbf{P}}(0)^T \mathbf{M}_{\mathcal{R}} \bar{\mathbf{P}}(0) < 0$. On the other hand, $\bar{\mathbf{p}}(1)$ is outside of the ellipse $\mathcal{F}_{\mathcal{O}}$ because the two ellipses are interior-disjoint, that is $\gamma(1) = \bar{\mathbf{P}}(1)^T \mathbf{M}_{\mathcal{R}} \bar{\mathbf{P}}(1) > 0$. Because $\gamma(\zeta)$ is a continuous function of

$\zeta \in [0,1]$, there exists $\zeta_1 \in (0,1)$ such that $\gamma(\zeta_1) = 0$, which means that $\bar{\mathbf{p}}(\zeta_1)$ is on the boundary of the ellipse $\mathcal{F}_{\mathcal{R}}$.

Since the two ellipses are either separated or touching, the point $\bar{\mathbf{p}}(\zeta_1)$ can be outside of or on the boundary of $\mathcal{F}_{\mathcal{O}}$. If $\bar{\mathbf{p}}(\zeta_1) \notin \mathcal{F}_{\mathcal{O}}$, there exists $\epsilon > 0$ such that $D(\bar{\mathbf{p}}(\zeta_1), \epsilon) \cap \mathcal{F}_{\mathcal{O}} = \emptyset$ and $D(\bar{\mathbf{p}}(\zeta_1), \epsilon) \subset \mathcal{E}(\zeta)$. On the other hand, if $\bar{\mathbf{p}}(\zeta_1) \in \partial\mathcal{F}_{\mathcal{O}}$, there exists $\epsilon > 0$ such that $D(\bar{\mathbf{p}}(\zeta_1), \epsilon) \subset \mathcal{E}(\zeta)$. Then we can find a point $\mathbf{x}_1 \in D(\bar{\mathbf{p}}(\zeta_1), \epsilon)$ that is exterior to both $\mathcal{F}_{\mathcal{R}}$ and $\mathcal{F}_{\mathcal{O}}$. As a result, for $\mathbf{X}_1 = \begin{bmatrix}\mathbf{x}_1^T & 1\end{bmatrix}^T$, we have $\mathbf{X}_1^T\bar{\mathbf{M}}(\zeta_1)\mathbf{X}_1 < 0$, $\mathbf{X}_1^T\mathbf{M}_{\mathcal{R}}\mathbf{X}_1 > 0$, and $\mathbf{X}_1^T\mathbf{M}_{\mathcal{O}}\mathbf{X}_1 > 0$. However, because $\zeta \in [0,1]$,

$$\mathbf{X}_1^T\bar{\mathbf{M}}(\zeta_1)\mathbf{X}_1 = (1-\zeta_1)\mathbf{X}_1^T\mathbf{M}_{\mathcal{R}}\mathbf{X}_1 + \zeta_1\mathbf{X}_1^T\mathbf{M}_{\mathcal{O}}\mathbf{X}_1 > 0, \qquad (3.36)$$

which is absurd. Therefore, $h(\zeta) = 0$ has a real root in $(0,1)$ if the two ellipses are interior-disjoint.

Now for the "only if" part, which also be proved by contradiction on the assumption that the two ellipses $\mathcal{F}_{\mathcal{R}}$ and $\mathcal{F}_{\mathcal{O}}$ overlap. To begin with, let $\mathbf{x}_0$ denote a common interior point of $\mathcal{F}_{\mathcal{R}}$ and $\mathcal{F}_{\mathcal{O}}$. Then for $\mathbf{X}_0 = \begin{bmatrix}\mathbf{x}_0^T & 1\end{bmatrix}^T$, we have $\mathbf{X}_0^T\mathbf{M}_{\mathcal{R}}\mathbf{X}_0 < 0$ and $\mathbf{X}_0^T\mathbf{M}_{\mathcal{O}}\mathbf{X}_0 < 0$. Let $\zeta_0 \in (0,1)$ denote a real root of the equation $h(\zeta) = 0$. Then

$$\mathbf{X}_0^T\bar{\mathbf{M}}(\zeta_0)\mathbf{X}_0 = (1-\zeta_0)\mathbf{X}_0^T\mathbf{M}_{\mathcal{R}}\mathbf{X}_0 + \zeta_0\mathbf{X}_0^T\mathbf{M}_{\mathcal{O}}\mathbf{X}_0 < 0. \qquad (3.37)$$

Also, since $\det\bar{\mathbf{M}}(\zeta_0) = 0$, there exists a point $\mathbf{X}_1$ such that $\bar{\mathbf{M}}(\zeta_0)\mathbf{X}_1 = 0$. Consider a line passing through the points $\mathbf{X}_0$ and $\mathbf{X}_1$. Because the regions $\mathcal{F}_{\mathcal{R}}$ and $\mathcal{F}_{\mathcal{O}}$ are bounded, there exists a point $\mathbf{X}_2$ on the line such that $\mathbf{X}_2^T\mathbf{M}_{\mathcal{R}}\mathbf{X}_2 >$

0 and $\mathbf{X}_2^T \mathbf{M}_{\mathcal{O}} \mathbf{X}_2 > 0$. Then

$$\mathbf{X}_2^T \bar{\mathbf{M}} \left(\zeta_0\right) \mathbf{X}_2 = \left(1 - \zeta_0\right) \mathbf{X}_2^T \mathbf{M}_{\mathcal{R}} \mathbf{X}_2 + \zeta_0 \mathbf{X}_2^T \mathbf{M}_{\mathcal{O}} \mathbf{X}_2 > 0. \tag{3.38}$$

However, since $\mathbf{X}_2$ can be written as $\mathbf{X}_2 = c_0 \mathbf{X}_0 + c_1 \mathbf{X}_1$ for $c_0, c_1 \in \mathbb{R}$ such that $c_0 + c_1 = 1$ and $\bar{\mathbf{M}} \left(\zeta_0\right) \mathbf{X}_1 = 0$, we get

$$\begin{aligned}\mathbf{X}_2^T \bar{\mathbf{M}} \left(\zeta_0\right) \mathbf{X}_2 &= c_0^2 \mathbf{X}_0^T \bar{\mathbf{M}} \left(\zeta_0\right) \mathbf{X}_0 + 2 c_0 c_1 \mathbf{X}_0^T \bar{\mathbf{M}} \left(\zeta_0\right) \mathbf{X}_1 + c_1^2 \mathbf{X}_1^T \bar{\mathbf{M}} \left(\zeta_0\right) \mathbf{X}_1 \\ &= c_0^2 \mathbf{X}_0^T \bar{\mathbf{M}} \left(\zeta_0\right) \mathbf{X}_0 < 0. \end{aligned} \tag{3.39}$$

This is a contradiction. Therefore, the two ellipses do not overlap if $h\left(\zeta\right) = 0$ has a real root in $(0, 1)$.

This completes the proof. □

From Lemma 3.7, the problem of collision detection between two ellipses has been shown to be equivalent to the problem of determining the signs of the roots of the characteristic polynomial. Since the sign behavior of the real roots depends on the coefficients of the polynomial, it is possible to know under what conditions it has two positive and one negative real roots without solving the equation $g\left(\xi\right) = 0$.

To this end, a sign counting function referred to as in [49, 50] is first defined.

Definition 3.8 (Sign Counting Function) *Let P and Q be polynomials with $\deg\left(P\right) = p$ for $p \in \mathbb{N}$. For $\epsilon \in \{-1, 0, 1\}$, the number of the sign ϵ of the evaluation of Q on the real roots of P is defined as*

$$c_\epsilon\left(P; Q\right) = \operatorname{card}\left(\left\{\, \alpha \in \mathbb{R} \middle|\, P\left(\alpha\right) = 0, \operatorname{sgn}\left(Q\left(\alpha\right)\right) = \epsilon\right\}\right), \tag{3.40}$$

where card (X) *is the cardinal number of a set* X *and* sgn (x) *is the signum function of a real number* x *defined as*

$$\operatorname{sgn}(x) = \begin{cases} -1 & \textit{if } x < 0, \\ 0 & \textit{if } x = 0, \\ 1 & \textit{if } x > 0. \end{cases} \tag{3.41}$$

According to the above definition, Lemma 3.7 is transformed into the following corollary.

Corollary 3.9 *Consider two ellipses* $\mathcal{F}_{\mathcal{R}}$ *and* $\mathcal{F}_{\mathcal{O}}$ *defined by* (3.15), (3.17), *and* (3.18) *and their characteristic polynomial* $g(\xi)$ *expressed as* (3.20). *Then the two ellipses are interior-disjoint if and only if*

$$c_1(g(\xi);\xi) = 1 \textit{ or } 2, \tag{3.42}$$

$$c_0(g(\xi);\xi) = 0, \tag{3.43}$$

$$c_{-1}(g(\xi);\xi) = 1. \tag{3.44}$$

If the characteristic polynomial $g(\xi)$ *has a positive double root,* $c_1(g(\xi);\xi) = 1$*. Otherwise,* $c_1(g(\xi);\xi) = 2$*.*

A sequence of polynomials introduced in [49, 50] and called the Sturm-Habicht sequence has been used to calculate the number of possible positive, zero, and negative real roots of a given polynomial P. To explain the relationship between the real roots of P and the polynomials in its Strum-Habicht sequence, the concept of the polynomial subresultant mentioned in [49] is first introduced as follows:

Definition 3.10 (Polynomial Subresultant) *Let P and Q be polynomials with real coefficients such that, for $p, q \in \mathbb{N}$,*

$$P = \sum_{k=0}^{p} a_k \xi^k, \tag{3.45}$$

$$Q = \sum_{k=0}^{q} b_k \xi^k. \tag{3.46}$$

If $i \in \{0, \cdots, \min(p,q)\}$, the polynomial subresultant associated with P, p, Q, and q of index i is defined as follows:

$$\mathrm{Sres}_i(P, p, Q, q) = \sum_{j=0}^{i} M_j^i(P, Q)\, \xi^j, \tag{3.47}$$

where $M_j^i(P,Q)$ is the determinant of the matrix constructed from the columns 1, 2, $\cdots$, $p+q-2i-1$, *and $p+q-i-j$ in the matrix*

$$\mathbf{M}_i(P, p, Q, q) = \overbrace{\begin{bmatrix} a_p & \cdots & a_0 & & \\ & \ddots & \ddots & & \\ & & a_p & \cdots & a_0 \\ b_q & \cdots & b_0 & & \\ & \ddots & \ddots & & \\ & & b_q & \cdots & b_0 \end{bmatrix}}^{p+q-i} \tag{3.48}$$

Hence, the subresultant is a polynomial whose coefficients are determinants of matrices made from the coefficients of the polynomials P and Q. With a sequence of the subresutlants, the Sturm-Habicht sequence [49] associated with P and Q is defined as follows:

Definition 3.11 (Sturm-Habicht Sequence) *Let P and Q be polynomials with degrees p and q, respectively, defined by* (3.45) *and* (3.46)*. Let us define $v = p + q - 1$ and*

$$\delta_k = (-1)^{\frac{k(k+1)}{2}} \tag{3.49}$$

for every $k \in \mathbb{Z}$. Then the Sturm-Habicht sequence associated to the polynomials P and Q is defined as the list of polynomials $\{\mathrm{StHa}_j(P,Q)\}_{j=0}^{v-1}$ where $\mathrm{StHa}_{v+1}(P,Q) = P$, $\mathrm{StHa}_v(P,Q) = P'Q$ and for every $j \in \{0, \cdots, v-1\}$:

$$\mathrm{StHa}_j(P,Q) = \delta_{v-j}\mathrm{Sres}_j\left(P, v+1, P'Q, v\right). \tag{3.50}$$

For every $j \in \{0, \cdots, v+1\}$, the principal j^{th} Sturm-Habicht coefficient is defined as:

$$\mathrm{stha}_j(P,Q) = \mathrm{coef}_j(\mathrm{StHa}_j(P,Q)), \tag{3.51}$$

where $\mathrm{coef}_k(P)$ is the coefficient of ξ^k in P defined as

$$\mathrm{coef}_k(P) = \left.\frac{d^kP}{d\xi^k}\right|_{x=0}. \tag{3.52}$$

The next definitions are needed to explain how the Sturm-Habicht sequence of P and Q are exploited to compute the number of real zeros of P.

Definition 3.12 (the Number of Sign Variations) *Let $\{a_i\}_{i=0}^n$ be a finite sequence of nonzero real numbers. Then the number of sign variations in the sequence is*

$$\mathrm{sv}(\{a_i\}_{i=0}^n) = \sum_{i=1}^n H(-a_{i-1}a_i), \tag{3.53}$$

where $H(x)$ is the unit step function such that $H(x) = (1 + \mathrm{sgn}(x))/2$.

Definition 3.13 (the Number of Sign Permanences) *Let $\{a_i\}_{i=0}^n$ be a finite sequence of nonzero real numbers. Then the number of sign permanences in the sequence is defined as follows:*

$$\operatorname{sp}\left(\{a_i\}_{i=0}^n\right) = \sum_{i=1}^{n} H\left(a_{i-1}a_i\right). \tag{3.54}$$

Definition 3.14 *Let $\{a_i\}_{i=0}^n$ be a finite sequence of real numbers such that $a_0 \neq 0$. By marking all the zero elements of the sequence, it can be written as follows:*

$$\begin{aligned}\{a_i\}_{i=0}^n =&\{a_0, \cdots, a_{i_1}, \overbrace{0, \cdots, 0}^{k_1}, a_{i_1+k_1+1}, \cdots, a_{i_2}, \overbrace{0, \cdots, 0}^{k_2}, a_{i_2+k_2+1}, \cdots, a_{i_3},\\ &0, \cdots\cdots, 0, a_{i_{m-1}+k_{m-1}+1}, \cdots, a_{i_m}, \overbrace{0, \cdots, 0}^{k_m}\},\end{aligned} \tag{3.55}$$

where $i_m + k_m = n$. Let $C\left(\{a_i\}_{i=0}^n\right)$ be a function defined as

$$\begin{aligned}C\left(\{a_i\}_{i=0}^n\right) =&\sum_{j=1}^{m} \operatorname{sp}\left(\{a_i\}_{i=i_{j-1}+k_{j-1}+1}^{i_j}\right) - \sum_{j=1}^{m} \operatorname{sv}\left(\{a_i\}_{i=i_{j-1}+k_{j-1}+1}^{i_j}\right)\\ &+ \sum_{j=1}^{m-1} \epsilon_{i_j},\end{aligned} \tag{3.56}$$

where $i_0 + k_0 + 1 = 0$ and

$$\epsilon_{i_j} = \begin{cases} 0, & \text{if } k_j \text{ is odd,} \\ (-1)^{\frac{k_j}{2}} \operatorname{sgn}\left(\frac{a_{i_j+k_j+1}}{a_{i_j}}\right), & \text{if } k_j \text{ is even.} \end{cases} \tag{3.57}$$

The following lemma shows that the number of sign variations and permanences in the Sturm-Habicht principal coefficients associated to the polynomials

P and Q has a close relation to the number of real roots of P.

Lemma 3.15 *Let P and Q be polynomials with* $\deg(P) = p$ *for* $p \in \mathbb{N}$. *Then*

$$C\left(\left\{\mathrm{stha}_p(P,Q), \cdots, \mathrm{stha}_0(P,Q)\right\}\right) = c_1(P;Q) - c_{-1}(P;Q). \tag{3.58}$$

The proof of Lemma 3.15 can be found in [50]. From the above lemma, the following corollary and lemma can be derived.

Corollary 3.16 *Consider two ellipses $\mathcal{F}_{\mathcal{R}}$ and $\mathcal{F}_{\mathcal{O}}$ defined by* (3.15), (3.17), *and* (3.18) *and their characteristic polynomial $g(\xi)$. Then*

$$C\left(\left\{\mathrm{stha}_3(g(\xi),\xi), \cdots, \mathrm{stha}_0(g(\xi),\xi)\right\}\right) = c_1(g(\xi);\xi) - c_{-1}(g(\xi);\xi). \tag{3.59}$$

Lemma 3.17 *Consider two ellipses $\mathcal{F}_{\mathcal{R}}$ and $\mathcal{F}_{\mathcal{O}}$ defined by* (3.15), (3.17), *and* (3.18) *and their characteristic polynomial $g(\xi)$. Then*

$$C\left(\left\{\mathrm{stha}_3(g(\xi),1), \cdots, \mathrm{stha}_0(g(\xi),1)\right\}\right) = c_1(g(\xi);\xi) + c_{-1}(g(\xi);\xi). \tag{3.60}$$

Proof. Put $P = g(\xi)$ and $Q = 1$ in (3.60), so that

$$C\left(\left\{\mathrm{stha}_3(g(\xi),1), \cdots, \mathrm{stha}_0(g(\xi),1)\right\}\right) = c_1(g(\xi);1) - c_{-1}(g(\xi);1). \tag{3.61}$$

Since $c_1(g(\xi);1)$ is the number of the real roots of the characteristic polynomial $g(\xi)$ and $c_{-1}(g(\xi);1) = 0$, we get $c_1(g(\xi);1) = c_1(g(\xi);\xi) + c_0(g(\xi);\xi) + c_{-1}(g(\xi);\xi)$. This completes the proof because $c_0(g(\xi);\xi) = 0$. □

Accordingly, the condition of the coefficients of the characteristic polynomial that has two positive and one negative real roots is described as follows:

Theorem 3.18 *Consider two ellipses $\mathcal{F}_{\mathcal{R}}$ and $\mathcal{F}_{\mathcal{O}}$ defined by* (3.15), (3.17), *and* (3.18) *and their characteristic polynomial $g(\xi)$ expressed as* (3.20). *Then the two ellipses are interior-disjoint if and only if*

$$\begin{cases} a_2 \geq 0 \quad \textit{or} \quad 3a_3a_2a_0 - 4a_3a_1^2 + a_2^2a_1 \geq 0, \\ 3a_3a_1 - a_2^2 < 0, \\ 27a_3^2a_0^2 - 18a_3a_2a_1a_0 + 4a_3a_1^3 + 4a_2^3a_0 - a_2^2a_1^2 \leq 0. \end{cases} \tag{3.62}$$

Proof. From Corollaries 3.9 and 3.16 and Lemma 3.17, the necessary and sufficient condition that the two ellipses are touching externally or separated is either

$$C\left(\left\{\mathrm{stha}_3\left(g\left(\xi\right),1\right),\cdots,\mathrm{stha}_0\left(g\left(\xi\right),1\right)\right\}\right) = 3, \tag{3.63}$$

$$C\left(\left\{\mathrm{stha}_3\left(g\left(\xi\right),\xi\right),\cdots,\mathrm{stha}_0\left(g\left(\xi\right),\xi\right)\right\}\right) = 1, \tag{3.64}$$

or

$$C\left(\left\{\mathrm{stha}_3\left(g\left(\xi\right),1\right),\cdots,\mathrm{stha}_0\left(g\left(\xi\right),1\right)\right\}\right) = 2, \tag{3.65}$$

$$C\left(\left\{\mathrm{stha}_3\left(g\left(\xi\right),\xi\right),\cdots,\mathrm{stha}_0\left(g\left(\xi\right),\xi\right)\right\}\right) = 0. \tag{3.66}$$

To expand the left-side hands of the above equations, the Strum-Habicht sequence associated to $g(\xi) = a_3\xi^3 + a_2\xi^2 + a_1\xi + a_0$ is expressed as

$$\mathrm{StHa}_3\left(g,1\right) = a_3\xi^3 + a_2\xi^2 + a_1\xi + a_0, \tag{3.67}$$

$$\mathrm{StHa}_2\left(g,1\right) = 3a_3\xi^2 + 2a_2\xi + a_1, \tag{3.68}$$

$$\mathrm{StHa}_1\left(g,1\right) = -2a_3\left(3a_3a_1 - a_2^2\right)\xi + a_3\left(a_2a_1 - 9a_3a_0\right), \tag{3.69}$$

Table 3.1 All the possible sign combinations of the sequence of the principal Sturm-Habicht coefficients associated to $g(\xi)$ and the corresponding results

$sgn\left(\{\mathrm{stha}_i\,(g,1)\}_{i=3}^{0}\right)$	$\sum sp\,(\cdot)$	$\sum sv\,(\cdot)$	$\sum \epsilon_{i_j}$	$c_1\,(g\,(\xi)\,;1)$
$\{-1,-1,-1,-1\}$	3	0	0	**3**
$\{-1,-1,-1,\;\;0\}$	2	0	0	**2**
$\{-1,-1,-1,\;\;1\}$	2	1	0	1
$\{-1,-1,\;\;0,-1\}$	1	0	0	1
$\{-1,-1,\;\;0,\;\;0\}$	1	0	0	1
$\{-1,-1,\;\;0,\;\;1\}$	1	0	0	1
$\{-1,-1,\;\;1,-1\}$	1	2	0	-1
$\{-1,-1,\;\;1,\;\;0\}$	1	1	0	0
$\{-1,-1,\;\;1,\;\;1\}$	2	1	0	1

$$\mathrm{StHa}_0\,(g,1) = -a_3\left(27a_3^2a_0^2 - 18a_3a_2a_1a_0 + 4a_3a_1^3 + 4a_2^3a_0 - a_2^2a_1^2\right). \qquad (3.70)$$

Hence, the principal Sturm-Habicht coefficients associated to $g(\xi)$ are

$$\mathrm{stha}_3\,(g,1) = a_3, \qquad (3.71)$$

$$\mathrm{stha}_2\,(g,1) = 3a_3, \qquad (3.72)$$

$$\mathrm{stha}_1\,(g,1) = -2a_3\left(3a_3a_1 - a_2^2\right), \qquad (3.73)$$

$$\mathrm{stha}_0\,(g,1) = -a_3\left(27a_3^2a_0^2 - 18a_3a_2a_1a_0 + 4a_3a_1^3 + 4a_2^3a_0 - a_2^2a_1^2\right). \qquad (3.74)$$

Because the $a_3 < 0$, all the possible sign combinations of the coefficients sequence $\{\mathrm{stha}_i\,(g,1)\}_{i=3}^{0} = \{\mathrm{stha}_3\,(g\,(\xi)\,,1)\,,\cdots,\mathrm{stha}_0\,(g\,(\xi)\,,1)\}$ are presented in Table 3.1. As a result, $g(\xi)$ has three different real roots if and only if

$$\begin{cases} 3a_3a_1 - a_2^2 < 0, \\ 27a_3^2a_0^2 - 18a_3a_2a_1a_0 + 4a_3a_1^3 + 4a_2^3a_0 - a_2^2a_1^2 < 0. \end{cases} \qquad (3.75)$$

In addition, it has a double real root and another distinct single real root if and only if

$$\begin{cases} 3a_3a_1 - a_2^2 < 0, \\ 27a_3^2a_0^2 - 18a_3a_2a_1a_0 + 4a_3a_1^3 + 4a_2^3a_0 - a_2^2a_1^2 = 0. \end{cases} \tag{3.76}$$

On the other hand, the Strum-Habicht sequence associated to $g\left(\xi\right)$ and ξ is written as

$$\mathrm{StHa}_4\left(g,\xi\right) = a_3\xi^3 + a_2\xi^2 + a_1\xi + a_0, \tag{3.77}$$

$$\mathrm{StHa}_3\left(g,\xi\right) = 3a_3\xi^3 + 2a_2\xi^2 + a_1\xi, \tag{3.78}$$

$$\mathrm{StHa}_2\left(g,\xi\right) = -3a_3^2a_2\xi^2 - 6a_3^2a_1\xi - 9a_3^2a_0, \tag{3.79}$$

$$\mathrm{StHa}_1\left(g,\xi\right) = 3a_3^2\left(3a_3a_2a_0 - 4a_3a_1^2 + a_2^2a_1\right)\xi - 6a_3^2a_0\left(3a_3a_1 - a_2^2\right), \tag{3.80}$$

$$\mathrm{StHa}_0\left(g,\xi\right) = 3a_3^2a_0\left(27a_3^2a_0^2 - 18a_3a_2a_1a_0 + 4a_3a_1^3 + 4a_2^3a_0 - a_2^2a_1^2\right). \tag{3.81}$$

Therefore, the principal Sturm-Habicht coefficients associated to $g\left(\xi\right)$ and ξ are

$$\mathrm{stha}_4\left(g,\xi\right) = 0, \tag{3.82}$$

$$\mathrm{stha}_3\left(g,\xi\right) = 3a_3, \tag{3.83}$$

$$\mathrm{stha}_2\left(g,\xi\right) = -3a_3^2a_2, \tag{3.84}$$

$$\mathrm{stha}_1\left(g,\xi\right) = 3a_3^2\left(3a_3a_2a_0 - 4a_3a_1^2 + a_2^2a_1\right), \tag{3.85}$$

$$\mathrm{stha}_0\left(g,\xi\right) = 3a_3^2a_0\left(27a_3^2a_0^2 - 18a_3a_2a_1a_0 + 4a_3a_1^3 + 4a_2^3a_0 - a_2^2a_1^2\right). \tag{3.86}$$

First, the case that $g\left(\xi\right) = 0$ has three different roots is considered. Because $a_3 < 0$, $a_0 < 0$, and $27a_3^2a_0^2 - 18a_3a_2a_1a_0 + 4a_3a_1^3 + 4a_2^3a_0 - a_2^2a_1^2 < 0$, we have $\mathrm{stha}_3\left(g,\xi\right) < 0$ and $\mathrm{stha}_0\left(g,\xi\right) > 0$. Hence, all the possible sign combinations

Table 3.2 All the possible sign combinations of $\{\mathrm{stha}_i\,(g,\xi)\}_{i=3}^{0}$ when $g\,(\xi)$ has three different real roots and the corresponding results

$sgn\left(\{\mathrm{stha}_i\,(g,\xi)\}_{i=3}^{0}\right)$	$\sum sp\,(\cdot)$	$\sum sv\,(\cdot)$	$\sum \epsilon_{i_j}$	$C\left(\{\mathrm{stha}_i\,(g,\xi)\}_{i=3}^{0}\right)$
$\{-1,-1,-1,\ \ 1\}$	2	1	0	**1**
$\{-1,-1,\ \ 0,\ \ 1\}$	1	0	0	**1**
$\{-1,-1,\ \ 1,\ \ 1\}$	2	1	0	**1**
$\{-1,\ \ 0,-1,\ \ 1\}$	0	1	0	-1
$\{-1,\ \ 0,\ \ 0,\ \ 1\}$	0	0	1	**1**
$\{-1,\ \ 0,\ \ 1,\ \ 1\}$	1	0	0	**1**
$\{-1,\ \ 1,-1,\ \ 1\}$	0	3	0	-3
$\{-1,\ \ 1,\ \ 0,\ \ 1\}$	0	1	0	-1
$\{-1,\ \ 1,\ \ 1,\ \ 1\}$	2	1	0	**1**

of the coefficients sequence $\{\mathrm{stha}_3\,(g\,(\xi)\,,\xi)\,,\cdots,\mathrm{stha}_0\,(g\,(\xi)\,,\xi)\}$ are presented in Table 3.2, where the cases that $C\,(\{\mathrm{stha}_3\,(g\,(\xi)\,,\xi)\,,\cdots,\mathrm{stha}_0\,(g\,(\xi)\,,\xi)\}) = 1$ are highlighted in yellow.

However, since the product of the three roots of the polynomial $g\,(\xi)$ is less than 0, there does not exist $\{\mathrm{stha}_3\,(g\,(\xi)\,,\xi)\,,\cdots,\mathrm{stha}_0\,(g\,(\xi)\,,\xi)\}$ such that $C\,(\{\mathrm{stha}_3\,(g\,(\xi)\,,\xi)\,,\cdots,\mathrm{stha}_0\,(g\,(\xi)\,,\xi)\}) = -1$, which means the product is greater than 0. Thus the 4th and 8th cases in Table 3.2, emphasized in red, can be included in any sets to find the simplest possible form for the expression.

Consequently, the characteristic polynomial has two different positive and one negative roots if and only if (3.75) and

$$a_2 \geq 0 \quad \text{or} \quad 3a_3a_2a_0 - 4a_3a_1^2 + a_2^2a_1 \geq 0 \tag{3.87}$$

hold.

Next, the case that $g\,(\xi) = 0$ has a double root is considered. Because $a_3 < 0$, $a_0 < 0$, and $27a_3^2a_0^2 - 18a_3a_2a_1a_0 + 4a_3a_1^3 + 4a_2^3a_0 - a_2^2a_1^2 = 0$, we

Table 3.3 All the possible sign combinations of $\{\text{stha}_i\left(g,\xi\right)\}_{i=3}^{0}$ when $g\left(\xi\right)$ has a double real root and another distinct single one and the corresponding results

$sgn\left(\{\text{stha}_i\left(g,\xi\right)\}_{i=3}^{0}\right)$	$\sum sp\left(\cdot\right)$	$\sum sv\left(\cdot\right)$	$\sum \epsilon_{i_j}$	$C\left(\{\text{stha}_i\left(g,\xi\right)\}_{i=3}^{0}\right)$
$\{-1,-1,-1,\ \ 0\}$	2	0	0	2
$\{-1,-1,\ \ 0,\ \ 0\}$	1	0	0	1
$\{-1,-1,\ \ 1,\ \ 0\}$	1	1	0	**0**
$\{-1,\ \ 0,-1,\ \ 0\}$	0	0	0	**0**
$\{-1,\ \ 0,\ \ 0,\ \ 0\}$	0	0	0	**0**
$\{-1,\ \ 0,\ \ 1,\ \ 0\}$	0	0	0	**0**
$\{-1,\ \ 1,-1,\ \ 0\}$	0	2	0	-2
$\{-1,\ \ 1,\ \ 0,\ \ 0\}$	0	1	0	-1
$\{-1,\ \ 1,\ \ 1,\ \ 0\}$	1	1	0	**0**

have $\text{stha}_3\left(g,\xi\right) < 0$ and $\text{stha}_0\left(g,\xi\right) = 0$. All the possible sign combinations of the coefficients sequence $\{\text{stha}_3\left(g\left(\xi\right),\xi\right),\cdots,\text{stha}_0\left(g\left(\xi\right),\xi\right)\}$ are presented in Table 3.3, where the cases that $C\left(\{\text{stha}_3\left(g\left(\xi\right),\xi\right),\cdots,\text{stha}_0\left(g\left(\xi\right),\xi\right)\}\right) = 0$ are highlighted in yellow.

However, there does not exist a sign combination of $\{-1,-1,-1,0\}$ from Lemma 3.6. In addition, the combinations of the Sturm-Habicht principal coefficients that satisfy $C\left(\{\text{stha}_3\left(g\left(\xi\right),\xi\right),\cdots,\text{stha}_0\left(g\left(\xi\right),\xi\right)\}\right) = \pm 1$ are redundant since the number of real roots cannot be a rational number. Hence, the characteristic polynomial has a positive double root and a negative root if and only if (3.76) and (3.87) hold.

This completes the proof. □

Meanwhile, it was mentioned in the preceding section that the collision-free orientations of the robot would be derived in this section when the relative position of the obstacle with respect to the robot is given. This can be done by substituting (3.21) – (3.30) into the four inequalities in (3.62). If we put

$\cos 2\theta_{\mathcal{R}} = \left(1 - t^2\right) / \left(1 + t^2\right)$ and $\sin 2\theta_{\mathcal{R}} = 2t / \left(1 + t^2\right)$, the polynomial inequalities of degree 2, 6, 4, and 8 are obtained in regular sequence. Theses inequalities can be solved by finding the roots of the polynomials. Moreover, this algebraic condition can be utilized to predict whether an elliptic robot under arbitrary motion collides with obstacles by representing the position and orientation of ellipses as a function of time.

3.5 Summary

This chapter has introduced two different frameworks to address the problem of collision detection between an elliptic robot and an elliptic obstacle. In Section 3.3, the collision space obstacle has been defined, and its region has been identified by finding the locus of its boundary points. It also has been shown that whether the rotation of the robot causes a collision depends on the relative position of the obstacle with respect to the robot, which inspires the key idea of the next chapter that the elliptic robot can be regarded as a circular robot with radius equal to its minor radius by adequately controlling its orientation.

In Section 3.4, an algebraic condition for the interior-disjoint of the robot and the obstacle is clarified on the basis of the relation between Sturm-Habicht sequences and real roots of the characteristic polynomial. It was shown that the condition is obtained from the separation condition by replacing three of four strict inequalities to their corresponding non-strict ones by following the proofs in [23] and [36]. As a result, the robot can determine whether its nonlinear motion causes a collision in the near future, which is described in detail in the next chapter.

Chapter 4

Obstacle Avoidance

4.1 Introduction

In the velocity-based motion planning, a robot is assumed to receive a control command containing its new linear and angular velocities for the next sampling period. Because this command determines whether the robot collides with other robots and obstacles, it is important to find out under what conditions the collisions occur. By selecting the new velocities that do not satisfying those conditions, the robot can avoid the collisions and reach the goal safely.

To determine the sufficient and necessary condition for the collision between two moving objects, the following two assumptions are necessary.

- The one can measure the shape, position, and velocity of the other.
- The one can predict the motion of the other in the near future.

As described in Section 1.1, an elliptic robot $\mathcal{R}$ can measure the position and the velocity of obstacles in its detection range and estimate their shape by gathering the sensor data. The robot also predicts the motion of the obstacles based on their current position and velocity. As a result, the collision condition between a robot and an obstacle can be calculated. However, things are different when the collision between robots is considered. Since robots change their behavior based on input from their environment, it is very difficult for a robot to predict the motion of another robot. In other words, the collision between robots does not satisfy the second assumption. Therefore, the collision between a robot and obstacles is only addressed in this chapter.

When it comes to the collision-free motion planning, the smaller a robot is, the more efficiently it can avoid obstacles. For example, a compact car can take a shortcut through a narrow street, whereas a cargo truck has to take a long way around. For an anisotropic and holonomic robot, it is possible to produce the same effect as the size of the robot decreases because the distance from the robot to obstacles depends on its orientation. If the orientation of the robot were able to change instantly, an elliptic robot would be able to follow the same path as a circular robot with radius equal to the minor radius moves to avoid an obstacle. Therefore, the objective of this chapter is to suggest a framework that an elliptic robot efficiently avoids collisions with obstacles with rotation.

This chapter is organized as follows. In Section 4.2, the problem that a robot avoids collisions with obstacles is explained, and how to divide it into two subproblems, linear and angular motion planning, is described. Section 4.3 is a preliminary section that derives tangent line equations of the C-obstacle in Section 3.3 and finds the closest point on the boundary of C-obstacle to a given point. Section 4.4 and 4.5 deal with the first subproblem. The sufficient and necessary condition for the collision between a non-rotating elliptic robot and an obstacle, called the Ellipse-based Velocity Obstacle (EBVO), is derived in Section 4.4, and a strategy for how to select the new linear velocity of the robot based on the EBVO is elucidated in Section 4.5. On the other hand, Section 4.6 and 4.7 address the second subproblem in a similar way described in the preceding sections. The interval of collision-free rotation angles of an elliptic robot moving with a constant linear velocity is derived in Section 4.6, and a method to select the new angular velocity from the interval is presented in Section 4.7. Finally, the proposed approach is summarized in Section 4.8.

4.2 Problem Formulation and Approach

In this chapter, the problem of real-time local collision avoidance for an elliptic robot from obstacles is considered, which is a subproblem of Problem 1.1.

Let $\mathcal{R}$ be an elliptic robot in a planar workspace $\mathcal{W}$. The robot has major and minor radii of $M_{\mathcal{R}}$ and $m_{\mathcal{R}}$, and its shape is characterized by a symmetric positive definite matrix $\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right) \in \mathbb{R}^{2\times 2}$. It also has a current position $\mathbf{p}_{\mathcal{R}}$, orientation $\theta_{\mathcal{R}}$, linear velocity $\mathbf{v}_{\mathcal{R}}$, and angular velocity $w_{\mathcal{R}}$. In addition, its mobility is limited by the dynamic constraints such as the maximum linear speed $v_{\mathcal{R}}^{max}$, angular speed $w_{\mathcal{R}}^{max}$, linear acceleration $a_{\mathcal{R}}^{max}$, and angular acceleration $\alpha_{\mathcal{R}}^{max}$. The robot $\mathcal{R}$ equipped with a range sensor detects the set of obstacles $\{\mathcal{O}_1, \mathcal{O}_2, \cdots, \mathcal{O}_{N_{\mathcal{O}}}\}$ and measures their shape matrix $\mathbf{S}_{\mathcal{O}_j}$, position $\mathbf{p}_{\mathcal{O}_j}$, and velocity $\mathbf{v}_{\mathcal{O}_j}$ for $j = 1, \cdots, N_{\mathcal{O}}$.

The objective of the robot is to reach the goal position $\mathbf{p}_{\mathcal{R}}^{goal}$ given by an external global planner through the shortest path possible without collisions. Because a discrete-time robot model is employed, the robot selects its new linear velocity $\mathbf{v}_{\mathcal{R}}^{new}$ and angular velocity $w_{\mathcal{R}}^{new}$ on the basis of the current positions and velocities of itself and the observed obstacles at each time step. As a consequence, the problem to be solved in this chapter is defined as follows.

Problem 4.1 (Obstacle Avoidance of an Elliptic Robot) *For an elliptic robot $\mathcal{R}$, compute the new linear velocity $\mathbf{v}_{\mathcal{R}}^{new}$ and angular velocity $w_{\mathcal{R}}^{new}$ in order to generate a trajectory to the goal that circumvents obstacles with the minimum possible deviation from*

- *the measurements of obstacles in its detection area D,*
- *the robot's goal position $\mathbf{p}_{\mathcal{R}}^{goal}$.*

However, it is infeasible to find both the optimal $\mathbf{v}_{\mathcal{R}}^{new}$ and $w_{\mathcal{R}}^{new}$ at the same time with a high enough sampling rate to be used for practical real-time applications. Although Section 4.3 derives the boundary equation of 3D C-space by convolving the footprints of obstacles and the robot at each orientation, it is difficult to determine whether the robot is inside the C-obstacle or not. For this reason, it takes much computation time to consider both the velocities at the same time. Consequently, we decompose it into two subproblems: Problem 4.2 and Problem 4.3.

Problem 4.2 (Obstacle Avoidance of an Elliptic Robot Capable of Only Translational Motion) *Identical to Problem 4.1 except that $w_{\mathcal{R}}^{new} = 0$ is assumed.*

Problem 4.3 (Angular Motion Control for an Elliptic Robot Circumventing Obstacles) *Compute the new angular velocity $w_{\mathcal{R}}^{new}$ of an elliptic robot $\mathcal{R}$ moving at $\mathbf{v}_{\mathcal{R}}$ in order to potentially circumvent an obstacle interrupting the robot's path to the goal with the minimum deviation without collisions.*

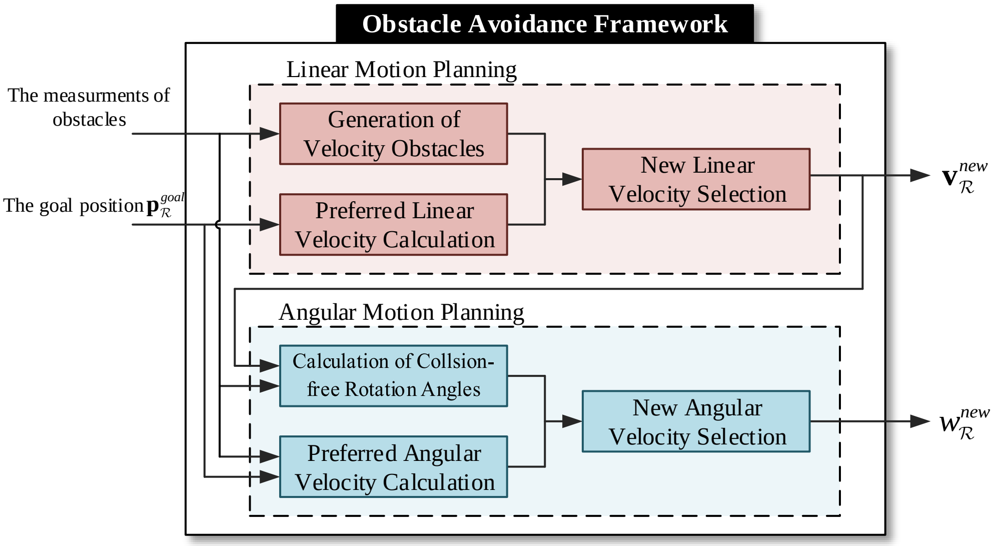

Obstacle Avoidance Framework
Linear Motion Planning
The measurments of obstacles
The goal position $\mathbf{p}_{\mathcal{R}}^{goal}$
Generation of Velocity Obstacles
Preferred Linear Velocity Calculation
New Linear Velocity Selection
$\mathbf{v}_{\mathcal{R}}^{new}$
Angular Motion Planning
Calculation of Collsion-free Rotation Angles
Preferred Angular Velocity Calculation
New Angular Velocity Selection
$w_{\mathcal{R}}^{new}$

Figure 4.1 The proposed obstacle avoidance framework for an elliptic robot.

The linear motion is determined in Problem 4.2, and the angular one is decided in Problem 4.3. At each time step, the two problems are sequentially solved. Hence, the solution of Problem 4.2 is assigned to $\mathbf{v}_{\mathcal{R}}$ in Problem 4.3.

The proposed obstacle avoidance framework for an elliptic robot is shown in Fig. 4.1. The velocity obstacles are generated from the measurements of obstacles in the detection area, and the preferred linear velocity is calculated from the goal position given by its global motion planner. The new linear velocity is selected based on the two data. In the angular motion planning, the collision free rotation angles are computed from the measurements and the goal position. To calculate the preferred angular velocities, both the measurement and the goal position are necessary since they enable the robot circumvent obstacles with the minimum deviation from the straight-line path to the goal. Afterward, the new angular velocity is also determined.

4.3 Preliminaries: Properties of C-obstacles for an Elliptic Robot

As a preliminary step for the subsequent sections, the tangent lines to the C-obstacle given a basis vector and a point on the line are computed. In addition, the closest point on the boundary of the C-obstacle given a point is calculated. Because the linear and angular motion planning problems are decomposed in this chapter, the orientation of the robot is assumed to be fixed. Hence, the slice of the C-obstacle for a given orientation $\theta_{\mathcal{R}}$, $QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$ is regarded as the C-obstacle.

4.3.1 Tangent lines to C-obstacle

When an elliptic robot $\mathcal{R}$ adopts the configuration space framework to avoid an elliptic obstacle $\mathcal{O}$ that blocks the robot's straight-line path to the goal, it is efficient for a point, which is the configuration of the robot, in the C-space $\mathcal{Q}$ to move tangent to the surface of the C-obstacle $QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$. In this subsection, the tangent lines to the C-obstacle are derived under two different conditions that a basis vector of the line is given and that a point on the line is given, as shown in Fig. 4.2.

First of all, put $\mathbf{u} = \begin{bmatrix}\cos\phi & \sin\phi\end{bmatrix}^T$ in (3.3). Then we obtain the tangent vector to the C-obstacle $QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$ at a point $\mathbf{q}_{\mathbf{x},0} \in QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$ by taking the derivative of (3.3) with respect to ϕ:

$$\left.\frac{d\mathbf{q}_{\mathbf{x}}}{d\phi}\right|_{\mathbf{q}_{\mathbf{x}}=\mathbf{q}_{\mathbf{x},0}} = \left(\mathbf{S}_{\mathcal{O}}^{\frac{1}{2}} + k\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\right)\left.\frac{d\mathbf{u}}{d\phi}\right|_{\mathbf{u}=\mathbf{u}_0} + \left.\frac{dk}{d\phi}\right|_{\mathbf{u}=\mathbf{u}_0}\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_0. \tag{4.1}$$

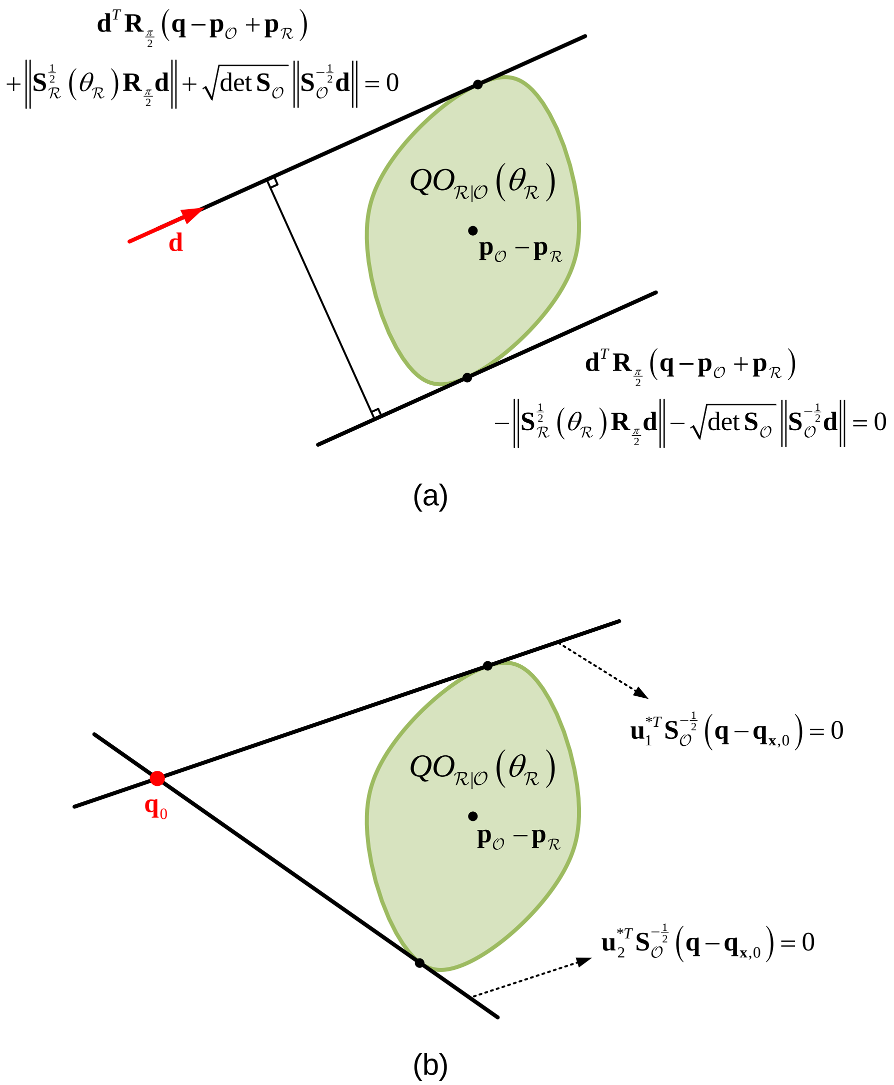

$\mathbf{d}^T\mathbf{R}_{\frac{\pi}{2}}\left(\mathbf{q}-\mathbf{p}_{\mathcal{O}}+\mathbf{p}_{\mathcal{R}}\right)$
$+\left\|\mathbf{S}_{\mathcal{R}}^{\frac{1}{2}}\left(\theta_{\mathcal{R}}\right)\mathbf{R}_{\frac{\pi}{2}}\mathbf{d}\right\|+\sqrt{\det\mathbf{S}_{\mathcal{O}}}\left\|\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{d}\right\|=0$
$QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$
$\mathbf{d}$
$\mathbf{p}_{\mathcal{O}}-\mathbf{p}_{\mathcal{R}}$
$\mathbf{d}^T\mathbf{R}_{\frac{\pi}{2}}\left(\mathbf{q}-\mathbf{p}_{\mathcal{O}}+\mathbf{p}_{\mathcal{R}}\right)$
$-\left\|\mathbf{S}_{\mathcal{R}}^{\frac{1}{2}}\left(\theta_{\mathcal{R}}\right)\mathbf{R}_{\frac{\pi}{2}}\mathbf{d}\right\|-\sqrt{\det\mathbf{S}_{\mathcal{O}}}\left\|\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{d}\right\|=0$
(a)
$\mathbf{u}_1^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\left(\mathbf{q}-\mathbf{q}_{\mathbf{x},0}\right)=0$
$QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$
$\mathbf{q}_0$
$\mathbf{p}_{\mathcal{O}}-\mathbf{p}_{\mathcal{R}}$
$\mathbf{u}_2^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\left(\mathbf{q}-\mathbf{q}_{\mathbf{x},0}\right)=0$
(b)

Figure 4.2 Tangent lines to the C-obstacle: (a) tangent lines with a given basis vector; (b) tangent lines at a given point.

Put

$$\left.\frac{d\mathbf{u}}{d\phi}\right|_{\mathbf{u}=\mathbf{u}_0} = \mathbf{R}_{\frac{\pi}{2}}\mathbf{u}_0, \tag{4.2}$$

$$\left.\frac{dk}{d\phi}\right|_{\mathbf{u}=\mathbf{u}_0} = -k^3\mathbf{u}_0^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{u}_0, \tag{4.3}$$

where $\mathbf{R}_{\frac{\pi}{2}}$ is the rotation matrix corresponding to a counter-clockwise rotation of the right angle. Then

$$\begin{aligned}\left.\frac{d\mathbf{q}_{\mathbf{x}}}{d\phi}\right|_{\mathbf{q}_{\mathbf{x}}=\mathbf{q}_{\mathbf{x},0}} &= \left(\mathbf{S}_{\mathcal{O}}^{\frac{1}{2}} + k\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\right)\mathbf{R}_{\frac{\pi}{2}}\mathbf{u}_0 \\ &\quad - k^3\left(\mathbf{u}_0^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{u}_0\right)\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_0. \end{aligned} \tag{4.4}$$

Given a basis vector

The derivation is started with the following lemma concerning about the tangent line in a given direction.

Lemma 4.4 *Consider the C-obstacle $QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$ whose boundary is represented by* (3.3). *If a basis vector $\mathbf{d}$ of the tangent line to $QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$ is given, then the parameter $\mathbf{u}^*$ of the contact point $\mathbf{q}_{\mathbf{x}}^* = \mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}} + \mathbf{S}_{\mathcal{O}}^{\frac{1}{2}}\mathbf{u}^* + k\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}^*$ satisfies*

$$\mathbf{u}^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{d} = 0. \tag{4.5}$$

Proof. At the contact point $\mathbf{q}_{\mathbf{x}}^*$, it is satisfied that

$$\left.\frac{d\mathbf{q}_{\mathbf{x}}}{d\phi}\right|_{\mathbf{q}_{\mathbf{x}}=\mathbf{q}_{\mathbf{x}}^*} = m\mathbf{d}, \tag{4.6}$$

where $m \in \mathbb{R} - \{0\}$. Substituting (4.4) into (4.6) and left multiplying $\mathbf{u}^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}$

to both sides of (4.6), we can rewrite the equation as

$$\begin{aligned}\mathbf{u}^{*T}\mathbf{R}_{\frac{\pi}{2}}\mathbf{u}^{*} + k\mathbf{u}^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{u}^{*} \\ -k^{3}\left(\mathbf{u}^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{u}^{*}\right)\left(\mathbf{u}^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}^{*}\right) \\ = m\mathbf{u}^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{d}.\end{aligned} \tag{4.7}$$

Since $\mathbf{u}^{*T}\mathbf{R}_{\frac{\pi}{2}}\mathbf{u}^{*} = 0$ and $k^{-2} = \mathbf{u}^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}^{*}$, the left-hand side of (4.7) equals to 0. Because $m \neq 0$, we arrive at (4.5). □

Lemma 4.5 *Consider the C-obstacle* $QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$ *whose boundary is represented by* (3.3)*. The line tangent to* $QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$ *at a point* $\mathbf{q}_{\mathbf{x}}^{*} \in \partial QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$ *has the form*

$$\mathbf{u}^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\left(\mathbf{q} - \mathbf{p}_{\mathcal{O}} + \mathbf{p}_{\mathcal{R}}\right) - \left\|\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)^{\frac{1}{2}}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}^{*}\right\| - 1 = 0, \tag{4.8}$$

where $\mathbf{u}^{*}$ *is the parameter of* $\mathbf{q}_{\mathbf{x}}^{*}$ *in* (3.3)*.*

Proof. If a direction $\mathbf{d}$ of the tangent line is given, the equation of that line has the form of $\left(\mathbf{d}^{\perp}\right)^{T}\left(\mathbf{q} - \mathbf{q}_{\mathbf{x}}^{*}\right) = 0$. Since $\mathbf{d}^{\perp} = \mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}^{*}$ from (4.5),

$$\mathbf{u}^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\left(\mathbf{q} - \mathbf{q}_{\mathbf{x}}^{*}\right) = 0. \tag{4.9}$$

From (3.3),

$$\mathbf{u}^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\left(\mathbf{q} - \mathbf{p}_{\mathcal{O}} + \mathbf{p}_{\mathcal{R}}\right) - \mathbf{u}^{*T}\mathbf{u}^{*} - k\mathbf{u}^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}^{*T} = 0. \tag{4.10}$$

Since $\mathbf{u}^{*T}\mathbf{u}^{*} = 1$ and $k^{-2} = \mathbf{u}^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}^{*}$, (4.8) holds. □

By Lemmas 4.4 and 4.5, we can derive the equation of the tangent line when its direction is given.

Lemma 4.6 *Consider the C-obstacle $QO_{\mathcal{R}|\mathcal{O}}(\theta_{\mathcal{R}})$ of an elliptic obstacle $\mathcal{O}$ to an elliptic robot $\mathcal{R}$ with orientation $\theta_{\mathcal{R}}$. If a basis vector $\mathbf{d}$ of the tangent line is given, the equation of the line can be expressed as*

$$\pm\mathbf{d}^T\mathbf{R}_{\frac{\pi}{2}}\left(\mathbf{q}-\mathbf{p}_{\mathcal{O}}+\mathbf{p}_{\mathcal{R}}\right)+\left\|\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)^{\frac{1}{2}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{d}\right\|+\sqrt{\det\mathbf{S}_{\mathcal{O}}}\left\|\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{d}\right\|=0. \quad (4.11)$$

Proof. Suppose that the direction vector $\mathbf{d}$ of the tangent line is given. From Lemma 4.4, the parameter $\mathbf{u}^*$ is given by

$$\mathbf{u}^*=\pm\frac{\mathbf{R}_{\frac{\pi}{2}}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{d}}{\left\|\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{d}\right\|}. \quad (4.12)$$

Putting $\mathbf{u}^*$ into (4.8) in Lemma 4.5, we obtain

$$\pm\mathbf{d}^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\left(\mathbf{q}-\mathbf{p}_{\mathcal{O}}+\mathbf{p}_{\mathcal{R}}\right)+\left\|\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)^{\frac{1}{2}}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{d}\right\|+\left\|\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{d}\right\|=0. \quad (4.13)$$

Because

$$\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}=\frac{1}{\sqrt{\det\mathbf{S}_{\mathcal{O}}}}\mathbf{R}_{\frac{\pi}{2}} \quad (4.14)$$

and $\det\mathbf{S}_{\mathcal{O}}>0$, (4.13) can be written as (4.11). □

Given a point on the line

The tangent line equation given a point on that line is calculated. Suppose that the point is given by $\mathbf{q}_{\mathbf{x},0}$. Because the case that $\mathbf{q}_{\mathbf{x},0}\in\partial QO_{\mathcal{R}|\mathcal{O}}(\theta_{\mathcal{R}})$ is explained in Lemma 4.5, we focuses on the case that $\mathbf{q}_{\mathbf{x},0}\notin\partial QO_{\mathcal{R}|\mathcal{O}}(\theta_{\mathcal{R}})$. From

Lemma 4.5, $\mathbf{u}^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\left(\mathbf{q}_{\mathbf{x},0}-\mathbf{p}_{\mathcal{O}}+\mathbf{p}_{\mathcal{R}}\right)-\left\|\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)^{\frac{1}{2}}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}^{*}\right\|-1=0$ is obtained. This equation is transformed into the quadratic form

$$\mathbf{u}^{*T}\mathbf{A}\mathbf{u}^{*}-2\mathbf{u}^{*T}\mathbf{b}+1=0 \tag{4.15}$$

subject to the constraint $\mathbf{u}^{*T}\mathbf{b}-1>0$, where

$$\mathbf{b}=\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\left(\mathbf{q}_{\mathbf{x},0}-\mathbf{p}_{\mathcal{O}}+\mathbf{p}_{\mathcal{R}}\right), \tag{4.16}$$

$$\mathbf{A}=\mathbf{b}\mathbf{b}^{T}-\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}. \tag{4.17}$$

In order to solve (4.15), the unit vector $\mathbf{u}^{*}$ is parameterized rationally as

$$\mathbf{u}^{*}=\frac{1}{1+t^{2}}\begin{bmatrix}1-t^{2}\\ 2t\end{bmatrix}. \tag{4.18}$$

Then the left-hand side of (4.15) is written as a fraction whose numerator is $\left(1+t^{2}\right)^{2}$ and denominator is

$$\begin{aligned}p_{den}\left(t\right)=&\left(a_{11}+2b_{1}+1\right)t^{4}-4\left(a_{12}+b_{2}\right)t^{3}+2\left(2a_{22}-a_{11}+1\right)t^{2}\\&+4\left(a_{12}-b_{2}\right)t+a_{11}-2b_{1}+1,\end{aligned} \tag{4.19}$$

where a_{ij} is the element of matrix $\mathbf{A}$ in the i^{th} row and j^{th} column and b_i is the i^{th} element of vector $\mathbf{b}$.

The equation (4.19) is solved as mentioned in [80]. First of all, the Frobenius companion matrix of the polynomial is computed. For a given polynomial $p\left(t\right)=$

$t^n + c_{n-1}t^{n-1} + \cdots + c_1 t + c_0$, the companion matrix is defined as

$$\mathbf{C}\left(p\right) = \begin{bmatrix} 0 & 0 & \cdots & 0 & -c_0 \\ 1 & 0 & \cdots & 0 & -c_1 \\ 0 & 1 & \cdots & 0 & -c_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & -c_{n-1} \end{bmatrix}. \tag{4.20}$$

Therefore, for (4.19), the companion matrix has the form

$$\mathbf{C}\left(p_{den}\right) = \begin{bmatrix} 0 & 0 & 0 & \frac{-a_{11}+2b_1-1}{a_{11}+2b_1+1} \\ 1 & 0 & 0 & \frac{-4(a_{12}-b_2)}{a_{11}+2b_1+1} \\ 0 & 1 & 0 & \frac{-2(2a_{22}-a_{11}+1)}{a_{11}+2b_1+1} \\ 0 & 0 & 1 & \frac{4(a_{12}+b_2)}{a_{11}+2b_1+1} \end{bmatrix}. \tag{4.21}$$

Next, the eigenvalues of the companion matrix, corresponding to the roots of the polynomial, are calculated by the QZ algorithm [81].

If the degree of the denominator is equal to that of the numerator, which is four, the solution of (4.15) is equivalent to the roots of (4.19). If it is less than four, the solution of (4.15) involves not only unit vectors associated with the roots of (4.19) but also $\mathbf{u}^* = \begin{bmatrix}-1 & 0\end{bmatrix}^T$, which is the limit as $t \to \infty$. Among the four solutions of (4.15), only two of them, $\mathbf{u}_1^*$ and $\mathbf{u}_2^*$, satisfy $\mathbf{u}^{*T}\mathbf{b} - 1 > 0$. Accordingly, the tangent lines containing a given point $\mathbf{q}_{\mathbf{x},0}$ are represented by, for $i = 1, 2$,

$$\mathbf{u}_i^{*T}\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\left(\mathbf{q} - \mathbf{q}_{\mathbf{x},0}\right) = 0. \tag{4.22}$$

4.3.2 Closest point on the outline of C-obstacle

In the motion planning, robots and obstacles are given sufficient security margins due to the uncertainty incurred in motion estimation. Therefore, in some cases, even though a robot $\mathcal{R}$ does not collide with an obstacle $\mathcal{O}$, the origin, the position of the robot in its configuration space, may be contained in $QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$, which means that the obstacle breaks into the safe margin of the robot. At this point, the robot should find the closest point on the outline of the C-obstacle and move to that point in order to escape the $QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$ as soon as possible. In addition to the case that $\mathbf{0} \in QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$, it is useful in motion planning to calculate the closest point on $QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$ to the origin.

The problem to find the closest point on the outline of the C-obstacle $QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)$ from the origin is formulated as

$$\mathbf{q}_c = \underset{\mathbf{q}\in\partial QO_{\mathcal{R}|\mathcal{O}}\left(\theta_{\mathcal{R}}\right)}{\arg\min} \left\|\mathbf{q}\right\|. \tag{4.23}$$

Let $\mathbf{q}_c$ and $\mathbf{u}_c$ be the closest point and its parameter. Then the $\mathbf{q}_c$ is parallel to its surface normal vector $\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_c$ from (4.5), as shown in Fig. 4.3 . Thus

$$\mathbf{u}_c^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{q}_c = 0 \tag{4.24}$$

is satisfied. From (3.3), this can be expressed as

$$\mathbf{u}_c^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{R}_{\frac{\pi}{2}}\left(\mathbf{p}_{\mathcal{O}}-\mathbf{p}_{\mathcal{R}}\right)+\mathbf{u}_c^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{S}_{\mathcal{O}}^{\frac{1}{2}}\mathbf{u}_c+\frac{\mathbf{u}_c^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_c}{\left\|\mathbf{S}_{\mathcal{R}}^{\frac{1}{2}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_c\right\|}=0. \tag{4.25}$$

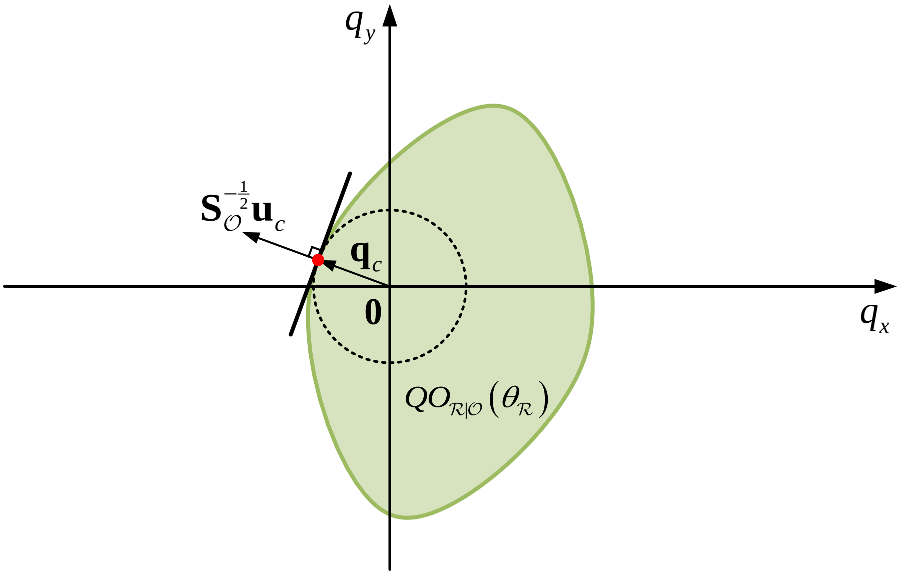

q_y
$\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_c$
$\mathbf{q}_c$
$\mathbf{0}$
q_x
$QO_{\mathcal{R}|\mathcal{O}}(\theta_{\mathcal{R}})$

Figure 4.3 The closest point on the outline of the C-obstacle from the origin.

This implies that

$$\begin{aligned}&\left(\mathbf{u}_c^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{R}_{\frac{\pi}{2}}\left(\mathbf{p}_{\mathcal{O}}-\mathbf{p}_{\mathcal{R}}\right)+\mathbf{u}_c^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{S}_{\mathcal{O}}^{\frac{1}{2}}\mathbf{u}_c\right)^2\mathbf{u}_c^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_c\\&=\left(\mathbf{u}_c^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_c\right)^2.\end{aligned}\tag{4.26}$$

If the unit vector $\mathbf{u}_c$ is parameterized rationally likewise (4.18), (4.26) is transformed into an univariate polynomial equation of degree 12. This equation also can be solved by the QZ algorithm as described in [80]. Although there are several solutions of (4.26), the one that corresponds to the closest point to the origin is the solution of (4.25).

4.4 Ellipse-based Velocity Obstacles

In this section, the ellipse-based velocity obstacle (EBVO) is defined on the basis of the tangent line equations of the C-obstacle derived in the preceding section. Because we concentrate on the linear motion planning of the robot $\mathcal{R}$, its configuration is only characterized by the position, so that $\mathcal{Q} \in \mathbb{R}^2$. Also, the orientation of the robot, $\theta_{\mathcal{R}}$ is omitted. As a result, $QO_{\mathcal{R}|\mathcal{O}} = QO_{\mathcal{R}|\mathcal{O}}(\theta_{\mathcal{R}})$ and $\mathbf{S}_{\mathcal{R}} = \mathbf{S}_{\mathcal{R}}(\theta_{\mathcal{R}})$ hereafter.

The velocity obstacle (VO) was defined as the result of the mapping of the C-obstacle to its velocity space in [40]. When both the robot and the obstacle are ellipses, the VO formed by them is called the EBVO.

Definition 4.7 (Ellipse-based Velocity Obstacle) *The ellipse-based velocity obstacle $VO^{\tau}_{\mathcal{R}|\mathcal{O}}$ for an elliptic robot $\mathcal{R}$ induced by an elliptic obstacle $\mathcal{O}$ is the set of all the robot's linear velocities $\mathbf{v}$ that would cause a collision with $\mathcal{O}$ within a time horizon $\tau > 0$, given by*

$$VO^{\tau}_{\mathcal{R}|\mathcal{O}} = \left\{ \mathbf{v} \in \mathbb{R}^2 \,\middle|\, l\left(\mathbf{0}, \mathbf{v} - \mathbf{v}_{\mathcal{O}}, \tau\right) \cap QO_{\mathcal{R}|\mathcal{O}} \neq \emptyset \right\}, \tag{4.27}$$

where $l(\mathbf{p}, \mathbf{v}, \tau) = \{\mathbf{p} + t\mathbf{v} |\, 0 \leq t \leq \tau\}$, which is a line segment starting at $\mathbf{p}$ and ending at $\mathbf{p} + \tau\mathbf{v}$.

In [18], it was proven that a necessary and sufficient condition for the collision between $\mathcal{R}$ and $\mathcal{O}$ was that $\mathbf{v}_{\mathcal{O}} - \mathbf{v}_{\mathcal{R}} \in \operatorname{cone}\left(QO_{\mathcal{R}|\mathcal{O}}\right)$, where $\operatorname{cone}(\mathcal{C})$ was the conic hull of a set $\mathcal{C}$, given by

$$\operatorname{cone}(\mathcal{C}) = \left\{ \sum_{i=1}^{k} \alpha_i \mathbf{q}_i \,\middle|\, \mathbf{q}_i \in \mathcal{C}, \alpha_i \geq 0 \text{ for all } i, k \in \mathbb{N} \right\}. \tag{4.28}$$

From the above condition, the EBVO $VO^{\infty}_{\mathcal{R}|\mathcal{O}}$ with infinite time horizon can be represented by the translation of the conic hull of the C-obstacle $QO_{\mathcal{R}|\mathcal{O}}$ by $\mathbf{v}_{\mathcal{O}}$:

$$VO^{\infty}_{\mathcal{R}|\mathcal{O}} = \text{cone}\left(QO_{\mathcal{R}|\mathcal{O}}\right) \oplus \{\mathbf{v}_{\mathcal{O}}\}\,. \tag{4.29}$$

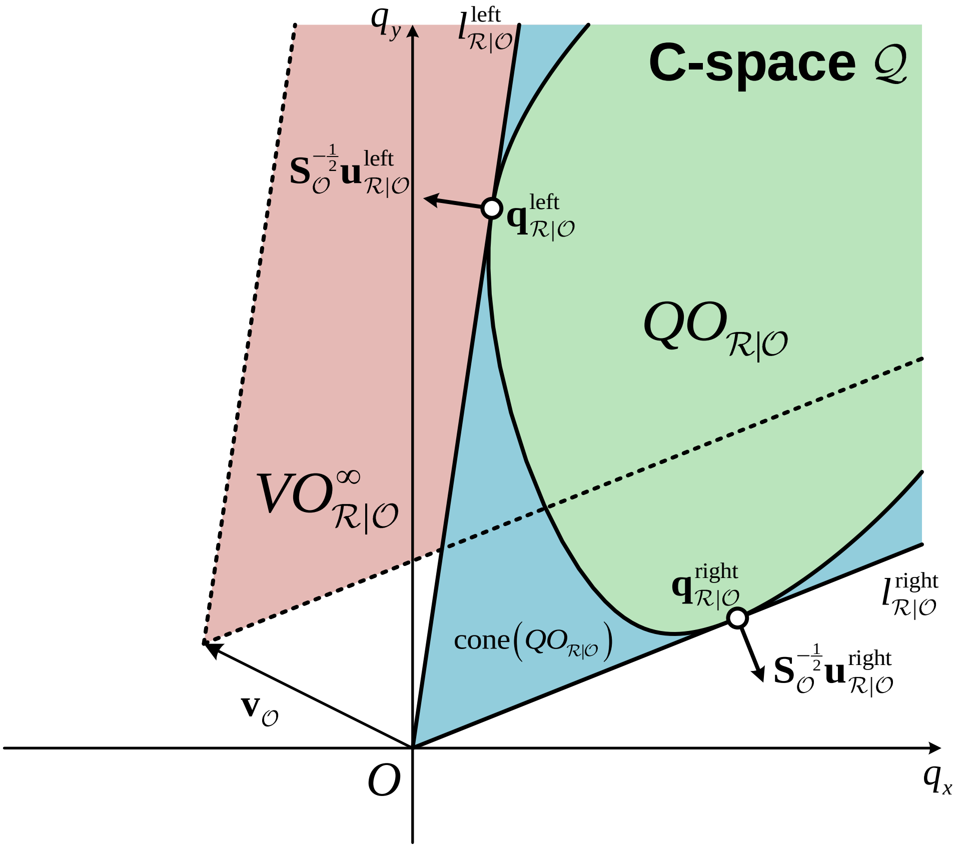
As shown in Fig. 4.4, $\text{cone}\left(QO_{\mathcal{R}|\mathcal{O}}\right)$ is bounded by the two tangent lines $l^{\text{left}}_{\mathcal{R}|\mathcal{O}}$ and $l^{\text{right}}_{\mathcal{R}|\mathcal{O}}$. On the basis of the tangent line equations in Section 4.3, the points of tangency can be easily obtained. Let $\mathbf{q}^{\text{left}}_{\mathcal{R}|\mathcal{O}}$ and $\mathbf{q}^{\text{right}}_{\mathcal{R}|\mathcal{O}}$ be the contact points of the lines that pass through the origin and touch the left and right side of $\mathcal{QO}$, respectively. Also, let $\mathbf{u}^{\text{left}}_{\mathcal{R}|\mathcal{O}}$ and $\mathbf{u}^{\text{right}}_{\mathcal{R}|\mathcal{O}}$ be the parameters of $\mathbf{q}^{\text{left}}_{\mathcal{R}|\mathcal{O}}$ and $\mathbf{q}^{\text{right}}_{\mathcal{R}|\mathcal{O}}$. Then $\text{cone}\left(QO_{\mathcal{R}|\mathcal{O}}\right)$ is the intersection of two half-spaces as the following lemma.

Lemma 4.8 *Suppose $\mathbf{u}^{\text{left}}_{\mathcal{R}|\mathcal{O}}$ and $\mathbf{u}^{\text{right}}_{\mathcal{R}|\mathcal{O}}$ are the parameters of the points $\mathbf{q}^{\text{left}}_{\mathcal{R}|\mathcal{O}}$ and $\mathbf{q}^{\text{right}}_{\mathcal{R}|\mathcal{O}}$ on the lines that pass through the origin and tangent to the C-obstacle $QO_{\mathcal{R}|\mathcal{O}}$. Then the conic hull of $QO_{\mathcal{R}|\mathcal{O}}$, $\text{cone}\left(QO_{\mathcal{R}|\mathcal{O}}\right)$ has the form*

$$\text{cone}\left(QO_{\mathcal{R}|\mathcal{O}}\right) = \left\{\mathbf{q} \in \mathcal{Q} \,\middle|\, \mathbf{q}^T \mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}} \mathbf{u}^{\text{left}}_{\mathcal{R}|\mathcal{O}} \leq 0, \mathbf{q}^T \mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}} \mathbf{u}^{\text{right}}_{\mathcal{R}|\mathcal{O}} \leq 0 \right\}. \tag{4.30}$$

Proof. Let g denote a function $g : \mathcal{Q} \to \mathbb{R}$ such that $g\left(\mathbf{q}\right) = \mathbf{q}^T \mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}} \mathbf{u}^{\text{left}}_{\mathcal{R}|\mathcal{O}}$. Then a line $\{\mathbf{q} \in \mathcal{Q}\,|g\left(\mathbf{q}\right) = 0\}$ is tangent to the left-side of $QO_{\mathcal{R}|\mathcal{O}}$ and pass through the origin by (4.5). Because $\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}} \in QO_{\mathcal{R}|\mathcal{O}}$ and

$$\begin{aligned} g\left(\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}}\right) &= \left(\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}}\right)^T \mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}} \mathbf{u}^{\text{left}}_{\mathcal{R}|\mathcal{O}} \\ &= -\left\|\mathbf{S}_{\mathcal{R}}^{\frac{1}{2}} \mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}} \mathbf{u}^{\text{left}}_{\mathcal{R}|\mathcal{O}}\right\| - 1 < 0 \end{aligned} \tag{4.31}$$



q_y
$l^{\text{left}}_{\mathcal{R}|\mathcal{O}}$
C-space $\mathcal{Q}$
$\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}^{\text{left}}_{\mathcal{R}|\mathcal{O}}$
$\mathbf{q}^{\text{left}}_{\mathcal{R}|\mathcal{O}}$
$QO_{\mathcal{R}|\mathcal{O}}$
$VO^{\infty}_{\mathcal{R}|\mathcal{O}}$
$\mathbf{q}^{\text{right}}_{\mathcal{R}|\mathcal{O}}$
$l^{\text{right}}_{\mathcal{R}|\mathcal{O}}$
$\text{cone}\left(QO_{\mathcal{R}|\mathcal{O}}\right)$
$\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}^{\text{right}}_{\mathcal{R}|\mathcal{O}}$
$\mathbf{v}_{\mathcal{O}}$
O
q_x

Figure 4.4 The conic hull of the C-obstacle $QO_{\mathcal{R}|\mathcal{O}}$, cone$\left(QO_{\mathcal{R}|\mathcal{O}}\right)$, and the EBVO $VO^{\infty}_{\mathcal{R}|\mathcal{O}}$. Two tangent lines $l^{\text{left}}_{\mathcal{R}|\mathcal{O}}$ and $l^{\text{right}}_{\mathcal{R}|\mathcal{O}}$ bound the conic hull and touch $QO_{\mathcal{R}|\mathcal{O}}$ at $\mathbf{q}^{\text{left}}_{\mathcal{R}|\mathcal{O}}$ and $\mathbf{q}^{\text{right}}_{\mathcal{R}|\mathcal{O}}$, respectively. The normal vectors of the lines are $\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}^{\text{left}}_{\mathcal{R}|\mathcal{O}}$ and $\mathbf{S}_{\mathcal{R}|\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}^{\text{right}}_{\mathcal{R}|\mathcal{O}}$, where $\mathbf{u}^{\text{left}}_{\mathcal{R}|\mathcal{O}}$ and $\mathbf{u}^{\text{right}}_{\mathcal{R}|\mathcal{O}}$ is the parameters of $\mathbf{q}^{\text{left}}_{\mathcal{R}|\mathcal{O}}$ and $\mathbf{q}^{\text{right}}_{\mathcal{R}|\mathcal{O}}$ in (3.3). Furthermore, the EBVO $VO^{\infty}_{\mathcal{R}|\mathcal{O}}$ is the translation of cone$\left(QO_{\mathcal{R}|\mathcal{O}}\right)$ by the velocity of the obstacle $\mathcal{O}$.

from (4.8), we see that the region cone $\left(QO_{\mathcal{R}|\mathcal{O}}\right)$ is contained in the half-space $\{\mathbf{q} \in \mathcal{Q}\,|g(\mathbf{q}) \leq 0\}$. Hence,

$$\text{cone}\left(QO_{\mathcal{R}|\mathcal{O}}\right) \subset \left\{\mathbf{q} \in \mathcal{Q}\,\middle|\,\mathbf{q}^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_{\mathcal{R}|\mathcal{O}}^{\text{left}} \leq 0\right\}. \tag{4.32}$$

Similarly, it can be proven that

$$\text{cone}\left(QO_{\mathcal{R}|\mathcal{O}}\right) \subset \left\{\mathbf{q} \in \mathcal{Q}\,\middle|\,\mathbf{q}^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_{\mathcal{R}|\mathcal{O}}^{\text{right}} \leq 0\right\}. \tag{4.33}$$

Therefore, the left-hand side of (4.30) is a subset of the right-hand side.

Conversely, suppose that there exists a point $\mathbf{q} \in \text{cone}\left(QO_{\mathcal{R}|\mathcal{O}}\right)$ satisfying $\mathbf{q}^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_{\mathcal{R}|\mathcal{O}}^{\text{left}} > 0$. Since the region $QO_{\mathcal{R}|\mathcal{O}}$ is convex and $\mathbf{q}$ is not on the line $l_{\mathcal{R}|\mathcal{O}}^{\text{left}}$ from Lemma 4.4, there exists $\mathbf{q}_1 \in QO_{\mathcal{R}|\mathcal{O}} \setminus \partial QO_{\mathcal{R}|\mathcal{O}}$ such that $\mathbf{q} = k_1\mathbf{q}_1$ for some $k_1 > 0$ and $B(\mathbf{q}_1, \epsilon) = \{\mathbf{q}\,|\|\mathbf{q} - \mathbf{q}_1\| \leq \epsilon\} \subset QO_{\mathcal{R}|\mathcal{O}}$ for some $\epsilon > 0$. Let

$$\mathbf{q}_2 = k_2(\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}}) + (1 - k_2)\,\mathbf{q}_1 \tag{4.34}$$

be given, where

$$k_2 = \frac{\mathbf{q}_1^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_{\mathcal{R}|\mathcal{O}}^{\text{left}}}{(\mathbf{q}_1 - \mathbf{p}_{\mathcal{O}} + \mathbf{p}_{\mathcal{R}})^T\,\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_{\mathcal{R}|\mathcal{O}}^{\text{left}}}. \tag{4.35}$$

Because $B(\mathbf{q}_2, k_2\epsilon) \subset QO_{\mathcal{R}|\mathcal{O}}$, $\mathbf{q}_2 \in QO_{\mathcal{R}|\mathcal{O}} \setminus \partial QO_{\mathcal{R}|\mathcal{O}}$. However, $\mathbf{q}_2$ is a point of tangency since $\mathbf{q}_2^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_{\mathcal{R}|\mathcal{O}}^{\text{left}} = 0$ from Lemma 4.4, which implies that $\mathbf{q}_2 \in \partial QO_{\mathcal{R}|\mathcal{O}}$. This contradicts the assumption that $\mathbf{q} \in \text{cone}\left(QO_{\mathcal{R}|\mathcal{O}}\right)$ satisfies $\mathbf{q}^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_{\mathcal{R}|\mathcal{O}}^{\text{left}} > 0$. The case of $\mathbf{q}^T\mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_{\mathcal{R}|\mathcal{O}}^{\text{right}} > 0$ can be proven similarly. Thus the right-hand side of (4.30) is a subset of the left-hand side.

This completes the proof. □

Corollary 4.9 *If* $\mathbf{u}^{\text{left}}_{\mathcal{R}|\mathcal{O}}$ *and* $\mathbf{u}^{right}_{\mathcal{R}|\mathcal{O}}$ *are the parameters of the contact points between the lines passing through the origin and the C-obstacle* $\mathcal{QO}$*, then* $VO^{\infty}_{\mathcal{R}|\mathcal{O}}$ *is expressed as*

$$VO^{\infty}_{\mathcal{R}|\mathcal{O}} = \left\{ \mathbf{v} \in \mathbb{R}^2 \,\middle|\, (\mathbf{v} - \mathbf{v}_{\mathcal{O}})^T \, \mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}} \mathbf{u}^{\text{left}}_{\mathcal{R}|\mathcal{O}} \leq 0, \; (\mathbf{v} - \mathbf{v}_{\mathcal{O}})^T \, \mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}} \mathbf{u}^{\text{right}}_{\mathcal{R}|\mathcal{O}} \leq 0 \right\}. \tag{4.36}$$

For a finite value of τ, let $\text{cone}^{\tau}(\mathcal{C})$ is a set of combinations of points in $\mathcal{C}$ such that

$$\text{cone}^{\tau}(\mathcal{C}) = \left\{ \sum_{i=1}^{k} \alpha_i \mathbf{q}_i \,\middle|\, \mathbf{q}_i \in \mathcal{C}, \tau\alpha_i \geq 1 \text{ for all } i, k \in \mathbb{N} \right\}. \tag{4.37}$$

Similar to (4.29), the region $VO^{\tau}_{\mathcal{R}|\mathcal{O}}$ is expressed as

$$VO^{\tau}_{\mathcal{R}|\mathcal{O}} = \text{cone}^{\tau}\left(QO_{\mathcal{R}|\mathcal{O}} \right) \oplus \{\mathbf{v}_{\mathcal{O}}\}, \tag{4.38}$$

As a result, the region in (4.36) is truncated by a curve segment $\gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ satisfying the following equation:

$$\gamma^{\tau}_{\mathcal{R}|\mathcal{O}} \subset \left\{ \mathbf{v} \in \mathbb{R}^2 \,\middle|\, \tau(\mathbf{v} - \mathbf{v}_{\mathcal{O}}) \in \partial QO_{\mathcal{R}|\mathcal{O}} \right\}. \tag{4.39}$$

However, the equation of $\gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ is not appropriate for the determination of the region $VO^{\tau}_{\mathcal{R}|\mathcal{O}}$ because it is difficult to know whether a given point is on the left or right side of the curve. Therefore, the region $VO^{\tau}_{\mathcal{R}|\mathcal{O}}$ is represented as the union of two regions as shown in Fig. 4.5.

The first region is $\tau^{-1}QO_{\mathcal{R}|\mathcal{O}} \oplus \{\mathbf{v}_{\mathcal{O}}\}$, which can be determined from the algebraic condition in Section 3.4 as follows. To consider the effect that the

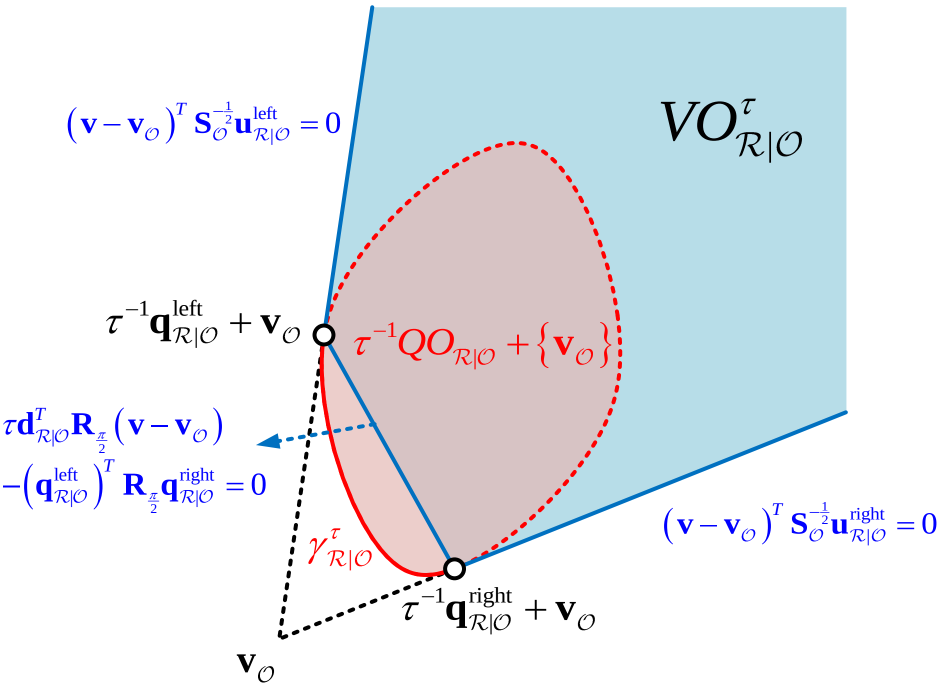

$\left(\mathbf{v}-\mathbf{v}_{\mathcal{O}}\right)^T \mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_{\mathcal{R}|\mathcal{O}}^{\text{left}} = 0$
$VO_{\mathcal{R}|\mathcal{O}}^{\tau}$
$\tau^{-1}\mathbf{q}_{\mathcal{R}|\mathcal{O}}^{\text{left}} + \mathbf{v}_{\mathcal{O}}$
$\tau^{-1}QO_{\mathcal{R}|\mathcal{O}} + \left\{\mathbf{v}_{\mathcal{O}}\right\}$
$\tau\mathbf{d}_{\mathcal{R}|\mathcal{O}}^T \mathbf{R}_{\frac{\pi}{2}}\left(\mathbf{v}-\mathbf{v}_{\mathcal{O}}\right)$
$-\left(\mathbf{q}_{\mathcal{R}|\mathcal{O}}^{\text{left}}\right)^T \mathbf{R}_{\frac{\pi}{2}}\mathbf{q}_{\mathcal{R}|\mathcal{O}}^{\text{right}} = 0$
$\left(\mathbf{v}-\mathbf{v}_{\mathcal{O}}\right)^T \mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}}\mathbf{u}_{\mathcal{R}|\mathcal{O}}^{\text{right}} = 0$
$\gamma_{\mathcal{R}|\mathcal{O}}^{\tau}$
$\tau^{-1}\mathbf{q}_{\mathcal{R}|\mathcal{O}}^{\text{right}} + \mathbf{v}_{\mathcal{O}}$
$\mathbf{v}_{\mathcal{O}}$

Figure 4.5 The EBVO $VO_{\mathcal{R}|\mathcal{O}}^{\tau}$ that is the union of $\tau^{-1}QO_{\mathcal{R}|\mathcal{O}} \oplus \{\mathbf{v}_{\mathcal{O}}\}$ and the region bounded by three line segments.

C-obstacle $QO_{\mathcal{R}|\mathcal{O}}$ is shrunk by a factor τ and to test the inclusion of the current robot's velocity $\mathbf{v}_{\mathcal{R}}$ in that region, the major and minor radii of the robot and the obstacle decrease by τ, so that $\bar{\mathbf{S}}_{\mathcal{R}} = \tau^{-2}\mathbf{S}_{\mathcal{R}}$ and $\bar{\mathbf{S}}_{\mathcal{O}} = \tau^{-2}\mathbf{S}_{\mathcal{O}}$. In addition, the relative position of the obstacle with respect to the robot turns into $\bar{\mathbf{p}}_{\mathcal{R}|\mathcal{O}} = \tau^{-1}\mathbf{p}_{\mathcal{R}|\mathcal{O}} - \mathbf{v}_{\mathcal{R}} + \mathbf{v}_{\mathcal{O}}$. If the coefficients of the characteristic polynomial of the robot and the obstacle for $\bar{\mathbf{S}}_{\mathcal{R}}$, $\bar{\mathbf{S}}_{\mathcal{O}}$, and $\bar{\mathbf{p}}_{\mathcal{R}|\mathcal{O}}$ are computed by the equations from (3.21) to (3.30), it can be determined by the four inequalities in Theorem 3.18 whether the current linear velocity $\mathbf{v}_{\mathcal{R}}$ is contained in the region $\tau^{-1}QO_{\mathcal{R}|\mathcal{O}} \oplus \{\mathbf{v}_{\mathcal{O}}\}$ or not .

The other region is the truncated cone whose apex is cut off by the line segment connecting the two points $\tau^{-1}\mathbf{q}^{\text{left}}_{\mathcal{R}|\mathcal{O}} + \mathbf{v}_{\mathcal{O}}$ and $\tau^{-1}\mathbf{q}^{\text{right}}_{\mathcal{R}|\mathcal{O}} + \mathbf{v}_{\mathcal{O}}$, which are the points of tangency between $\tau^{-1}QO_{\mathcal{R}|\mathcal{O}} \oplus \{\mathbf{v}_{\mathcal{O}}\}$ and the boundary of $VO^{\infty}_{\mathcal{R}|\mathcal{O}}$. Let $\mathbf{d}_{\mathcal{R}|\mathcal{O}}$ denote a vector $\mathbf{d}_{\mathcal{R}|\mathcal{O}} = \mathbf{q}^{\text{left}}_{\mathcal{R}|\mathcal{O}} - \mathbf{q}^{\text{right}}_{\mathcal{R}|\mathcal{O}}$. Then the line segment is a subset of the line represented by

$$\tau \mathbf{d}^{T}_{\mathcal{R}|\mathcal{O}} \mathbf{R}_{\frac{\pi}{2}} \left(\mathbf{v} - \mathbf{v}_{\mathcal{O}}\right) - \left(\mathbf{q}^{\text{left}}_{\mathcal{R}|\mathcal{O}}\right)^{T} \mathbf{R}_{\frac{\pi}{2}} \mathbf{q}^{\text{right}}_{\mathcal{R}|\mathcal{O}} = 0, \tag{4.40}$$

as shown in Fig. 4.5. In conclusion, the following theorem is derived.

Theorem 4.10 *Consider an elliptic robot $\mathcal{R}$ and an elliptic obstacle $\mathcal{O}$ in the planar workspace $\mathcal{W}$. If the robot selects its new linear velocity $\mathbf{v}^{new}_{\mathcal{R}}$ outside of*

$$\begin{aligned} VO^{\tau}_{\mathcal{R}|\mathcal{O}} = &\left\{\tau^{-1}QO_{\mathcal{R}|\mathcal{O}} \oplus \{\mathbf{v}_{\mathcal{O}}\}\right\} \\ &\cup \left\{ \mathbf{v} \in \mathbb{R}^2 \middle| \left(\mathbf{v} - \mathbf{v}_{\mathcal{O}}\right)^{T} \mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}} \mathbf{u}^{\text{left}}_{\mathcal{R}|\mathcal{O}} \leq 0, \left(\mathbf{v} - \mathbf{v}_{\mathcal{O}}\right)^{T} \mathbf{S}_{\mathcal{O}}^{-\frac{1}{2}} \mathbf{u}^{\text{right}}_{\mathcal{R}|\mathcal{O}} \leq 0, \right. \\ &\left. \tau \mathbf{d}^{T}_{\mathcal{R}|\mathcal{O}} \mathbf{R}_{\frac{\pi}{2}} \left(\mathbf{v} - \mathbf{v}_{\mathcal{O}}\right) - \left(\mathbf{q}^{\text{left}}_{\mathcal{R}|\mathcal{O}}\right)^{T} \mathbf{R}_{\frac{\pi}{2}} \mathbf{q}^{\text{right}}_{\mathcal{R}|\mathcal{O}} \geq 0 \right\}, \end{aligned} \tag{4.41}$$

then the robot $\mathcal{R}$ does not collide with the obstacle $\mathcal{O}$ within a time horizon τ.

4.5 Selection of Collision-free Linear Velocity

In this section, a strategy to determine the new linear velocity of the robot is presented on the basis of the EBVO derived in the preceding section. For the sake of efficient computation, the region defined in Theorem 4.10 is approximated by the intersection of three half spaces. Next, how to select the new linear velocity with the approximated EBVOs is explained in detail.

4.5.1 Conservative Approximation of the EBVOs

In order to avoid a collision with the obstacle $\mathcal{O}$, the robot has to select its new velocity outside of the EBVO. However, the computation of collision-free velocities is demanded when multiple obstacles are considered since the region in (4.41) has the curved boundary $\gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$. Therefore, we will conservatively approximate $VO^{\tau}_{\mathcal{R}|\mathcal{O}}$ to $\widetilde{VO}^{\tau}_{\mathcal{R}|\mathcal{O}}$ by replacing $\gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ to its tangent line $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$.

There are three methods for calculating the approximated tangent line $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$, which are all equivalent to one another when the robot and the obstacle are circular, as shown in Fig. 4.6.

- The normal vector of $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ is set to $\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}}$.
- The line $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ passes through the closest point in $VO^{\tau}_{\mathcal{R}|\mathcal{O}}$ to $\mathbf{v}_{\mathcal{O}}$.
- The direction vector of $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ is set to $\mathbf{d}_{\mathcal{R}|\mathcal{O}} = \mathbf{q}^{\text{left}}_{\mathcal{R}|\mathcal{O}} - \mathbf{q}^{\text{right}}_{\mathcal{R}|\mathcal{O}}$.

The first method has been suggested in the conventional studies that solve the local collision avoidance problem of circular robots based on the velocity-based approach. The line $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ is easily obtained from Lemma 4.6.

Lemma 4.11 *If $\mathbf{p}_{\mathcal{R}|\mathcal{O}} = \mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}}$ is the normal vector of $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$, then the half-space supported by $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ and containing $VO^{\tau}_{\mathcal{R}|\mathcal{O}}$ is expressed by*

$$\tau \mathbf{p}^{T}_{\mathcal{R}|\mathcal{O}} (\mathbf{v} - \mathbf{v}_{\mathcal{O}}) - \left\| \mathbf{p}_{\mathcal{R}|\mathcal{O}} \right\|^2 + \left\| \mathbf{S}^{\frac{1}{2}}_{\mathcal{R}} \mathbf{p}_{\mathcal{R}|\mathcal{O}} \right\| + \sqrt{\det \mathbf{S}_{\mathcal{O}}} \left\| \mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}} \mathbf{R}_{\frac{\pi}{2}} \mathbf{p}_{\mathcal{R}|\mathcal{O}} \right\| \geq 0. \tag{4.42}$$

Proof. Let $\mathbf{d} = \mathbf{R}_{\frac{\pi}{2}} (\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}})$ be substituted in (4.11). Then

$$\pm \left(\mathbf{p}^{T}_{\mathcal{R}|\mathcal{O}} \mathbf{q} - \left\| \mathbf{p}_{\mathcal{R}|\mathcal{O}} \right\|^2 \right) + \left\| \mathbf{S}^{\frac{1}{2}}_{\mathcal{R}} \mathbf{p}_{\mathcal{R}|\mathcal{O}} \right\| + \sqrt{\det \mathbf{S}_{\mathcal{O}}} \left\| \mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}} \mathbf{R}_{\frac{\pi}{2}} \mathbf{p}_{\mathcal{R}|\mathcal{O}} \right\| = 0. \tag{4.43}$$

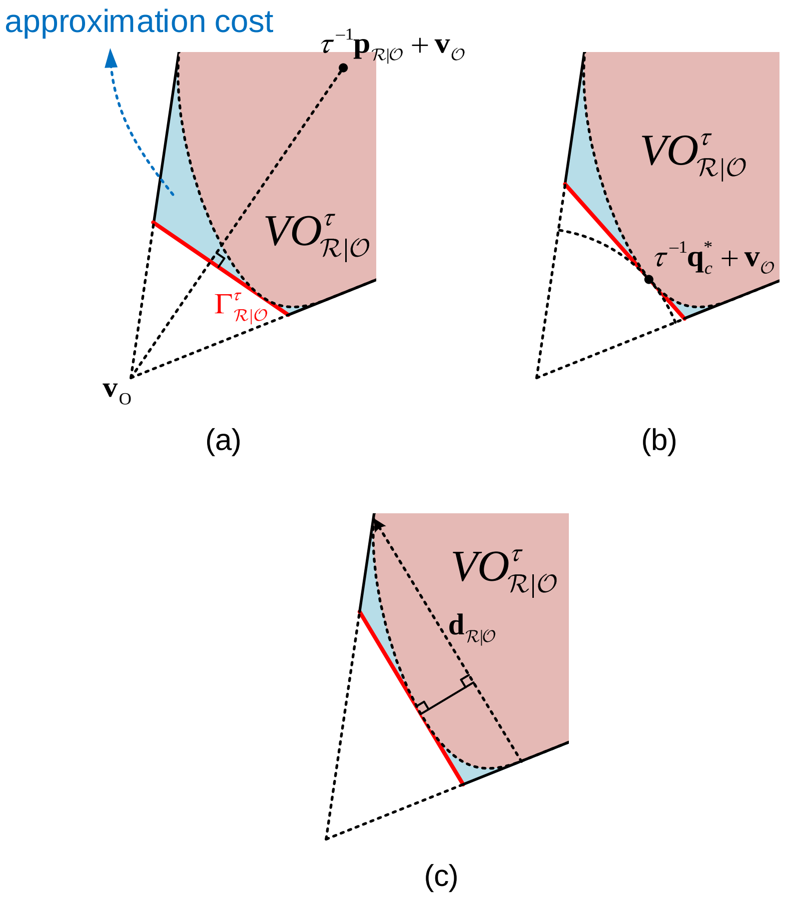

approximation cost
$\tau^{-1}\mathbf{p}_{\mathcal{R}|\mathcal{O}} + \mathbf{v}_{\mathcal{O}}$
$VO^{\tau}_{\mathcal{R}|\mathcal{O}}$
$\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$
$\mathbf{v}_{\mathrm{O}}$
(a)
$VO^{\tau}_{\mathcal{R}|\mathcal{O}}$
$\tau^{-1}\mathbf{q}^{*}_{c} + \mathbf{v}_{\mathcal{O}}$
(b)
$VO^{\tau}_{\mathcal{R}|\mathcal{O}}$
$\mathbf{d}_{\mathcal{R}|\mathcal{O}}$
(c)

Figure 4.6 The approximated tangent lines $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ for three methods: (a) The normal vector of $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ is set to $\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}}$; (b) The line $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ passes through the closest point in $VO^{\tau}_{\mathcal{R}|\mathcal{O}}$ to $\mathbf{v}_{\mathcal{O}}$; (c) The direction vector of $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ is set to $\mathbf{d}_{\mathcal{R}|\mathcal{O}} = \mathbf{q}^{\mathrm{left}}_{\mathcal{R}|\mathcal{O}} - \mathbf{q}^{\mathrm{right}}_{\mathcal{R}|\mathcal{O}}$. The red lines represent the approximated lines $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$. The cost of the approximation is represented by the area of the blue region.

Among the two lines, $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ is associated with the closer one to origin. Hence, the sign of the first term becomes positive. From (4.39), $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ has the form

$$\tau \mathbf{p}^{T}_{\mathcal{R}|\mathcal{O}} (\mathbf{v} - \mathbf{v}_{\mathcal{O}}) - \left\|\mathbf{p}_{\mathcal{R}|\mathcal{O}}\right\|^{2} + \left\|\mathbf{S}^{\frac{1}{2}}_{\mathcal{R}} \mathbf{p}_{\mathcal{R}|\mathcal{O}}\right\| + \sqrt{\det \mathbf{S}_{\mathcal{O}}} \left\|\mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}} \mathbf{R}_{\frac{\pi}{2}} \mathbf{p}_{\mathcal{R}|\mathcal{O}}\right\| = 0. \quad (4.44)$$

Because $\mathbf{v}_{\mathcal{O}} + \tau^{-1} \mathbf{p}_{\mathcal{R}|\mathcal{O}} \in VO^{\tau}_{\mathcal{R}|\mathcal{O}}$ and $\left\|\mathbf{S}^{\frac{1}{2}}_{\mathcal{R}} \mathbf{p}_{\mathcal{R}|\mathcal{O}}\right\| + \sqrt{\det \mathbf{S}_{\mathcal{O}}} \left\|\mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}} \mathbf{R}_{\frac{\pi}{2}} \mathbf{p}_{\mathcal{R}|\mathcal{O}}\right\| \geq 0$, (4.42) holds. □

However, this method does not reflect the geometry of $VO^{\tau}_{\mathcal{R}|\mathcal{O}}$ in that the curve $\gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ is not symmetric about the vector $\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}}$. Thus the approximation error, the area of $\widetilde{VO}^{\tau}_{\mathcal{R}|\mathcal{O}} \setminus VO^{\tau}_{\mathcal{R}|\mathcal{O}}$, is larger than those of the others.

The second method is to find the closest point in the region of $VO^{\tau}_{\mathcal{R}|\mathcal{O}}$ from $\mathbf{v}_{\mathcal{O}}$ and to define $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ as the supporting line of $VO_{\mathcal{R}|\mathcal{O}}$ at that point. First, the closest point in $QO_{\mathcal{R}|\mathcal{O}}$ to the origin is calculated as described in Section 4.3, and let $\mathbf{q}^{*}_{c}$ be that point. Then the tangent line pass through $\mathbf{q}^{*}_{c}$ is represented from (4.8). By (4.39), the $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ has the from of

$$\mathbf{u}^{*T}_{c} \mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}} (\tau \mathbf{v} - \tau \mathbf{v}_{\mathcal{O}} - \mathbf{p}_{\mathcal{O}} + \mathbf{p}_{\mathcal{R}}) - \left\|\mathbf{S}^{\frac{1}{2}}_{\mathcal{R}} \mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}} \mathbf{u}^{*}_{c}\right\| - 1 = 0. \quad (4.45)$$

Since $\mathbf{v}_{\mathcal{O}} + \tau^{-1} (\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}}) \in VO^{\tau}_{\mathcal{R}|\mathcal{O}}$, the half-space supported by $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ and containing $VO^{\tau}_{\mathcal{R}|\mathcal{O}}$ is expressed by

$$\mathbf{u}^{*T}_{c} \mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}} (\tau \mathbf{v} - \tau \mathbf{v}_{\mathcal{O}} - \mathbf{p}_{\mathcal{O}} + \mathbf{p}_{\mathcal{R}}) - \left\|\mathbf{S}^{\frac{1}{2}}_{\mathcal{R}} \mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}} \mathbf{u}^{*}_{c}\right\| - 1 \leq 0. \quad (4.46)$$

Although this method consider the asymmetry of $\gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ with respect to the vector $\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}}$, the computation is very expensive because the polynomial equation of degree 12 should be solved to find the closest point.

The last method is to denote the direction vector of $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ by $\mathbf{d}_{\mathcal{R}|\mathcal{O}} = \mathbf{q}^{\text{left}}_{\mathcal{R}|\mathcal{O}} - \mathbf{q}^{\text{right}}_{\mathcal{R}|\mathcal{O}}$. Not only can the line $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ be simply derived from Lemma 4.6 like the first method, but the resultant line also reflects the asymmetrical characteristic of $\gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ analogous to the second one.

Lemma 4.12 *If* $\mathbf{d}_{\mathcal{R}|\mathcal{O}}$ *is the direction vector of* $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$*, the half-space supported by* $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ *and containing* $VO^{\tau}_{\mathcal{R}|\mathcal{O}}$ *is expressed by*

$$\begin{aligned} &s_{\mathcal{R}|\mathcal{O}}\mathbf{d}^{T}_{\mathcal{R}|\mathcal{O}}\mathbf{R}_{\frac{\pi}{2}}\left(\tau\mathbf{v} - \tau\mathbf{v}_{\mathcal{O}} - \mathbf{p}_{\mathcal{O}} + \mathbf{p}_{\mathcal{R}}\right) \\ &\quad + \left\|\mathbf{S}^{\frac{1}{2}}_{\mathcal{R}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{d}_{\mathcal{R}|\mathcal{O}}\right\| + \sqrt{\det \mathbf{S}_{\mathcal{O}}}\left\|\mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}}\mathbf{d}_{\mathcal{R}|\mathcal{O}}\right\| \geq 0, \end{aligned} \tag{4.47}$$

where $s_{\mathcal{R}|\mathcal{O}} = \operatorname{sgn}\left(\mathbf{d}^{T}_{\mathcal{R}|\mathcal{O}}\mathbf{R}_{\frac{\pi}{2}}\left(\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}}\right)\right)$.

Proof. From Lemma 4.6, the tangent lines to $QO_{\mathcal{R}|\mathcal{O}}$ with $\mathbf{d}_{\mathcal{R}|\mathcal{O}}$ are

$$\pm\mathbf{d}^{T}_{\mathcal{R}|\mathcal{O}}\mathbf{R}_{\frac{\pi}{2}}\left(\mathbf{q} - \mathbf{p}_{\mathcal{O}} + \mathbf{p}_{\mathcal{R}}\right) + \left\|\mathbf{S}^{\frac{1}{2}}_{\mathcal{R}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{d}_{\mathcal{R}|\mathcal{O}}\right\| + \sqrt{\det \mathbf{S}_{\mathcal{O}}}\left\|\mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}}\mathbf{d}_{\mathcal{R}|\mathcal{O}}\right\| = 0. \tag{4.48}$$

Among the two lines, $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ is associated with the closer one to origin, which takes $s_{\mathcal{R}|\mathcal{O}} = \operatorname{sgn}\left(\mathbf{d}^{T}_{\mathcal{R}|\mathcal{O}}\mathbf{R}_{\frac{\pi}{2}}\left(\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}}\right)\right)$ as the sign of the first term in (4.48). From (4.39), $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ has the form

$$\begin{aligned} &s_{\mathcal{R}|\mathcal{O}}\mathbf{d}^{T}_{\mathcal{R}|\mathcal{O}}\mathbf{R}_{\frac{\pi}{2}}\left(\tau\mathbf{v} - \tau\mathbf{v}_{\mathcal{O}} - \mathbf{p}_{\mathcal{O}} + \mathbf{p}_{\mathcal{R}}\right) \\ &\quad + \left\|\mathbf{S}^{\frac{1}{2}}_{\mathcal{R}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{d}_{\mathcal{R}|\mathcal{O}}\right\| + \sqrt{\det \mathbf{S}_{\mathcal{O}}}\left\|\mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}}\mathbf{d}_{\mathcal{R}|\mathcal{O}}\right\| = 0. \end{aligned} \tag{4.49}$$

Since the line $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ supports $VO^{\tau}_{\mathcal{R}|\mathcal{O}}$, the region is entirely contained in one of the two closed half-spaces bounded by $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$. Because $\mathbf{v}_{\mathcal{O}} + (\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}})/\tau \in VO^{\tau}_{\mathcal{R}|\mathcal{O}}$ and $\left\|\mathbf{S}^{\frac{1}{2}}_{\mathcal{R}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{d}_{\mathcal{R}|\mathcal{O}}\right\| + \sqrt{\det \mathbf{S}_{\mathcal{O}}}\left\|\mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}}\mathbf{d}_{\mathcal{R}|\mathcal{O}}\right\| \geq 0$, (4.47) follows. □

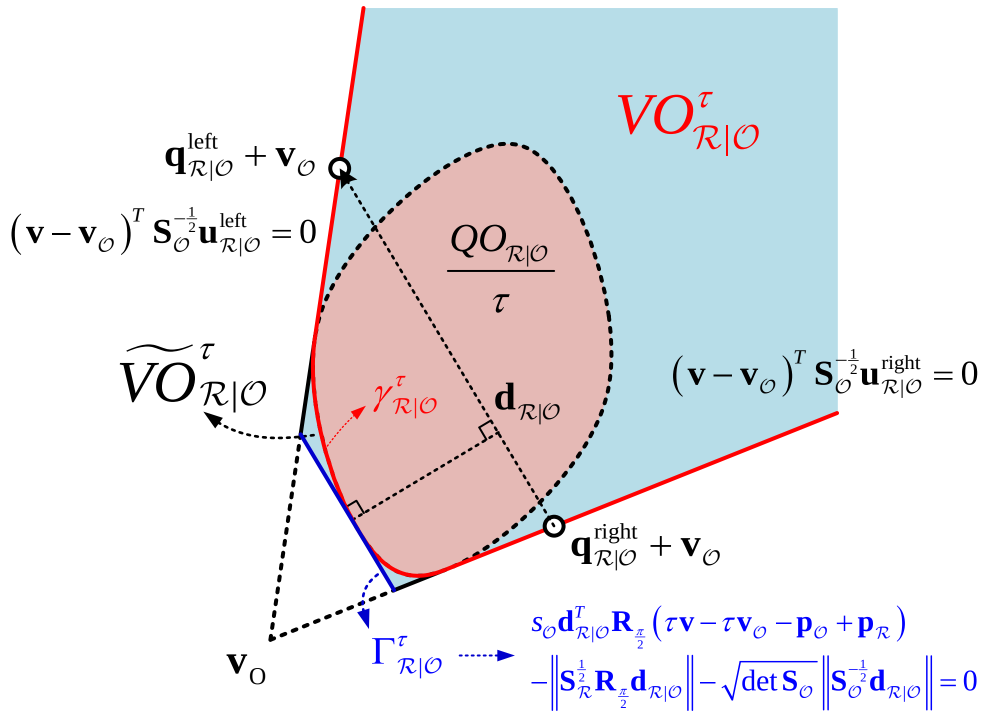

$\mathbf{q}^{\text{left}}_{\mathcal{R}|\mathcal{O}} + \mathbf{v}_{\mathcal{O}}$
$(\mathbf{v}-\mathbf{v}_{\mathcal{O}})^T \mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}} \mathbf{u}^{\text{left}}_{\mathcal{R}|\mathcal{O}} = 0$
$VO^{\tau}_{\mathcal{R}|\mathcal{O}}$
$\dfrac{QO_{\mathcal{R}|\mathcal{O}}}{\tau}$
$\widetilde{VO}^{\tau}_{\mathcal{R}|\mathcal{O}}$
$\gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$
$\mathbf{d}_{\mathcal{R}|\mathcal{O}}$
$(\mathbf{v}-\mathbf{v}_{\mathcal{O}})^T \mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}} \mathbf{u}^{\text{right}}_{\mathcal{R}|\mathcal{O}} = 0$
$\mathbf{q}^{\text{right}}_{\mathcal{R}|\mathcal{O}} + \mathbf{v}_{\mathcal{O}}$
$s_{\mathcal{O}}\mathbf{d}^{T}_{\mathcal{R}|\mathcal{O}}\mathbf{R}_{\frac{\pi}{2}}\left(\tau\mathbf{v}-\tau\mathbf{v}_{\mathcal{O}}-\mathbf{p}_{\mathcal{O}}+\mathbf{p}_{\mathcal{R}}\right)$
$-\left\|\mathbf{S}^{\frac{1}{2}}_{\mathcal{R}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{d}_{\mathcal{R}|\mathcal{O}}\right\|-\sqrt{\det\mathbf{S}_{\mathcal{O}}}\left\|\mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}}\mathbf{d}_{\mathcal{R}|\mathcal{O}}\right\|=0$
$\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$
$\mathbf{v}_{\mathrm{O}}$

Figure 4.7 The EBVO $VO^{\tau}_{\mathcal{R}|\mathcal{O}}$ and its conservative approximation $\widetilde{VO}^{\tau}_{\mathcal{R}|\mathcal{O}}$. The curve $\gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ is approximated to its tangent line $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$ whose direction vector is parallel to $\mathbf{d}_{\mathcal{R}|\mathcal{O}} = \mathbf{q}^{\text{left}}_{\mathcal{R}|\mathcal{O}} - \mathbf{q}^{\text{right}}_{\mathcal{R}|\mathcal{O}}$. The approximated EBVO $\widetilde{VO}^{\tau}_{\mathcal{R}|\mathcal{O}}$ is bounded by three lines.

In conclusion, the third method is employed in this dissertation. Hence, the approximated EBVO $\widetilde{VO}^{\tau}_{\mathcal{R}|\mathcal{O}}$ is bounded by three lines $l^{\text{left}}_{\mathcal{R}|\mathcal{O}}$, $l^{\text{right}}_{\mathcal{R}|\mathcal{O}}$, and $\Gamma^{\tau}_{\mathcal{R}|\mathcal{O}}$:

$$\begin{aligned}\widetilde{VO}^{\tau}_{\mathcal{R}|\mathcal{O}} = \Big\{ \mathbf{v} \in \mathbb{R}^2 \,\Big|\, & s_{\mathcal{R}|\mathcal{O}} \mathbf{d}^T_{\mathcal{R}|\mathcal{O}} \mathbf{R}_{\frac{\pi}{2}} \left(\tau \mathbf{v} - \tau \mathbf{v}_{\mathcal{O}} - \mathbf{p}_{\mathcal{O}} + \mathbf{p}_{\mathcal{R}} \right) \\ & + \left\| \mathbf{S}^{\frac{1}{2}}_{\mathcal{R}} \mathbf{R}_{\frac{\pi}{2}} \mathbf{d}_{\mathcal{R}|\mathcal{O}} \right\| + \sqrt{\det \mathbf{S}_{\mathcal{O}}} \left\| \mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}} \mathbf{d}_{\mathcal{R}|\mathcal{O}} \right\| \leq 0, \\ & \left(\mathbf{v} - \mathbf{v}_{\mathcal{O}} \right)^T \mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}} \mathbf{u}^{\text{left}}_{\mathcal{R}|\mathcal{O}} \leq 0, \left(\mathbf{v} - \mathbf{v}_{\mathcal{O}} \right)^T \mathbf{S}^{-\frac{1}{2}}_{\mathcal{O}} \mathbf{u}^{\text{right}}_{\mathcal{R}|\mathcal{O}} \leq 0 \Big\}, \end{aligned} \tag{4.50}$$

where $s_{\mathcal{R}|\mathcal{O}} = \operatorname{sgn} \left(\mathbf{d}^T_{\mathcal{R}|\mathcal{O}} \mathbf{R}_{\frac{\pi}{2}} \left(\mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}} \right) \right)$, as shown in Fig. 4.7.

4.5.2 New Linear Velocity Selection with Multiple Obstacles

Because the condition of the forbidden linear velocity of the robot is derived in the preceding section, it is extended to the case that multiple obstacles are moving in $\mathcal{W}$. In addition, the dynamic constraints is considered to find the reachable linear velocities in the next sampling period. Next, the preferred linear velocity, at which the robot can reach its goal in the shortest path when there is no obstacle, is calculated. Finally, the new linear velocity is determined based on the approximated velocity obstacles and the preferred linear velocity.

Combined velocity obstacle

There are $N_{\mathcal{O}}$ obstacles $\mathcal{O}_1$, $\mathcal{O}_2$, $\cdots$, $\mathcal{O}_{N_{\mathcal{O}}}$ detected by the robot as mentioned in Section 4.2. For any natural number $i \leq N_{\mathcal{O}}$, the conservative approximated EBVO, $\widetilde{VO}^{\tau}_{\mathcal{R}|\mathcal{O}_i}$, can be derived. If $\mathbf{v}^{new}_{\mathcal{R}} \in \widetilde{VO}^{\tau}_{\mathcal{R}|\mathcal{O}_i}$ for some $i \leq N_{\mathcal{O}}$, the collision occurs in time τ. Therefore, the combined EBVO is defined by the union of $\widetilde{VO}^{\tau}_{\mathcal{R}|\mathcal{O}_i}$ such that

$$\widetilde{VO}^{\tau}_{\mathcal{R}} = \bigcup_{i \leq N_{\mathcal{O}}} \widetilde{VO}^{\tau}_{\mathcal{R}|\mathcal{O}_i}. \tag{4.51}$$

If the robot does not move with a velocity outside $\widetilde{VO}^{\tau}_{\mathcal{R}}$, the robot does not collide with the obstacles.

Reachable avoidance linear velocities

Due to the robot's dynamic constraints, its available linear velocities during the next time period are limited by the set of reachable linear velocities

$$RV_{\mathcal{R}} = \left\{ \mathbf{v} \in \mathbb{R}^2 \,|\|\mathbf{v}\| \leq v^{max}_{\mathcal{R}}, \|\mathbf{v} - \mathbf{v}_{\mathcal{R}}\| \leq a^{max}_{\mathcal{R}} \Delta t \right\}. \tag{4.52}$$

By subtracting $\widetilde{VO}^{\tau}_{\mathcal{R}}$ from $RV_{\mathcal{R}}$, the set of reachable avoidance linear velocities is

$$RAV_{\mathcal{R}} = \left\{ \mathbf{v} \in \mathbb{R}^2 \middle| \mathbf{v} \in RV_{\mathcal{R}}, \mathbf{v} \notin \widetilde{VO}^{\tau}_{\mathcal{R}} \setminus \partial \widetilde{VO}^{\tau}_{\mathcal{R}} \right\}. \tag{4.53}$$

Hence, the robot has to select $\mathbf{v}^{new}_{\mathcal{R}}$ in $RAV_{\mathcal{R}}$.

Preferred linear velocity

If the robot cannot reach the goal in the next time period, the preferred linear velocity $\mathbf{v}^{pref}_{\mathcal{R}}$ has a magnitude of the robot's preferred linear speed $v^{pref}_{\mathcal{R}}$ and is directed toward $\mathbf{p}^{goal}_{\mathcal{R}}$:

$$\mathbf{v}^{pref}_{\mathcal{R}} = \frac{v^{pref}_{\mathcal{R}} \left(\mathbf{p}^{goal}_{\mathcal{R}} - \mathbf{p}_{\mathcal{R}} \right)}{\max \left\{ v^{pref}_{\mathcal{R}} \Delta t, \left\| \mathbf{p}^{goal}_{\mathcal{R}} - \mathbf{p}_{\mathcal{R}} \right\| \right\}}. \tag{4.54}$$

New linear velocity selection

Problem 4.2 is equivalent to finding the closest velocity to the preferred one in $RAV_{\mathcal{R}}$:

$$\mathbf{v}_{\mathcal{R}}^{new} = \underset{\mathbf{v} \in RAV_{\mathcal{R}}}{\arg\min} \left\| \mathbf{v} - \mathbf{v}_{\mathcal{R}}^{pref} \right\|. \tag{4.55}$$

However, solving the problem of (4.55) is computationally demanding because it has the non-convex domain $RAV_{\mathcal{R}}$. Therefore, the ClearPath efficient geometric algorithm introduced in [53] is employed.

If $\mathbf{v}_{\mathcal{R}}^{pref} \notin \widetilde{VO}_{\mathcal{R}} \setminus \partial\widetilde{VO}_{\mathcal{R}}$, then $\mathbf{v}_{\mathcal{R}}^{pref} \in RAV_{\mathcal{R}}$ and $\mathbf{v}_{\mathcal{R}}^{new} = \mathbf{v}_{\mathcal{R}}^{pref}$. Otherwise, $\mathbf{v}_{\mathcal{R}}^{new}$ is contained in $\partial\widetilde{VO}_{\mathcal{R}}$. Since $\partial\widetilde{VO}_{\mathcal{R}}$ is the union of line segments, the candidates of $\mathbf{v}_{\mathcal{R}}^{new}$ is classified into three groups:

- VC_A is a set of projections of $\mathbf{v}_{\mathcal{R}}^{pref}$ onto $\partial\widetilde{VO}_{\mathcal{R}}$,
- VC_B is a set of cross velocities between the line segments of $\partial\widetilde{VO}_{\mathcal{R}}$,
- VC_C is a set of cross velocities between $\partial RV_{\mathcal{R}}$ and $\partial\widetilde{VO}_{\mathcal{R}}$.

Figure 4.8 presents the combined EBVO $\widetilde{VO}_{\mathcal{R}}$, the region of $RAV_{\mathcal{R}}$, and the set of candidates of $\mathbf{v}_{\mathcal{R}}^{new}$, which is $VC = \{VC_A, VC_B, VC_C\}$. The white, gray, and black marks represent VC_A, VC_B, and VC_C, respectively. Among those points, the closest point to $\mathbf{v}_{\mathcal{R}}^{pref}$ is the solution of (4.55).

The candidates are again divided into two parts: VC_h and VC_l. The set VC_h is defined by

$$VC_h = VC \cap \bigcup_{i \leq N_{\mathcal{O}}} \partial VO_{\mathcal{R}|\mathcal{O}_i}^{\infty}, \tag{4.56}$$

and the set VC_l is defined by

$$VC_l = VC \setminus VC_h. \tag{4.57}$$

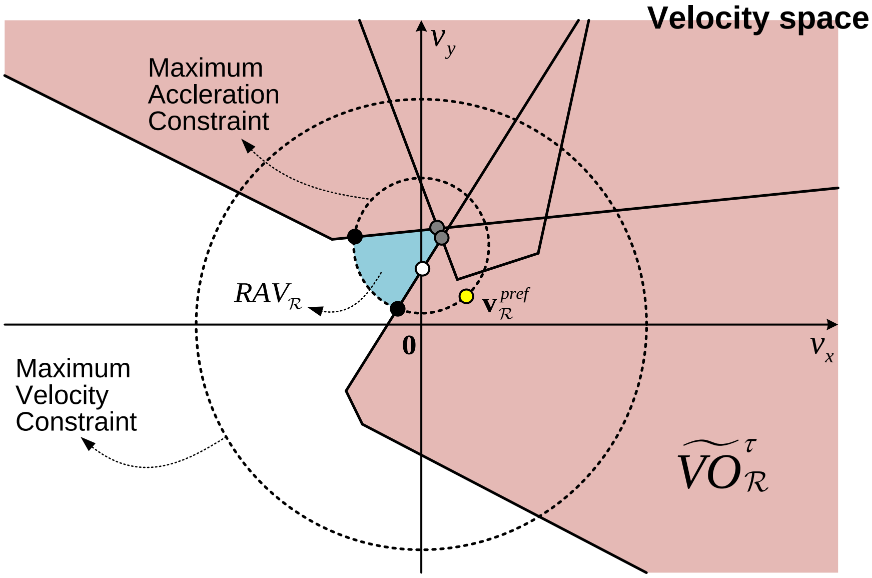


Figure 4.8 The selection of the new linear velocity $\mathbf{v}_{\mathcal{R}}^{new}$ in the velocity space. The red region represents $\widetilde{VO}_{\mathcal{R}}^{\tau}$, the dotted circles indicate the robot's dynamic constraints, and the blue region represents RAV. The yellow mark is $\mathbf{v}_{\mathcal{R}}^{pref}$. The white, gray, and black marks are included in the groups VC_A, VC_B, and VC_C, respectively. In this case, the white mark is selected to $\mathbf{v}_{\mathcal{R}}^{new}$.

If the robot moves with a velocity in VC_h, the robot grazes the obstacles and completely avoid the collisions with them. On the other hand, if the robot moves with a velocity in VC_l, the robot will eventually collide with some of the obstacles after τ. For this reason, we give a high priority to velocities in VC_h when $\mathbf{v}_{\mathcal{R}}^{new}$ is determined. As a result, when $\mathbf{v}_{\mathcal{R}}^{pref} \in \widetilde{VO}_{\mathcal{R}} \setminus \partial\widetilde{VO}_{\mathcal{R}}$, $\mathbf{v}_{\mathcal{R}}^{new}$ is decided as follows:

$$\mathbf{v}_{\mathcal{R}}^{new} = \begin{cases} \underset{\mathbf{v}\in VC_h}{\arg\min} \left\| \mathbf{v} - \mathbf{v}_{\mathcal{R}}^{pref} \right\|, & \text{if } VC_h \neq \emptyset, \\ \underset{\mathbf{v}\in VC_l}{\arg\min} \left\| \mathbf{v} - \mathbf{v}_{\mathcal{R}}^{pref} \right\|, & \text{if } VC_h = \emptyset. \end{cases} \tag{4.58}$$

If the problem of (4.48) is infeasible (for example, due to a short sampling period or densely packed conditions), the time horizon τ decreases until the feasible solution is guaranteed.

4.6 Collision-free Rotation Angles

In this and the next sections, we deal with Problem 4.3, where the angular velocity $w_{\mathcal{R}}^{new}$ is determined when the robot is moving with the linear velocity $\mathbf{v}_{\mathcal{R}}^{new}$ computed in Section 4.5. The objective of the rotation is to change its orientation so that the robot circumvents obstacles with the minimum deviation from the shortest path to the goal. If $\mathbf{v}_{\mathcal{R}}^{new} \in RAV_{\mathcal{R}} \setminus \partial\widetilde{VO}_{\mathcal{R}}$, the rotation is unnecessary because the robot $\mathcal{R}$ already moves along the shortest one. On the other hand, if $\mathbf{v}_{\mathcal{R}}^{new} \notin RAV_{\mathcal{R}} \setminus \partial\widetilde{VO}_{\mathcal{R}}$, the robot have to change its orientation to increase the efficiency of the collision avoidance. Hence, the latter condition is only considered in this section.

When $\mathbf{v}_{\mathcal{R}}^{new} \notin RAV_{\mathcal{R}} \setminus \partial\widetilde{VO}_{\mathcal{R}}$, the new linear velocity is selected in order to avoid the obstacle the robot first grazes when it maintains the current velocity for a period of time. In other words, the objective of the angular motion planning is to change its orientation until it contacts that obstacle. Hence, the shortest time the robot takes to contact obstacles is first calculated, and then the collision-free interval of the rotation angles before the time-to-contact is computed.

4.6.1 The Shortest Time-to-contact

In this subsection, we find the obstacle $\mathcal{O}_F$ that the robot first contacts when it maintains its new velocity $\mathbf{v}_{\mathcal{R}}^{new}$ and the time T_F it takes to contact $\mathcal{O}_F$.

Since it is assumed that $\mathbf{v}_{\mathcal{R}}^{new} \in \partial \widetilde{VO}_{\mathcal{R}}$, $\mathbf{v}_{\mathcal{R}}^{new}$ belongs to either VC_A, VC_B, or VC_C defined in Section 4.5. If $\mathbf{v}_{\mathcal{R}}^{new} \in VC_A \cup VC_C$, the robot will contact only one obstacle, which is $\mathcal{O}_F$. Furthermore, if $\mathbf{v}_{\mathcal{R}}^{new} \in \Gamma_{\mathcal{R}|\mathcal{O}_F}$, the time-to-contact is $T_F = \tau$. Otherwise, without loss of generality, suppose the robot pass by the left side of $\mathcal{O}_F$ at a future time, that is $\mathbf{v}_{\mathcal{R}}^{new} \in l_{\mathcal{R}|\mathcal{O}_F}^{\text{left}}$. Then T_F is

$$T_F = \min \left\{ \frac{\left\| \mathbf{q}_{\mathcal{R}|\mathcal{O}_F}^{\text{left}} \right\|}{\left\| \mathbf{v}_{\mathcal{R}}^{new} - \mathbf{v}_{\mathcal{O}_F} \right\|}, \tau \right\}. \tag{4.59}$$

Because τ is the maximum time for which the collision is considered, T_F is limited below τ.

If $\mathbf{v}_{\mathcal{R}}^{new} \in VC_B$, the robot will contact two obstacles, denoted by $\mathcal{O}_{F,1}$ and $\mathcal{O}_{F,2}$. For each of them, the time-to-contact $T_{F,i}$ is calculated from (4.59). Afterward, $\mathcal{O}_F = \mathcal{O}_{F,i}$ and $T_F = T_{F,i}$ for $i = \arg\min \{T_{F,1}, T_{F,2}\}$.

4.6.2 Collision-free Interval of the Rotation Angles

With respect to a local frame attached to $\mathcal{R}$, the robot is rotating at the origin and the obstacle is moving with the velocity $\mathbf{v}_{\mathcal{R}|\mathcal{O}} = \mathbf{v}_{\mathcal{O}} - \mathbf{v}_{\mathcal{R}}^{new}$ at $\mathbf{p}_{\mathcal{R}|\mathcal{O}} = \mathbf{p}_{\mathcal{O}} - \mathbf{p}_{\mathcal{R}}$ at time t_0. Let $\Delta\theta_{\mathcal{R}|\mathcal{O}}$ be the rotation angle of the robot and $\Theta_{\mathcal{R}|\mathcal{O}}^{T_F} = \left[\Delta\theta_{\mathcal{R}|\mathcal{O}}^{-}, \Delta\theta_{\mathcal{R}|\mathcal{O}}^{+}\right]$ be the set of collision-free rotation angles of the robot within T_F. Here, $\Delta\theta_{\mathcal{R}|\mathcal{O}}^{-} \leq 0$ and $\Delta\theta_{\mathcal{R}|\mathcal{O}}^{+} \geq 0$ because a collision does not occur when $\Delta\theta_{\mathcal{R}|\mathcal{O}} = 0$ since the new velocity is selected on the assumption that the robot does not rotate. The objective of this subsection is to find out the interval $\Theta_{\mathcal{R}|\mathcal{O}}^{T_F}$.

As a preliminary step, the collision-free rotation angles are calculated for a fixed elliptic obstacle and a fixed line segment obstacle. Next, the collision-free interval induced by a moving elliptic obstacle is calculated based on the results.

Fixed elliptic obstacle

Suppose that there are a rotating elliptic robot $\mathcal{R}$ at the origin and an elliptic obstacle fixed at $\mathbf{p}_{\mathcal{R}|\mathcal{O}}$, which are initially separated each other. If $\mathbf{p}_{\mathcal{R}|\mathcal{O}} \notin QO_{D(\mathbf{p}_{\mathcal{R}},M_{\mathcal{R}})|\mathcal{O}} \setminus \partial QO_{D(\mathbf{p}_{\mathcal{R}},M_{\mathcal{R}})|\mathcal{O}}$, where $M_{\mathcal{R}}$ is the major radius of the robot and $D(\mathbf{p}, r)$ is the disk located at $\mathbf{p}$ with radius of r as defined in (3.11), then $\Theta^{T_F}_{\mathcal{R}|\mathcal{O}} = [-\pi, \pi]$. Otherwise, the collision-free interval is derived from the algebraic condition in Section 3.4.

The coefficient matrices of the robot and the obstacle are represented by

$$\mathbf{M}_{\mathcal{R}}(\Delta\theta_{\mathcal{R}}) = \begin{bmatrix} \mathbf{R}_{\Delta\theta_{\mathcal{R}}}\mathbf{S}_{\mathcal{R}}(\theta_{\mathcal{R}})^{-1}\mathbf{R}^{T}_{\Delta\theta_{\mathcal{R}}} & \mathbf{0} \\ \mathbf{0} & -1 \end{bmatrix}, \tag{4.60}$$

$$\mathbf{M}_{\mathcal{O}} = \begin{bmatrix} \mathbf{S}_{\mathcal{O}}^{-1} & -\mathbf{S}_{\mathcal{O}}^{-1}\mathbf{p}_{\mathcal{R}|\mathcal{O}} \\ -\mathbf{p}^{T}_{\mathcal{R}|\mathcal{O}}\mathbf{S}_{\mathcal{O}}^{-1} & \mathbf{p}^{T}_{\mathcal{R}|\mathcal{O}}\mathbf{S}_{\mathcal{O}}^{-1}\mathbf{p}_{\mathcal{R}|\mathcal{O}} - 1 \end{bmatrix}. \tag{4.61}$$

Then the characteristic polynomial of the robot and the obstacle is

$$g(\xi) = a_3\xi^3 + a_2\xi^2 + a_1\xi + a_0, \tag{4.62}$$

where

$$a_3 = -\det \mathbf{S}_{\mathcal{R}}(\theta_{\mathcal{R}})^{-1}, \tag{4.63}$$

$$a_2 = a_{22}\cos 2\Delta\theta_{\mathcal{R}} + a_{21}\sin 2\Delta\theta_{\mathcal{R}} + a_{20}, \tag{4.64}$$

$$a_1 = a_{12}\cos 2\Delta\theta_{\mathcal{R}} + a_{11}\sin 2\Delta\theta_{\mathcal{R}} + a_{10}, \tag{4.65}$$

$$a_0 = -\det \mathbf{S}_{\mathcal{O}}^{-1}. \tag{4.66}$$

The values of a_{22}, a_{21}, a_{20}, a_{12}, a_{11}, and a_{10} are obtained by putting $\bar{\mathbf{S}}_{\mathcal{R}} =$

$\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)$ into (3.25)–(3.30). From Theorem 3.18, they do not overlap if and only if

$$\begin{cases} a_2 \geq 0 \quad \text{or} \quad 3a_3a_2a_0 - 4a_3a_1^2 + a_2^2a_1 \geq 0, \\ 3a_3a_1 - a_2^2 < 0, \\ 27a_3^2a_0^2 - 18a_3a_2a_1a_0 + 4a_3a_1^3 + 4a_2^3a_0 - a_2^2a_1^2 \leq 0. \end{cases} \tag{4.67}$$

In order to solve the four inequalities in (4.67), put

$$\cos 2\Delta\theta_{\mathcal{R}} = \frac{1-t^2}{1+t^2}, \tag{4.68}$$

$$\sin 2\Delta\theta_{\mathcal{R}} = \frac{2t}{1+t^2}. \tag{4.69}$$

As a result, four polynomial inequalities of degree 2, 6, 4, and 8 are obtained, respectively. Theses inequalities can be solved by finding the roots of the polynomials, and how to solve them is described in Section 4.2. Among these roots, even multiple roots are eliminated because the graph does not cross the axis at those points. For each of the inequalities, all the roots t_i are reversely mapped into the domain of the rotation angles as follows:

$$\Delta\theta_{\mathcal{R}|\mathcal{O},2i-1} = \frac{\operatorname{atan2}\left(2t, 1-t^2\right)}{2}, \tag{4.70}$$

$$\Delta\theta_{\mathcal{R}|\mathcal{O},2i} = \frac{\operatorname{atan2}\left(2t, 1-t^2\right)}{2} - \pi\operatorname{sgn}\left(t\right). \tag{4.71}$$

With the above values, the interval of collision-free rotation angles is determined. For the inequalities $3a_3a_1 - a_2^2 < 0$ and $27a_3^2a_0^2 - 18a_3a_2a_1a_0 + 4a_3a_1^3 + 4a_2^3a_0 - a_2^2a_1^2 \leq 0$, the intervals have endpoints at the minimum positive and the maximum negative values because $\Delta\theta_{\mathcal{R}|\mathcal{O}} = 0$ is always included. However, the interval that satisfy $a_2 \geq 0$ or $3a_3a_2a_0 - 4a_3a_1^2 + a_2^2a_1 \geq 0$ is determined after monitoring the sign of all the interval.

In conclusion, the interval of the collision free rotation angles of the robot $\mathcal{R}$ induced by the obstacle $\mathcal{O}$, $\Theta_{\mathcal{R}|\mathcal{O}}^{T_F} = \left[\Delta\theta_{\mathcal{R}|\mathcal{O}}^{-}, \Delta\theta_{\mathcal{R}|\mathcal{O}}^{+}\right]$ is determined by intersecting the three intervals.

Fixed line segment obstacle

Suppose that there are a rotating elliptic robot $\mathcal{R}$ at the origin and a line segment obstacle $\mathcal{O}$ whose end points are $\mathbf{p}_{\mathcal{R}|\mathcal{O}}$ and $\mathbf{p}_{\mathcal{R}|\mathcal{O}} + T_F \mathbf{v}_{\mathcal{R}|\mathcal{O}}$, that is $\mathcal{F}_{\mathcal{O}} = l\left(\mathbf{p}_{\mathcal{R}|\mathcal{O}}, \mathbf{v}_{\mathcal{R}|\mathcal{O}}, T_F\right)$. Also, suppose that they are initially separated each other. Then the distance between the line segment and the origin is given by

$$d_{\mathcal{R}|\mathcal{O}} = \left\|\mathbf{p}_{\mathcal{R}|\mathcal{O}} + \zeta \mathbf{v}_{\mathcal{R}|\mathcal{O}}\right\|, \tag{4.72}$$

where

$$\zeta = \begin{cases} T_F & \text{if } \mathbf{p}_{\mathcal{R}|\mathcal{O}}^T \mathbf{v}_{\mathcal{R}|\mathcal{O}} \le -T_F \mathbf{v}_{\mathcal{R}|\mathcal{O}}^T \mathbf{v}_{\mathcal{R}|\mathcal{O}}, \\ -\frac{\mathbf{p}_{\mathcal{R}|\mathcal{O}}^T \mathbf{v}_{\mathcal{R}|\mathcal{O}}}{\mathbf{v}_{\mathcal{R}|\mathcal{O}}^T \mathbf{v}_{\mathcal{R}|\mathcal{O}}} & \text{if } -T_F \mathbf{v}_{\mathcal{R}|\mathcal{O}}^T \mathbf{v}_{\mathcal{R}|\mathcal{O}} < \mathbf{p}_{\mathcal{R}|\mathcal{O}}^T \mathbf{v}_{\mathcal{R}|\mathcal{O}} \le 0, \\ 0 & \text{if } \mathbf{p}_{\mathcal{R}|\mathcal{O}}^T \mathbf{v}_{\mathcal{R}|\mathcal{O}} > 0. \end{cases} \tag{4.73}$$

If $d_{\mathcal{R}|\mathcal{O}} \ge M_{\mathcal{R}}$, the robot is free of collision and $\Theta_{\mathcal{R}|\mathcal{O}}^{T_F} = [-\pi, \pi]$. Otherwise, the interval $\Theta_{\mathcal{R}|\mathcal{O}}^{T_F}$ is obtained through the next three steps. First, the interval of the collision-free rotation between $\mathcal{R}$ and the line containing $\mathcal{O}$, which is

$$d_{line} = \left\|\mathbf{p}_{\mathcal{R}|\mathcal{O}} - \frac{\mathbf{p}_{\mathcal{R}|\mathcal{O}}^T \mathbf{v}_{\mathcal{R}|\mathcal{O}}}{\mathbf{v}_{\mathcal{R}|\mathcal{O}}^T \mathbf{v}_{\mathcal{R}|\mathcal{O}}} \mathbf{v}_{\mathcal{R}|\mathcal{O}}\right\| \tag{4.74}$$

away from the origin, is calculated.

Lemma 4.13 *Consider a rotating ellipse $\mathcal{R}$ at the origin and a line d_{line} far from the origin. Then the point $\mathbf{x} \in \partial\mathcal{F}_{\mathcal{R}}$ will contact the line if*

$$\mathbf{x}^T \mathbf{S}_{\mathcal{R}} \left(\theta_{\mathcal{R}}\right)^{-2} \mathbf{x} = \frac{1}{d_{line}^2}. \tag{4.75}$$

Proof. For $\mathbf{x} \in \partial\mathcal{F}_{\mathcal{R}}$, $f\left(\mathbf{x}; \mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right), \mathbf{0}\right) = \mathbf{x}^T \mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)^{-1} \mathbf{x} - 1 = 0$ holds from (1.2). Suppose the ellipse touches the line when it rotates by $\Delta\theta$. Then $\mathbf{x}^* = \mathbf{R}_{\Delta\theta}\mathbf{x}$ belongs to the line and its normal vector $\mathbf{n}^* = \mathbf{R}_{\Delta\theta}\mathbf{S}_{\mathcal{R}}^{-1}\mathbf{x}$ is perpendicular to the line, which implies

$$d_{line} = \frac{\mathbf{x}^{*T}\mathbf{n}^*}{\|\mathbf{n}^*\|} = \frac{1}{\sqrt{\mathbf{x}^T \mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)^{-2} \mathbf{x}}}. \tag{4.76}$$

Hence, (4.75) follows. □

Since $\mathbf{x} = \mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)^{\frac{1}{2}} \mathbf{u}$ can be parameterized by a unit vector $\mathbf{u}$, (4.75) can be written as $\mathbf{u}^T \mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)^{-1} \mathbf{u} = 1/d_{line}^2$. This can be solved by putting $\mathbf{u} = \left[1 - t^2 \quad 2t\right]^T / \left(1 + t^2\right)$. As a result, we obtain candidate points $\mathbf{x}_i = \mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)^{\frac{1}{2}} \mathbf{u}_i$ and the corresponding normal vectors $\mathbf{n}_i = \mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)^{-\frac{1}{2}} \mathbf{u}_i$ for $i = 1, \ldots, 4$. As a result, the robot touches the line when it rotates by angles of $\Delta\theta_i$ such that, for $\mu > 0$,

$$\mathbf{p}_{\mathcal{R}|\mathcal{O}} - \frac{\mathbf{p}_{\mathcal{R}|\mathcal{O}}^T \mathbf{v}_{\mathcal{R}|\mathcal{O}}}{\mathbf{v}_{\mathcal{R}|\mathcal{O}}^T \mathbf{v}_{\mathcal{R}|\mathcal{O}}} \mathbf{v}_{\mathcal{R}|\mathcal{O}} = \mu \mathbf{R}_{\Delta\theta_i}^T \mathbf{n}_i. \tag{4.77}$$

Let $\Delta\theta_-$ and $\Delta\theta_+$ be the maximum negative and the minimum positive values among $\Delta\theta_i$ for $i = 1, \ldots, 4$. Hence, the interval $\Theta_{\mathcal{R}|\mathcal{O},1}^{T_F} = \left[\Delta\theta_{\mathcal{R}|\mathcal{O},1}^-, \Delta\theta_{\mathcal{R}|\mathcal{O},1}^+\right]$ is determined as follows:

$$\Delta\theta^{-}_{\mathcal{R}|\mathcal{O},1} = \begin{cases} \Delta\theta_{-} & \text{if } \mathbf{R}_{\Delta\theta_{-}}\mathbf{x}_{-} \in \mathcal{F}_{\mathcal{O}}, \\ -\pi & \text{otherwise}, \end{cases} \tag{4.78}$$

$$\Delta\theta^{+}_{\mathcal{R}|\mathcal{O},1} = \begin{cases} \Delta\theta_{+} & \text{if } \mathbf{R}_{\Delta\theta_{+}}\mathbf{x}_{+} \in \mathcal{F}_{\mathcal{O}}, \\ \pi & \text{otherwise}. \end{cases} \tag{4.79}$$

In the above equations, it is neglected if the robot does not contact with the obstacle with the rotation of $\Delta\theta_{-}$ or $\Delta\theta_{-}$, which is complemented in the next.

Second, the collision-free intervals induced by the two endpoints are calculated. For $\mathbf{p}_{\mathcal{R}|\mathcal{O}}$, it follows that $\|\mathbf{x}\| = \left\|\mathbf{p}_{\mathcal{R}|\mathcal{O}}\right\|$. Writing $\mathbf{x} = \mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)^{\frac{1}{2}}\mathbf{u}$,

$$\mathbf{u}^{T}\mathbf{S}_{\mathcal{R}}\left(\theta_{\mathcal{R}}\right)\mathbf{u} = \mathbf{p}^{T}_{\mathcal{R}|\mathcal{O}}\mathbf{p}_{\mathcal{R}|\mathcal{O}}. \tag{4.80}$$

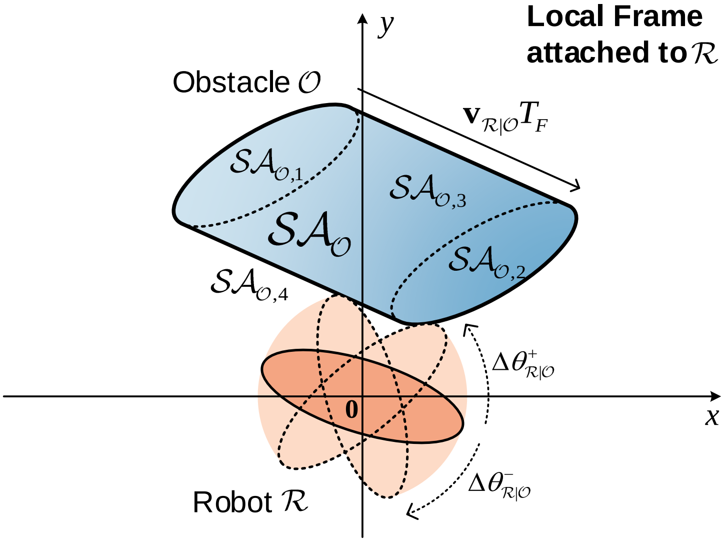
This can be solved in the similar way as before, and we can obtain the interval $\Theta^{T_F}_{\mathcal{R}|\mathcal{O},2}$ induced by the point $\mathbf{p}_{\mathcal{R}|\mathcal{O}}$. Likewise, we can get the interval $\Theta^{T_F}_{\mathcal{R}|\mathcal{O},3}$ induced by the point $\mathbf{p}_{\mathcal{R}|\mathcal{O}} + T_F\mathbf{v}_{\mathcal{R}|\mathcal{O}}$.

Finally, $\Theta^{T_F}_{\mathcal{R}|\mathcal{O}}$, the collision-free interval induced by $\mathcal{O}$, is computed as

$$\Theta^{T_F}_{\mathcal{R}|\mathcal{O}} = \bigcap_{i=1,2,3} \Theta^{T_F}_{\mathcal{R}|\mathcal{O},i}. \tag{4.81}$$

Moving elliptic obstacle

Suppose that there are a rotating elliptic robot $\mathcal{R}$ at the origin and an elliptic obstacle $\mathcal{O}$ moving with the velocity $\mathbf{v}_{\mathcal{R}|\mathcal{O}}$ at $\mathbf{p}_{\mathcal{R}|\mathcal{O}}$ at time t_0. Let $\mathcal{SA}_{\mathcal{O}}$ denote



y
Local Frame attached to $\mathcal{R}$
Obstacle $\mathcal{O}$
$\mathbf{v}_{\mathcal{R}|\mathcal{O}}T_F$
$\mathcal{SA}_{\mathcal{O},1}$
$\mathcal{SA}_{\mathcal{O},3}$
$\mathcal{SA}_{\mathcal{O}}$
$\mathcal{SA}_{\mathcal{O},2}$
$\mathcal{SA}_{\mathcal{O},4}$
$\Delta\theta^{+}_{\mathcal{R}|\mathcal{O}}$
$\mathbf{0}$
x
$\Delta\theta^{-}_{\mathcal{R}|\mathcal{O}}$
Robot $\mathcal{R}$

Figure 4.9 The collision-free rotation angles of the robot $\mathcal{R}$ in time interval from $t = t_0$ to $t = t_0 + T_F$. The swept region by $\mathcal{O}$ is represented by the region $\mathcal{SA}_{\mathcal{O}}$. The obstacle $\mathcal{O}$ is $\mathcal{SA}_{\mathcal{O},1}$ at t_0 and $\mathcal{SA}_{\mathcal{O},2}$ at $t_0 + T_F$. The line segments $\mathcal{SA}_{\mathcal{O},3}$ and $\mathcal{SA}_{\mathcal{O},4}$ are the boundary lines of $\mathcal{SA}_{\mathcal{O}}$. The red region indicates the collision-free rotation angles, $\Theta^{T_F}_{\mathcal{R}|\mathcal{O}} = \left[\Delta\theta^{-}_{\mathcal{R}|\mathcal{O}}, \Delta\theta^{+}_{\mathcal{R}|\mathcal{O}}\right]$.

the area swept by the obstacle $\mathcal{O}$ for $[t_0, t_0 + T_F]$ such that

$$\mathcal{SA}_{\mathcal{O}} = \bigcup_{t\in[0,T_F]} \left\{\mathbf{x} \in \mathbb{R}^2 \,\middle|\, f\left(\mathbf{x}; \mathbf{S}_{\mathcal{O}}, \mathbf{p}_{\mathcal{R}|\mathcal{O}} + t\mathbf{v}_{\mathcal{R}|\mathcal{O}}\right) \leq 0\right\}. \tag{4.82}$$

As shown in Fig. 4.9, its boundary $\partial\mathcal{SA}_{\mathcal{O}}$ consists of segments of two ellipses and two lines. More specifically, the two ellipses are

$$\mathcal{SA}_{\mathcal{O},1} = \left\{\mathbf{x} \in \mathbb{R}^2 \,\middle|\, f\left(\mathbf{x}; \mathbf{S}_{\mathcal{O}}, \mathbf{p}_{\mathcal{R}|\mathcal{O}}\right) \leq 0\right\}, \tag{4.83}$$

$$\mathcal{SA}_{\mathcal{O},2} = \left\{\mathbf{x} \in \mathbb{R}^2 \,\middle|\, f\left(\mathbf{x}; \mathbf{S}_{\mathcal{O}}, \mathbf{p}_{\mathcal{R}|\mathcal{O}} + T_F\mathbf{v}_{\mathcal{R}|\mathcal{O}}\right) \leq 0\right\}, \tag{4.84}$$

and the two line segments are, for $i = 3, 4$,

$$\mathcal{SA}_{\mathcal{O},i} = l\left(\mathbf{p}_{\mathcal{R}|\mathcal{O}} \pm \frac{\mathbf{S}_{\mathcal{O}}\mathbf{v}_{\mathcal{R}|\mathcal{O}}}{\mathbf{p}_{\mathcal{R}|\mathcal{O}}^T\mathbf{S}_{\mathcal{O}}\mathbf{v}_{\mathcal{R}|\mathcal{O}}}, \mathbf{v}_{\mathcal{R}|\mathcal{O}}, T_F\right). \tag{4.85}$$

The collision-free intervals of the rotation angles induced by the fixed elliptic obstacles $\mathcal{SA}_{\mathcal{O},i}$ for $i = 1, 2$, denoted by $\Theta_{\mathcal{R}|\mathcal{SA}_{\mathcal{O},i}}^{T_F}$ and those induced by the fixed line segment obstacles $\mathcal{SA}_{\mathcal{O}_i}$ for $i = 3, 4$, $\Theta_{\mathcal{R}|\mathcal{T}_{\mathcal{O},i}}^{T_F}$, can be derived as mentioned before. Finally, the collision-free rotation angles induced by $\mathcal{O}$ is computed by

$$\Theta_{\mathcal{R}|\mathcal{O}}^{T_F} = \bigcap_{i\in\{1,\ldots,4\}} \Theta_{\mathcal{R}|\mathcal{SA}_{\mathcal{O},i}}^{T_F}. \tag{4.86}$$

4.7 Selection of Collision-free Angular Velocity

In this section, a strategy to determine the new angular velocity is presented by utilizing the interval of the collision-free rotation angles. First, it is explained how the preferred angular velocities are determined. Next, the selection of the new angular velocity is described in detail.

4.7.1 Preferred Angular Velocities

In this subsection, it is assumed that $\mathbf{v}_{\mathcal{R}}^{new} \notin RAV_{\mathcal{R}} \setminus \partial\widetilde{VO}_{\mathcal{R}}$. As mentioned in Section 1.2, the preferred angular velocities are determined to lead the robot to potentially circumvent obstacles with the minimum possible deviation. Because the new linear velocity is selected to avoid the obstacle $\mathcal{O}_F$ it first contacts when it maintains the current velocity for a period of time, the robot is expected to reach its goal with a less traveled path by changing its orientation to make the robot avoid the obstacle $\mathcal{O}_F$ more efficiently.

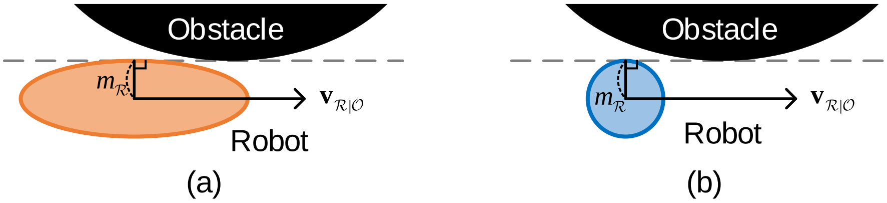

Obstacle
$m_{\mathcal{R}}$
$\mathbf{v}_{\mathcal{R}|\mathcal{O}}$
Robot
(a)
Obstacle
$m_{\mathcal{R}}$
$\mathbf{v}_{\mathcal{R}|\mathcal{O}}$
Robot
(b)

Figure 4.10 A basic concept of determining preferred angular velocities. (a) An elliptic robot whose orientation is aligned with $\mathbf{v}_{\mathcal{R}|\mathcal{O}}$ can avoid obstacles with the same path as (b) a circular robot with radius equal to minor radius of the elliptic one does.

Corollary 3.4 clarifies the robot $\mathcal{R}$ with rotation can circumvent an obstacle as if it is a circular robot with radius equal to its minor radius, as shown in Fig. 4.10. Hence, the approximated EBVO $\widetilde{VO}^{\tau}_{D(\mathbf{p}_{\mathcal{R}},m_{\mathcal{R}})|\mathcal{O}_F}$ for the circular robot $D\left(\mathbf{p}_{\mathcal{R}}, m_{\mathcal{R}}\right)$ induced by the obstacle $\mathcal{O}_F$. If $\mathbf{v}^{pref}_{\mathcal{R}} \in \widetilde{VO}^{\tau}_{D(\mathbf{p}_{\mathcal{R}},m_{\mathcal{R}})|\mathcal{O}_F}$, the closest velocity $\mathbf{v}^*$ to $\mathbf{v}^{pref}_{\mathcal{R}}$ is selected outside $\widetilde{VO}^{\tau}_{\mathcal{R}_C|\mathcal{O}_F}$. Then the preferred angular orientations are aligned with the direction of $\mathbf{v}^* - \mathbf{v}_{\mathcal{O}_F}$. Due to symmetry of an ellipse, there are two preferred angular orientations: one is positive and the other is negative. Hence, the preferred angular velocities are defined as

$$w^{pref}_{\mathcal{R},\pm} = \frac{\text{wrapToPi}\left(\angle\left(\pm\left(\mathbf{v}^* - \mathbf{v}_{\mathcal{O}_F}\right)\right) - \theta_{\mathcal{R}}\right)}{\Delta t}, \tag{4.87}$$

where $\angle\left(\mathbf{v}\right)$ is the direction angle of a given vector $\mathbf{v}$ and $\text{wrapToPi}\left(\theta\right)$ is a function wrapping a given value θ to $[-\pi, \pi]$.

If $\mathbf{v}^{pref}_{\mathcal{R}} \notin \widetilde{VO}^{\tau}_{D(\mathbf{p}_{\mathcal{R}},m_{\mathcal{R}})|\mathcal{O}_F}$, the robot is expected to move with $\mathbf{v}^{pref}_{\mathcal{R}}$ at the next sampling time provided that its orientation changes adequately. To find the preferred orientations, the distance d_{line} between the obstacle's position $\mathbf{p}_{\mathcal{O}_F}$ and the line connecting $\mathbf{p}_{\mathcal{R}}$ and $\mathbf{p}^{goal}_{\mathcal{R}}$ is calculated by substituting $\mathbf{v}_{\mathcal{R}|\mathcal{O}} = \mathbf{v}^{pref}_{\mathcal{R}} - \mathbf{v}_{\mathcal{O}_F}$ and $\mathbf{p}_{\mathcal{R}|\mathcal{O}} = \mathbf{p}_{\mathcal{R}} - \mathbf{p}_{\mathcal{O}_F}$ into (4.74).

Next, the preferred angular orientations are calculated from Lemma 4.13. Among the four angles, the minimum positive and the maximum negative ones, $\Delta\theta^{pref}_{\mathcal{R},-}$ and $\Delta\theta^{pref}_{\mathcal{R},+}$, are assigned to the preferred angular orientations. Hence, the preferred angular velocities are defined as

$$w^{pref}_{\mathcal{R},\pm} = \frac{\Delta\theta^{pref}_{\mathcal{R},\pm}}{\Delta t}. \tag{4.88}$$

4.7.2 New Angular Velocity Selection

Based on the collision-free interval of the rotation angles in Section 4.6.2 and the preferred angular velocities in Section 4.7.1, the new angular velocity is determined.

Combined collision-free angular velocities interval

There are $N_{\mathcal{O}}$ obstacles $\mathcal{O}_1, \mathcal{O}_2, \cdots, \mathcal{O}_{N_{\mathcal{O}}}$ detected by the robot as mentioned in Section 4.2. For any natural number $i \leq N_{\mathcal{O}}$, the collision-free interval $\Theta^{T_F}_{\mathcal{R}|\mathcal{O}_i}$, can be derived. If $\Delta\theta^{new}_{\mathcal{R}} \notin \Theta^{T_F}_{\mathcal{R}|\mathcal{O}_i}$ for some $i \leq N_{\mathcal{O}}$, the collision occurs in time T_F. Therefore, the combined interval of collision-free rotating angles is defined by the intersection of $\Theta^{T_F}_{\mathcal{R}|\mathcal{O}_i}$ such that

$$\Theta^{T_F}_{\mathcal{R}} = \bigcap_{i \leq N_{\mathcal{O}}} \Theta^{T_F}_{\mathcal{R}|\mathcal{O}_i}. \tag{4.89}$$

If the robot moves with the velocity $\mathbf{v}^{new}_{\mathcal{R}}$ and rotates through an angle in $\Theta^{T_F}_{\mathcal{R}}$, the robot does not collide with the obstacles within T_F.

Because the objective is to determine the new angular velocity, it is essential to map $\Theta^{T_F}_{\mathcal{R}}$ to the domain of the robot's angular velocity. Moreover, the determined angular velocity is maintained only in the next sampling period Δt. Therefore, the collision-free interval of the robot's angular velocities is calculated by dividing the rotation angles with the sampling period Δt:

$$CFW_{\mathcal{R}} = \left\{ w \in \mathbb{R} \middle| w\Delta t \in \Theta_{\mathcal{R}}^{T_F} \right\}. \tag{4.90}$$

Reachable avoidance angular velocities

The available angular velocities during the next period are limited as the set of reachable angular velocities

$$RW_{\mathcal{R}} = \left\{ w \in \mathbb{R} \,|\, |w| \leq w_{\mathcal{R}}^{max}, |w - w_{\mathcal{R}}| \leq \alpha_{\mathcal{R}}^{max} \Delta t \right\}. \tag{4.91}$$

By intersecting $CFW_{\mathcal{R}}$ and $RW_{\mathcal{R}}$, the set of reachable avoidance angular velocities is denoted as

$$RAW_{\mathcal{R}} = CFW_{\mathcal{R}} \cap RW_{\mathcal{R}}. \tag{4.92}$$

Therefore, the robot has to select $w_{\mathcal{R}}^{new}$ in $RAW_{\mathcal{R}}$.

New angular velocity selection

Problem 4.3 is equivalent to finding the closest to $w_{\mathcal{R}}^{pref}$ in $RAW_{\mathcal{R}}$:

$$w_{\mathcal{R}}^{new} = \underset{w \in RAW_{\mathcal{R}}}{\arg\min} \min_{i=+,-} \left| w - w_{\mathcal{R},i}^{pref} \right|. \tag{4.93}$$

If both of the preferred angular velocities $w_{\mathcal{R},\pm}^{pref}$ are in $RAW_{\mathcal{R}}$, the closest to the current angular velocity $w_{\mathcal{R}}$ is selected. Figure 4.11 shows the selection of the new angular velocity through the optimization.

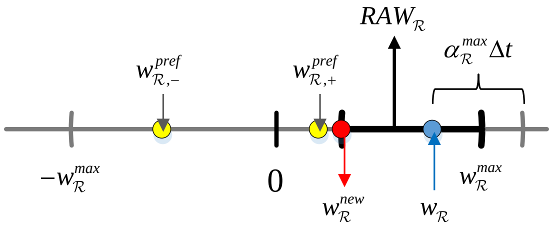


Figure 4.11 Optimization in the angular velocity space.

4.8 Summary

This chapter has suggested a local navigation algorithm for a holonomic elliptic robot, decomposed into two parts: linear and angular motion planning.

In the first part, the new linear velocity of an elliptic robot without rotation is selected in order to generate a trajectory to the goal without collision with obstacles. To this end, the ellipse-based velocity obstacle, the set of robot's linear velocities that would induce a collision with the obstacle within a finite time horizon, called EBVO, is derived. For the sake of efficiency, the EBVO is conservative approximated, so that the region of the forbidden velocities is bounded by three lines. Afterward, a strategy for selecting the new linear velocity outside of the approximated EBVO is presented.

In the second part, the new angular velocity of the robot moving with the linear velocity from the first part is selected to avoid obstacles with a less traveled path. The interval of the collision-free rotation angles is first derived by taking the swept region the obstacles may move through until it grazes some obstacle. Next, the preferred angular velocities are calculated that adjust its orientation to avoid obstacle as if it is a circular robot with radius of its minor radius. Finally, a strategy for selecting the new angular velocity is also presented based on the collision-free interval and the preferred angular velocities.

Chapter 5

Multi-Robot Collision Avoidance

5.1 Introduction

In the preceding chapter, the obstacle avoidance for an elliptic robot is considered on the assumption that obstacles maintain their velocities for a moment. In contrast, it is invalid on the collision avoidance between robots since the motion of robots constantly changes based on the sensor input from their environment. If one robot tries to avoid another robot with the prediction that it keeps moving with its current velocity, an oscillation occurs.

To remedy this problem, the concept of reciprocity was presented in [118], where robots take half of the responsibility for avoiding collisions with one another. Since it is assumed that all the robots use the same algorithm to plan their motions, one robot can predict the next velocities of other robots. Therefore, the method for obstacle avoidance presented in Chapter 4 is extended to that for multi-robot collision avoidance.

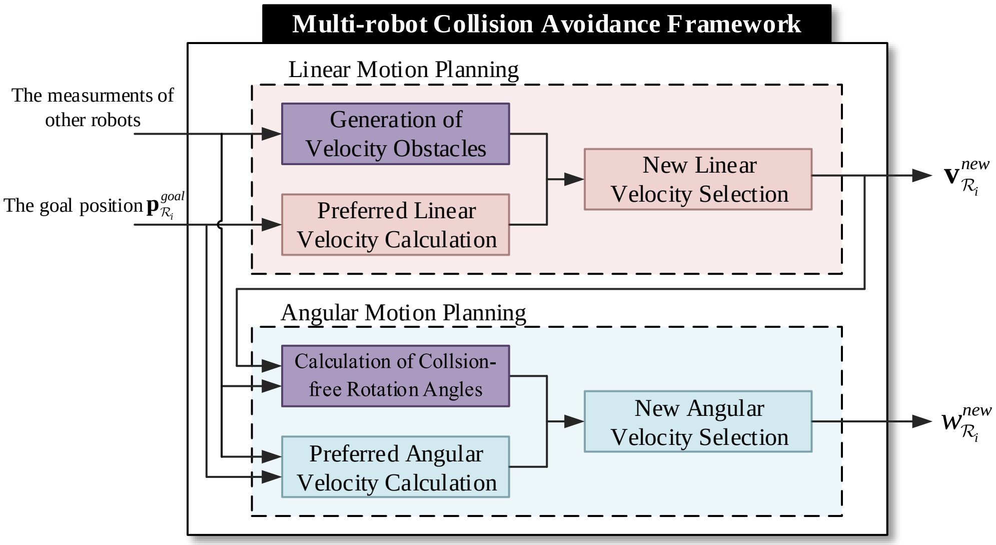

Multi-robot Collision Avoidance Framework
Linear Motion Planning
The measurments of other robots
The goal position $\mathbf{p}_{\mathcal{R}_i}^{goal}$
Generation of Velocity Obstacles
Preferred Linear Velocity Calculation
New Linear Velocity Selection
$\mathbf{v}_{\mathcal{R}_i}^{new}$
Angular Motion Planning
Calculation of Collsion-free Rotation Angles
Preferred Angular Velocity Calculation
New Angular Velocity Selection
$w_{\mathcal{R}_i}^{new}$

Figure 5.1 The proposed multi-robot collision avoidance framework for elliptic robots. The modified steps from the obstacle avoidance are presented in purple.

Since the model for predicting the movement of others is different from the previous chapter, the steps of determining potential collisions have been revised. As a result, the generation of velocity obstacles and the calculation of collision-free rotation angles are modified to reflect the concept of reciprocity between robots. The other steps are the same with the previous method, as shown in Fig. 5.1.

This chapter is organized as follows. Section 5.2 formulates the problem of collision avoidance between elliptic robots. Section 5.3 adapts the ellipse-based velocity obstacle to multi-robot collision avoidance scenarios. Section 5.4 calculates the collision-free reciprocal rotation angles on the basis of the reciprocity of rotation. Finally, this chapter is summarized in Section 5.5.

5.2 Problem Formulation

In this chapter, the problem of real-time local collision avoidance for elliptic robots is considered, which is another subproblem of Problem 1.1.

Consider holonomic elliptic robots $\mathcal{R}_i$, $i = 1, \cdots, N_{\mathcal{R}}$ moving in a planar workspace $\mathcal{W}$. Each robot $\mathcal{R}_i$ has major and minor radii of $M_{\mathcal{R}_i}$ and $m_{\mathcal{R}_i}$, and its configuration consists of the position $\mathbf{p}_{\mathcal{R}_i}$ and orientation $\theta_{\mathcal{R}_i}$. In addition, the shape of robot $\mathcal{R}_i$ is represented by a symmetric positive definite matrix $\mathbf{S}_{\mathcal{R}_i}(\theta_{\mathcal{R}_i}) \in \mathbb{R}^{2\times 2}$. Accordingly, a robot $\mathcal{R}_i$ occupies the region of $\mathcal{F}_{\mathcal{R}_i}(\theta_{\mathcal{R}_i}) = \{\mathbf{x} \in \mathcal{W} \,|\, f(\mathbf{x}; \mathbf{S}_{\mathcal{R}_i}(\theta_{\mathcal{R}_i}), \mathbf{p}_{\mathcal{R}_i}) \le 0\}$ in its workspace. Furthermore, each robot $\mathcal{R}_i$ moves with its linear velocity $\mathbf{v}_{\mathcal{R}_i}$ and rotates with its angular velocity $w_{\mathcal{R}_i}$. Its mobility is limited by the dynamic constraints such as the maximum linear speed $v_{\mathcal{R}_i}^{max}$, angular speed $w_{\mathcal{R}_i}^{max}$, linear acceleration $a_{\mathcal{R}_i}^{max}$, and angular acceleration $\alpha_{\mathcal{R}_i}^{max}$.

Next, a robot detects other robots in its detection range with an omnidirectional range sensor with a detection range up to ρ, and there is no communication between robots. Let $\mathcal{NR}_i$ denote the sets of robots detected by robot $\mathcal{R}_i$. Since each robot is assumed to identify all other robots, robot $\mathcal{R}_i$ knows the shape of robot $\mathcal{R}_j \in \mathcal{NR}_i$. Thus robot $\mathcal{R}_i$ can measure the position $\mathbf{p}_{\mathcal{R}_j}$, orientation $\theta_{\mathcal{R}_j}$, linear velocity $\mathbf{v}_{\mathcal{R}_j}$, and angular velocity $w_{\mathcal{R}_j}$ of robot $\mathcal{R}_j$.

The objective of all the robots is to reach their goal position $\mathbf{p}_{\mathcal{R}_i}^{goal}$ given by an external global planner through the shortest path possible without collisions. Because a discrete-time robot model is employed, the robots select their new linear velocity $\mathbf{v}_{\mathcal{R}_i}^{new}$ and angular velocity $w_{\mathcal{R}_i}^{new}$ based on the current positions and velocities of itself and the observed other robots at each time step. Therefore, the problem to be solved in this chapter is defined as follows.

Problem 5.1 (Collision Avoidance for Elliptic Robots) *For any elliptic robot $\mathcal{R}_i$, compute the new linear velocity $\mathbf{v}_{\mathcal{R}_i}^{new}$ and angular velocity $w_{\mathcal{R}_i}^{new}$ in order to generate a trajectory to the goal that circumvents other robots with the minimum possible deviation from*

- *the measurements of other robots in its detection area D_i,*
- *the robot's goal position $\mathbf{p}_{\mathcal{R}_i}^{goal}$.*

As explained in Section 4.2, this problem is decomposed into two subproblems: Problem 5.2 and Problem 5.3.

Problem 5.2 (Collision Avoidance for Elliptic Robots Capable of Only Translational Motion) *Identical to Problem 5.1 except that $w_{\mathcal{R}_i}^{new} = 0$ is assumed.*

Problem 5.3 (Angular Motion Control for Elliptic Robots Circumventing Other Robots) *Compute the new angular velocity $w_{\mathcal{R}_i}^{new}$ of an elliptic robot $\mathcal{R}_i$ moving at $\mathbf{v}_{\mathcal{R}_i}$ in order to potentially circumvent other robots interrupting its path to the goal with the minimum deviation from the shortest path.*

The linear motions of robots are determined in Problem 5.2, and the angular ones are decided in Problem 5.3. At each time step, Problem 5.2 and Problem 5.3 are sequentially solved. Hence, the solution of Problem 5.2 is assigned to $\mathbf{v}_{\mathcal{R}_i}$ in Problem 5.3.

5.3 Ellipse-based Reciprocal Velocity Obstacles

In this section, the concept of reciprocity is applied to the ellipse-based velocity obstacle. Because the EBVO is only associated with Problem 5.2, the angular motions are not considered.

First, the reason why the oscillation occurs is explained from [102] provided that a robot expects other robots to keep their current velocities for a moment. Suppose there are two elliptic robots $\mathcal{R}_1$ and $\mathcal{R}_2$ at $\mathbf{p}_{\mathcal{R}_1}(t_0)$ and $\mathbf{p}_{\mathcal{R}_2}(t_0)$ at time t_0, respectively. They are also currently moving with their preferred linear velocities, that is $\mathbf{v}_{\mathcal{R}_1}(t_0) = \mathbf{v}_{\mathcal{R}_1}^{pref}$ and $\mathbf{v}_{\mathcal{R}_2}(t_0) = \mathbf{v}_{\mathcal{R}_2}^{pref}$, which leads to a collision. In other words, $\mathbf{v}_{\mathcal{R}_1}(t_0) \in VO_{\mathcal{R}_1|\mathcal{R}_2}^{\infty}(t_0)$ and $\mathbf{v}_{\mathcal{R}_2}(t_0) \in VO_{\mathcal{R}_2|\mathcal{R}_1}^{\infty}(t_0)$. Hence, the robots $\mathcal{R}_1$ and $\mathcal{R}_2$ select their new linear velocities $\mathbf{v}_{\mathcal{R}_1}^{new}$ and $\mathbf{v}_{\mathcal{R}_2}^{new}$ such that

$$\mathbf{v}_{\mathcal{R}_1}^{new} \notin VO_{\mathcal{R}_1|\mathcal{R}_2}^{\infty}(t_0)\,, \tag{5.1}$$

$$\mathbf{v}_{\mathcal{R}_2}^{new} \notin VO_{\mathcal{R}_2|\mathcal{R}_1}^{\infty}(t_0) \tag{5.2}$$

for the time interval $(t_0, t_0 + \Delta t]$.

After a very short duration Δt, the preferred ones remain unchanged because $\mathbf{p}_{\mathcal{R}_1}(t_0) \approx \mathbf{p}_{\mathcal{R}_1}(t_0 + \Delta t)$ and $\mathbf{p}_{\mathcal{R}_2}(t_0) \approx \mathbf{p}_{\mathcal{R}_2}(t_0 + \Delta t)$. Nevertheless, the velocities change to $\mathbf{v}_{\mathcal{R}_1}(t_0 + \Delta t) = \mathbf{v}_{\mathcal{R}_1}^{new}$ and $\mathbf{v}_{\mathcal{R}_2}(t_0 + \Delta t) = \mathbf{v}_{\mathcal{R}_2}^{new}$. Since

$$\mathbf{v}_{\mathcal{R}_1}^{pref} \notin VO_{\mathcal{R}_1|\mathcal{R}_2}^{\infty}(t_0 + \Delta t)\,, \tag{5.3}$$

$$\mathbf{v}_{\mathcal{R}_2}^{pref} \notin VO_{\mathcal{R}_2|\mathcal{R}_1}^{\infty}(t_0 + \Delta t) \tag{5.4}$$

hold from (5.1) and (5.2), the two robot move with $\mathbf{v}_{\mathcal{R}_1}^{pref}$ and $\mathbf{v}_{\mathcal{R}_2}^{pref}$ again, at which they will collide with each other. As a consequence, regarding other robots as obstacles decreases the stability and makes the oscillation in their motion, as shown in Fig. 5.2.

To address this problem, the reciprocal velocity obstacle (RVO) was presented in [118], where a robot selected the middle point between its current linear velocity and a velocity outside the EBVO as its new linear velocity.

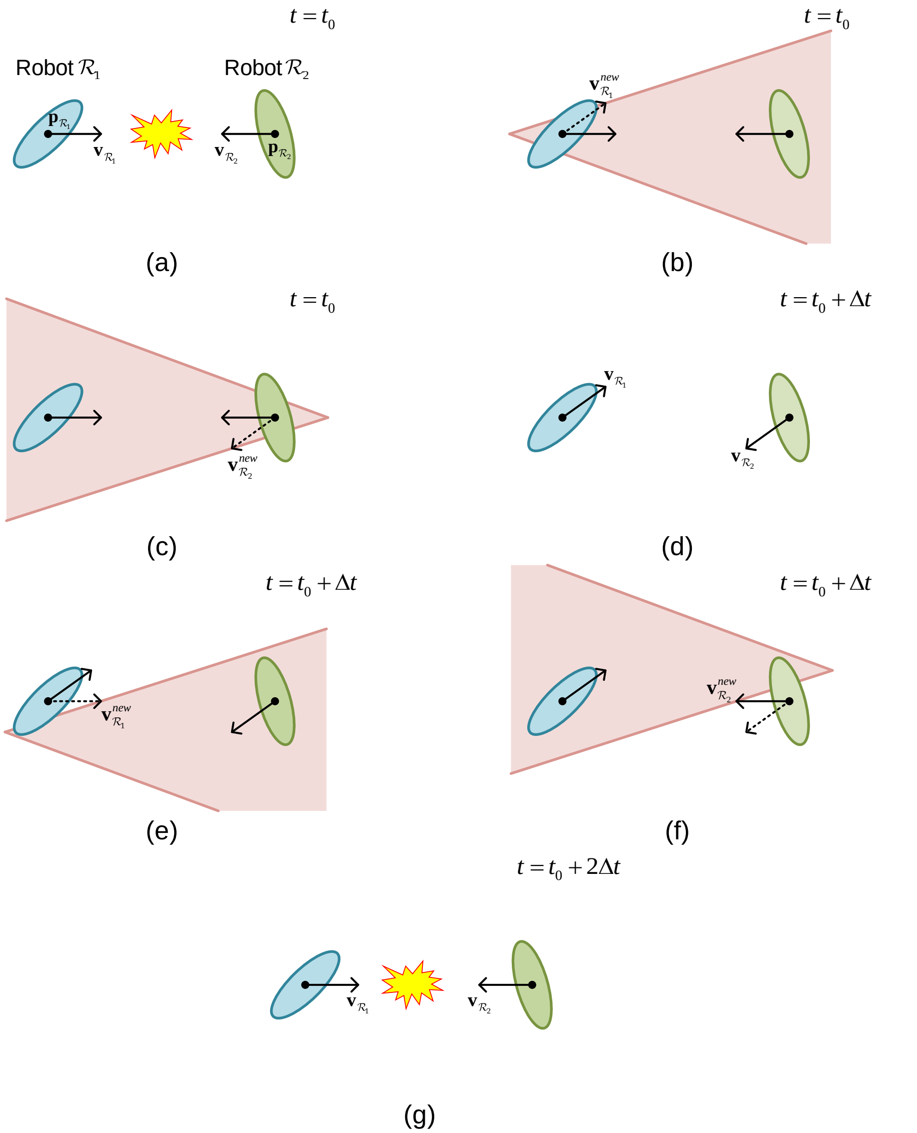

$t = t_0$
Robot $\mathcal{R}_1$
Robot $\mathcal{R}_2$
$\mathbf{p}_{\mathcal{R}_1}$
$\mathbf{v}_{\mathcal{R}_1}$
$\mathbf{v}_{\mathcal{R}_2}$
$\mathbf{p}_{\mathcal{R}_2}$
(a)
$t = t_0$
$\mathbf{v}_{\mathcal{R}_1}^{new}$
(b)
$t = t_0$
$\mathbf{v}_{\mathcal{R}_2}^{new}$
(c)
$t = t_0 + \Delta t$
$\mathbf{v}_{\mathcal{R}_1}$
$\mathbf{v}_{\mathcal{R}_2}$
(d)
$t = t_0 + \Delta t$
$\mathbf{v}_{\mathcal{R}_1}^{new}$
(e)
$t = t_0 + \Delta t$
$\mathbf{v}_{\mathcal{R}_2}^{new}$
(f)
$t = t_0 + 2\Delta t$
$\mathbf{v}_{\mathcal{R}_1}$
$\mathbf{v}_{\mathcal{R}_2}$
(g)

Figure 5.2 The oscillation in the motions of two robots that try to avoid each other based on the obstacle avoidance method on a head-on collision course [102]. (a) The two robots $\mathcal{R}_1$ and $\mathcal{R}_2$ are located at $\mathbf{p}_{\mathcal{R}_1}$ and $\mathbf{p}_{\mathcal{R}_2}$ and moving with velocities of $\mathbf{v}_{\mathcal{R}_1}$ and $\mathbf{v}_{\mathcal{R}_2}$ at t_0. The yellow spark shows that the two robot will collide in the future. (b)–(c) The new linear velocities of $\mathcal{R}_1$ and $\mathcal{R}_2$ are selected at t_0. (d) The positions and the velocities of the two robots at $t_0 + \Delta t$ are presented. (e)–(f) The new linear velocities of $\mathcal{R}_1$ and $\mathcal{R}_2$ are selected at $t_0 + \Delta t$. (g) The two robots are moving at $t_0 + 2\Delta t$ with the same velocities at t_0 that induces a collision.

Definition 5.4 (Reciprocal Velocity Obstacle) *The reciprocal velocity obstacle $RVO^{\tau}_{\mathcal{R}_i|\mathcal{R}_j}$ for a robot $\mathcal{R}_i$ induced by another robot $\mathcal{R}_j$ is defined as*

$$RVO^{\tau}_{\mathcal{R}_i|\mathcal{R}_j} = \left\{ \mathbf{v} \in \mathbb{R}^2 \,\middle|\, 2\mathbf{v} - \mathbf{v}_{\mathcal{R}_i} \in VO^{\tau}_{\mathcal{R}_i|\mathcal{R}_j} \right\}. \tag{5.5}$$

Hence, the RVO for the infinite time horizon is represented by a cone with apex at $\left(\mathbf{v}_{\mathcal{R}_i} + \mathbf{v}_{\mathcal{R}_j}\right)/2$ as follows:

Corollary 5.5 *The reciprocal velocity obstacle $RVO^{\tau}_{\mathcal{R}_i|\mathcal{R}_j}$ for a robot $\mathcal{R}_i$ induced by another robot $\mathcal{R}_j$ is represented by*

$$RVO^{\tau}_{\mathcal{R}_i|\mathcal{R}_j} = \text{cone}^{\tau}\left(QO_{\mathcal{R}_i|\mathcal{R}_j}\right) \oplus \left\{ \frac{\mathbf{v}_{\mathcal{R}_i} + \mathbf{v}_{\mathcal{R}_j}}{2} \right\}, \tag{5.6}$$

where $\text{cone}^{\tau}\left(\mathcal{C}\right)$ *is from* (4.37).

However, the two robots are guaranteed not to collide with each other if the new velocities are selected on the same side of the RVOs. If their velocities are on the different sides, a collision may occur because they do not cooperate. To remedy this problem, the hybrid reciprocal velocity obstacle (HRVO) was defined in [104]. They defined the preferred side of $RVO_{\mathcal{R}_i|\mathcal{R}_j}$ as the closer side to the current velocity $\mathbf{v}_{\mathcal{R}_i}$. If both robots select the new velocities on the preferred sides, it means that they avoid each other cooperatively. If a robot selects the new velocity on the other side, it means that the robot do not cooperate to avoid collisions, which leads to substitute the boundary of the original VO. In short, the HRVO is bounded by one line from the RVO and the other one from the VO, as shown in Fig. 5.3. Therefore, the new apex of the HRVO is obtained as follows.

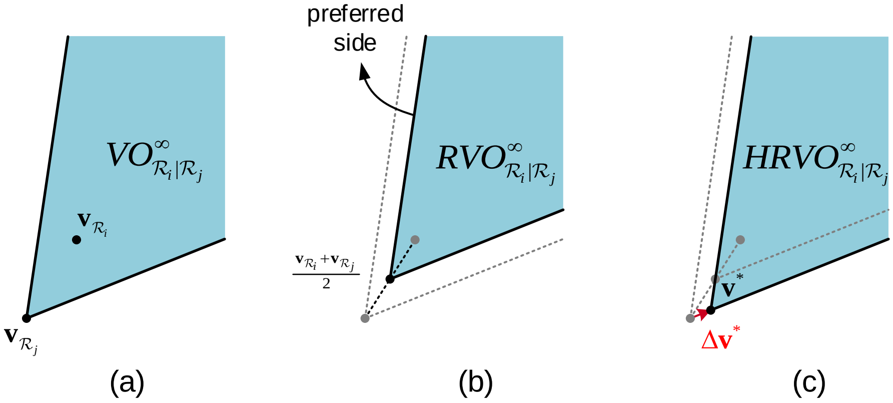

preferred side
$VO^{\infty}_{\mathcal{R}_i|\mathcal{R}_j}$
$\mathbf{v}_{\mathcal{R}_i}$
$\mathbf{v}_{\mathcal{R}_j}$
$RVO^{\infty}_{\mathcal{R}_i|\mathcal{R}_j}$
$\frac{\mathbf{v}_{\mathcal{R}_i}+\mathbf{v}_{\mathcal{R}_j}}{2}$
$HRVO^{\infty}_{\mathcal{R}_i|\mathcal{R}_j}$
$\mathbf{v}^*$
$\Delta\mathbf{v}^*$
(a)
(b)
(c)

Figure 5.3 Three kinds of velocities obstacles for a robot $\mathcal{R}_i$ induced by another robot $\mathcal{R}_j$: (a) the original velocity obstacle $VO^{\infty}_{\mathcal{R}_i|\mathcal{R}_j}$; (b) the reciprocal velocity obstacle $RVO^{\infty}_{\mathcal{R}_i|\mathcal{R}_j}$; (c) the hybrid reciprocal velocity obstacle $HRVO^{\infty}_{\mathcal{R}_i|\mathcal{R}_j}$.

First of all, suppose that the left side of $RVO^{\tau}_{\mathcal{R}_i|\mathcal{R}_j}$ is closer to the current velocity $\mathbf{v}_{\mathcal{R}_i}$ such that

$$\left(\mathbf{v}_{\mathcal{R}_i} - \mathbf{v}_{\mathcal{R}_j}\right)^T \mathbf{S}^{-\frac{1}{2}}_{\mathcal{R}_j} \left(\mathbf{u}^{\text{left}}_{\mathcal{R}_i|\mathcal{R}_j} - \mathbf{u}^{\text{right}}_{\mathcal{R}_i|\mathcal{R}_j}\right) \leq 0. \tag{5.7}$$

Then the apex of the cone, $\mathbf{v}^*$ is the intersection of the left leg of $RVO^{\tau}_{\mathcal{R}_i|\mathcal{R}_j}$ and the right leg of $VO^{\tau}_{\mathcal{R}_i|\mathcal{R}_j}$:

$$\begin{cases} \left(\mathbf{v}^* - \frac{\mathbf{v}_{\mathcal{R}_i}+\mathbf{v}_{\mathcal{R}_j}}{2}\right)^T \mathbf{S}^{-\frac{1}{2}}_{\mathcal{R}_j} \mathbf{u}^{\text{left}}_{\mathcal{R}_i|\mathcal{R}_j} & = 0, \\ \left(\mathbf{v}^* - \mathbf{v}_{\mathcal{R}_j}\right)^T \mathbf{S}^{-\frac{1}{2}}_{\mathcal{R}_j} \mathbf{u}^{\text{right}}_{\mathcal{R}_i|\mathcal{R}_j} & = 0. \end{cases} \tag{5.8}$$

Let $\Delta\mathbf{v}^* = \mathbf{v}^* - \mathbf{v}_{\mathcal{R}_j}$. Then the equation can be written as

$$\Delta\mathbf{v}^* = \frac{1}{2}\mathbf{S}^{\frac{1}{2}}_{\mathcal{R}_j} \left(\begin{bmatrix}\mathbf{u}^{\text{left}}_{\mathcal{R}_i|\mathcal{R}_j} & \mathbf{u}^{\text{right}}_{\mathcal{R}_i|\mathcal{R}_j}\end{bmatrix}^T\right)^{-1} \begin{bmatrix} \left(\mathbf{v}_{\mathcal{R}_i} - \mathbf{v}_{\mathcal{R}_j}\right)^T \mathbf{S}^{-\frac{1}{2}}_{\mathcal{R}_j} \mathbf{u}^{\text{left}}_{\mathcal{R}_i|\mathcal{R}_j} \\ 0 \end{bmatrix}. \tag{5.9}$$

If the right side of $RVO^{\tau}_{\mathcal{R}_i|\mathcal{R}_j}$ is closer to the current velocity $\mathbf{v}_{\mathcal{R}_i}$,

$$\begin{cases} \left(\mathbf{v}^* - \mathbf{v}_{\mathcal{R}_j}\right)^T \mathbf{S}_{\mathcal{R}_j}^{-\frac{1}{2}} \mathbf{u}^{\text{left}}_{\mathcal{R}_i|\mathcal{R}_j} & = 0, \\ \left(\mathbf{v}^* - \frac{\mathbf{v}_{\mathcal{R}_i}+\mathbf{v}_{\mathcal{R}_j}}{2}\right)^T \mathbf{S}_{\mathcal{R}_j}^{-\frac{1}{2}} \mathbf{u}^{\text{right}}_{\mathcal{R}_i|\mathcal{R}_j} & = 0. \end{cases} \tag{5.10}$$

Hence,

$$\Delta\mathbf{v}^* = \frac{1}{2}\mathbf{S}_{\mathcal{R}_j}^{\frac{1}{2}} \left(\begin{bmatrix} \mathbf{u}^{\text{left}}_{\mathcal{R}_i|\mathcal{R}_j} & \mathbf{u}^{\text{right}}_{\mathcal{R}_i|\mathcal{R}_j} \end{bmatrix}^T \right)^{-1} \begin{bmatrix} 0 \\ \left(\mathbf{v}_{\mathcal{R}_i} - \mathbf{v}_{\mathcal{R}_j}\right)^T \mathbf{S}_{\mathcal{R}_j}^{-\frac{1}{2}} \mathbf{u}^{\text{right}}_{\mathcal{R}_i|\mathcal{R}_j} \end{bmatrix}. \tag{5.11}$$

In conclusion, the following theorem is derived.

Theorem 5.6 *Consider two elliptic robots $\mathcal{R}_i$ and $\mathcal{R}_j$ in the planar workspace $\mathcal{W}$. If one robot $\mathcal{R}_i$ selects its new linear velocity $\mathbf{v}^{new}_{\mathcal{R}_i}$ outside of the region*

$$HRVO^{\tau}_{\mathcal{R}_i|\mathcal{R}_j} = \text{cone}^{\tau}\left(QO_{\mathcal{R}_i|\mathcal{R}_j}\right) \oplus \left\{\mathbf{v}_{\mathcal{R}_j} + \Delta\mathbf{v}^*\right\}, \tag{5.12}$$

then $\mathcal{R}_i$ does not collide with $\mathcal{R}_j$ within a time horizon τ. Here, $\Delta\mathbf{v}^*$ is computed from (5.9) *if* (5.7) *holds and is computed from* (5.11) *otherwise.*

5.4 Collision-free Reciprocal Rotation Angles

With respect to a local frame attached to $\mathcal{R}_i$, the robot is rotating at the origin at time t_0. Also, another robot $\mathcal{R}_j$ is located at $\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} = \mathbf{p}_{\mathcal{R}_j} - \mathbf{p}_{\mathcal{R}_i}$ and is moving with

$$\hat{\mathbf{v}}^{new}_{\mathcal{R}_i|\mathcal{R}_j} = \mathbf{v}_{\mathcal{R}_i} + \mathbf{v}_{\mathcal{R}_j} - 2\mathbf{v}^{new}_{\mathcal{R}_i}, \tag{5.13}$$

which is the expected relative velocity of $\mathcal{R}_j$ with respect to $\mathcal{R}_i$ during the next sampling period from (5.5).

In this section, it is assumed that the two robots rotate reciprocally. If one robot rotates by $\Delta\theta$, then the other robot may rotate by $\Delta\theta$ equally or $-\Delta\theta$ equally opposite as referred to in [45]. If a robot $\mathcal{R}$ rotates by an angle of $\Delta\theta$, the shape matrix $\mathbf{S}(\theta)$ is transformed into

$$\mathbf{S}(\theta + \Delta\theta) = \mathbf{R}_{\Delta\theta}\mathbf{S}(\theta)\,\mathbf{R}_{\Delta\theta}^{T}, \tag{5.14}$$

where $\mathbf{R}_\theta$ is the rotation matrix associated with a counter-clockwise rotation of angle θ. The swept area the robot rotates through from $\Delta\theta^-$ to $\Delta\theta^+$ and moves in T_F is defined as

$$\mathcal{SA}\,(\Theta) = \bigcup_{0 \le t \le T_F} \left(\{t\mathbf{v}\} \oplus \bigcup_{\Delta\theta \in \Theta} \mathcal{F}(\theta + \Delta\theta) \right), \tag{5.15}$$

where $\Theta = [\Delta\theta^-, \Delta\theta^+]$. Because the robot $\mathcal{R}_i$ is fixed at the origin within T_F,

$$\mathcal{SA}_i\,(\Theta_i) = \bigcup_{\Delta\theta_i \in \Theta_i} \mathcal{F}_i(\theta_i + \Delta\theta_i). \tag{5.16}$$

On the other hand, the robot $\mathcal{R}_j$ is expected to moves with $\hat{\mathbf{v}}^{new}_{\mathcal{R}_i|\mathcal{R}_j}$, so that

$$\mathcal{SA}_j\,(\Theta_j) = \bigcup_{0 \le t \le T_F} \left(\left\{t\hat{\mathbf{v}}^{new}_{\mathcal{R}_i|\mathcal{R}_j}\right\} \oplus \bigcup_{\Delta\theta_j \in \Theta_j} \mathcal{F}_j(\theta_j + \Delta\theta_j) \right), \tag{5.17}$$

Figure 5.4 shows these areas. The region $\mathcal{SA}_j\,(\Theta_i)$ is obtained by rotating the ellipse by a given set of rotation angles Θ_1. However, $\mathcal{SA}_j\,(\Theta_j)$ is obtained by translating a swept area an ellipse rotates through by Θ_2. The boundary of the region $\mathcal{SA}_j\,(\Theta_j)$ is decomposed into six parts. Two of them, $\mathcal{SA}_{j,1}$ and $\mathcal{SA}_{j,2}$, are induced by the rotating ellipses located at $\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}$ and $\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} + T_F\hat{\mathbf{v}}^{new}_{\mathcal{R}_i|\mathcal{R}_j}$,

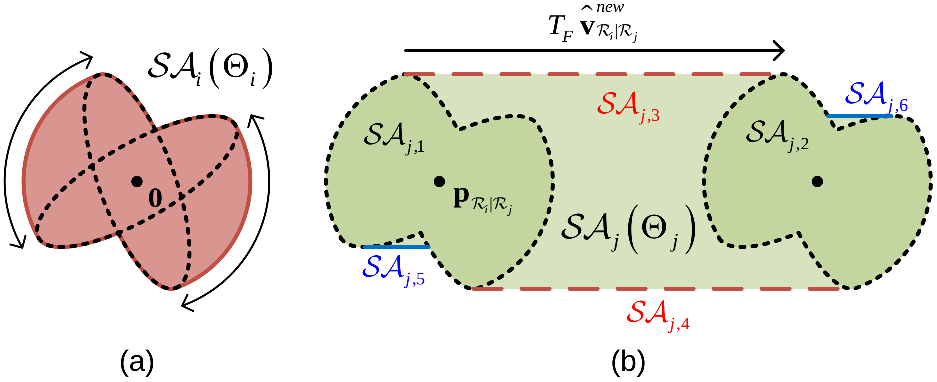

$\mathcal{SA}_i(\Theta_i)$
$\mathbf{0}$
$T_F\,\hat{\mathbf{v}}^{new}_{\mathcal{R}_i|\mathcal{R}_j}$
$\mathcal{SA}_{j,3}$
$\mathcal{SA}_{j,6}$
$\mathcal{SA}_{j,1}$
$\mathcal{SA}_{j,2}$
$\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}$
$\mathcal{SA}_j(\Theta_j)$
$\mathcal{SA}_{j,5}$
$\mathcal{SA}_{j,4}$
(a)
(b)

Figure 5.4 The swept areas $\mathcal{SA}_i\,(\Theta_i)$ and $\mathcal{SA}_j\,(\Theta_j)$: (a) the swept area robot $\mathcal{R}_i$ rotates through; (b) the swept area robot $\mathcal{R}_j$ rotates through and moves in the time-to-contact T_F. The boundary $\partial\mathcal{SA}_j\,(\Theta_j)$ is composed of the parts of the two swept areas $\mathcal{SA}_{j,1}$ and $\mathcal{SA}_{j,2}$ of the ellipses whose centers located at $\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}$ and $\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} + T_f\hat{\mathbf{v}}^{new}_{\mathcal{R}_i|\mathcal{R}_j}$ and of four lines $\mathcal{SA}_{j,3}$, $\mathcal{SA}_{j,4}$, $\mathcal{SA}_{j,5}$, and $\mathcal{SA}_{j,6}$.

and the others, $\mathcal{SA}_{j,3}$, $\mathcal{SA}_{j,4}$, $\mathcal{SA}_{j,5}$, and $\mathcal{SA}_{j,6}$, are line-segments. The line segments $\mathcal{SA}_{j,5}$ and $\mathcal{SA}_{j,6}$ may not be revealed depending on Θ_j.

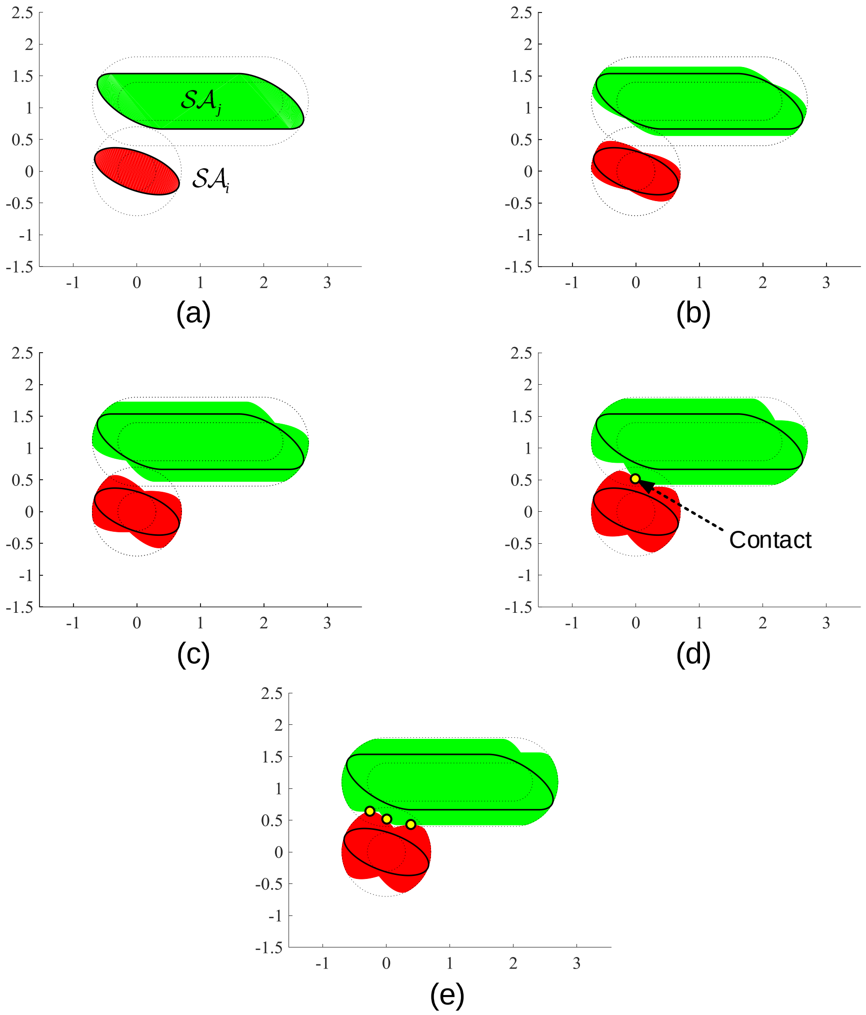
The objective of this section is to find the maximal intervals Θ_i and Θ_j such that the two swept areas $\mathcal{SA}_i\,(\Theta_i)$ and $\mathcal{SA}_j\,(\Theta_j)$ do not collide with each other. Technically speaking, let $\mathcal{L}_4$ denote a lexicographically ordered set of 4-tuples of positive real numbers between 0 and π, and $\mathcal{E}_4 : I^2 \to \mathcal{L}_4$ be a function that maps a pair of the intervals to a 4-tuple with the elements of ascending sorted absolute values of the endpoints. For instance, suppose $\Theta_1 = [-0.2, 0.3]$ and $\Theta_2 = [-0.4, 0.1]$. Then $\mathcal{E}_4(\Theta_1, \Theta_2) = \{0.1, 0.2, 0.3, 0.4\}$. In conclusion, the problem in this section is defined as follows:

$$\begin{aligned} \underset{\Theta_i,\Theta_j}{\operatorname{argmax}} \quad & \mathcal{E}_4(\Theta_i, \Theta_j)\,, \\ \text{subject to} \quad & \mathcal{SA}_i\,(\Theta_i) \cap \mathcal{SA}_j\,(\Theta_j) = \emptyset. \end{aligned} \tag{5.18}$$

An example of the intuitive approach to the above problem is described in Fig. 5.5. From (a) to (d), the swept areas gradually expand until contacting each other at $\Delta\theta = -42.90°$. Because they are still available to rotate in counterclockwise directions, they sweep in those directions until they contact again. As a result, the collision-free reciprocal rotation angles of the two robots are determined as shown in Fig. 5.5(e).

However, it is inefficient to find out the intervals by discretizing the rotation angles and checking whether a collision occurs for each of the angles or not. Hence, the proposed algorithm finds all the pairs of the rotation angles of the two robots at which the swept areas may touch each other. Let $\mathcal{C} \in \mathbb{R}^2$ denote the set of these candidates. Because there are four combinations of the rotation directions of two robots, the set $\mathcal{C}$ consists of four subsets: $\mathcal{C}_{--}$, $\mathcal{C}_{-+}$, $\mathcal{C}_{+-}$, and $\mathcal{C}_{++}$. The subscript of the subsets represents the rotation directions of the robots $\mathcal{R}_i$ and $\mathcal{R}_j$. For instance, the set $\mathcal{C}_{-+}$ includes the candidates of the pairs when $\mathcal{R}_i$ rotates in a clockwise direction and $\mathcal{R}_j$ rotates in a counterclockwise direction.

The derivation of these candidate sets is decomposed into two steps. To begin with, the first contact angle is found from the sets generated on the basis of reciprocity of rotation. There are two cases that the two swept areas contact with each other. The one is that the two robots are collided when they rotate by angles with the same absolute value. Accordingly, both of them can no longer continue to rotate in the directions they rotated. On the other hand, the other is that a robot touches a point that the other robot already rotate through. In this case, one robot can rotate in the direction it rotated, whereas the other cannot. The calculation of these sets is explained in detail in Section 5.4.1.



(a)

(b)

(c)

(d)

(e)

Figure 5.5 An intuitive approach to determining the collision-free reciprocal rotation angles of the two robots $\mathcal{R}_i$ and $\mathcal{R}_j$: (a) the swept areas of the two robots of $\Delta\theta = 0°$; (b) the swept areas of the two robots of $\Delta\theta = \pm 15°$; (c) those of $\Delta\theta = \pm 30°$; (d) the swept areas contact when $\Delta\theta = \pm 42.90°$; (e) the swept areas expand to fill the rest, so that $\Theta_i = [-42.90°, 48.44°]$ and $\Theta_j = [-42.90°, 63.75°]$. The yellow circles indicate the contact points between the two swept areas.

After the first contact rotation angle is determined, some of these subsets should be updated because the decision of the contact point affects them. For example, suppose that a collision between the two robots occurs when $\mathcal{R}_i$ and $\mathcal{R}_j$ rotate by $-\pi < \Delta\theta_i^- \leq 0$ and $0 \leq \Delta\theta_j^+ < \pi$, respectively, and the swept areas cannot expand anymore in those directions. This result has to be considered in $\mathcal{C}_{--}$ and $\mathcal{C}_{++}$ because $\mathcal{C}_{--}$ is concerned with the clockwise rotation of the robot $\mathcal{R}_i$ and $\mathcal{C}_{++}$ is associated with the counterclockwise rotation of the robot $\mathcal{R}_j$. This is explained in detail in Section 5.4.2.

The proposed algorithm solves the problem by deciding the endpoints of the rotation about one combination of the directions at a time. Hence, it obtains the answer by repeating the proccess of the decision and the update four times after it initializes the four subsets, which is elucidated in Section 5.4.3.

5.4.1 Candidates of the First Contact Rotation Angle

As shown in Fig. 5.4, the boundary of the region $\mathcal{SA}_j\left(\Theta_j\right)$ is classified into two parts: the segments of swept areas of ellipses and those of the moving lines. In order to obtain the set $\mathcal{C}$, the candidates of the rotation angles at which the swept regions of two ellipses touch each other are first investigated. Next, those at which the swept region and the moving line segments are in contact are calculated by calculating the distance between the center to the tangent line of an ellipse.

The swept area of an ellipse

Suppose that there are two elliptic robots $\mathcal{R}_i$ and $\mathcal{R}_j$ rotating at the origin and $\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}$, respectively, which are initially separated each other. If $\left\|\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}\right\| \geq$

$M_{\mathcal{R}_i} + M_{\mathcal{R}_j}$, where $M_{\mathcal{R}_i}$ and $M_{\mathcal{R}_j}$ is the major radius of the robots, then $\Theta_i = [-\pi, \pi]$ and $\Theta_j = [-\pi, \pi]$. Otherwise, there are three cases that the swept areas touch each other:

- the two robots contact when they rotate in the same direction;
- the two robots contact when they rotate in the opposite direction;
- the robot $\mathcal{R}_i$ contacts with the disk with radius of $M_{\mathcal{R}_j}$ or vice versa.

The sets $\mathcal{C}_{--}$ and $\mathcal{C}_{++}$ are generated in the first and third cases, and the sets $\mathcal{C}_{-+}$ and $\mathcal{C}_{+-}$ are produced in the second and third cases. This process is also repeated for the ellipse located at $\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} + \hat{\mathbf{v}}^{new}_{\mathcal{R}_i|\mathcal{R}_j}$.

First of all, the coefficient matrices of the two robots rotating in the same direction are represented by

$$\mathbf{M}_{\mathcal{R}_i}(\Delta\theta) = \begin{bmatrix} \mathbf{R}_{\Delta\theta}\mathbf{S}^{-1}_{\mathcal{R}_i}\mathbf{R}^T_{\Delta\theta} & \mathbf{0} \\ \mathbf{0} & -1 \end{bmatrix}, \tag{5.19}$$

$$\mathbf{M}_{\mathcal{R}_j}(\Delta\theta) = \begin{bmatrix} \mathbf{R}_{\Delta\theta}\mathbf{S}^{-1}_{\mathcal{R}_j}\mathbf{R}^T_{\Delta\theta} & -\mathbf{R}_{\Delta\theta}\mathbf{S}^{-1}_{\mathcal{R}_j}\mathbf{R}^T_{\Delta\theta}\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} \\ -\mathbf{p}^T_{\mathcal{R}_i|\mathcal{R}_j}\mathbf{R}_{\Delta\theta}\mathbf{S}^{-1}_{\mathcal{R}_j}\mathbf{R}^T_{\Delta\theta} & \mathbf{p}^T_{\mathcal{R}_i|\mathcal{R}_j}\mathbf{R}_{\Delta\theta}\mathbf{S}^{-1}_{\mathcal{R}_j}\mathbf{R}^T_{\Delta\theta}\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} - 1 \end{bmatrix}. \tag{5.20}$$

Put

$$\cos 2\Delta\theta = \frac{1-t^2}{1+t^2}, \tag{5.21}$$

$$\sin 2\Delta\theta = \frac{2t}{1+t^2}. \tag{5.22}$$

Then the characteristic polynomial of the two robots is

$$g(\xi) = a_3\xi^3 + a_2\xi^2 + a_1\xi + a_0, \tag{5.23}$$

where

$$
\begin{aligned}
a_3 &= -\det \mathbf{S}_{\mathcal{R}_i}^{-1}, && (5.24)\\
a_2 &= \frac{a_{22}t^2 + a_{21}t + a_{20}}{t^2+1}, && (5.25)\\
a_1 &= \frac{a_{12}t^2 + a_{11}t + a_{10}}{t^2+1}, && (5.26)\\
a_0 &= -\det \mathbf{S}_{\mathcal{R}_j}^{-1}. && (5.27)
\end{aligned}
$$

Here,

$$
\begin{aligned}
a_{22} &= \det \mathbf{S}_{\mathcal{R}_i}^{-1}\left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{R}_{\frac{\pi}{2}}^T \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} - 1\right) - \operatorname{tr}\left(\mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{R}_{\frac{\pi}{2}}^T\right), && (5.28)\\
a_{21} &= 2\det \mathbf{S}_{\mathcal{R}_i}^{-1}\left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}\right), && (5.29)\\
a_{20} &= \det \mathbf{S}_{\mathcal{R}_i}^{-1}\left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} - 1\right) - \operatorname{tr}\left(\mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{R}_{\frac{\pi}{2}}^T\right), && (5.30)\\
a_{12} &= \det \mathbf{S}_{\mathcal{R}_j}^{-1}\left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}}^T \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} - 1\right) - \operatorname{tr}\left(\mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}}^T \mathbf{S}_{\mathcal{R}_j}^{-1}\right), && (5.31)\\
a_{11} &= 2\det \mathbf{S}_{\mathcal{R}_j}^{-1}\left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}\right), && (5.32)\\
a_{10} &= \det \mathbf{S}_{\mathcal{R}_i}^{-1}\left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} - 1\right) - \operatorname{tr}\left(\mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{R}_{\frac{\pi}{2}}^T\right). && (5.33)
\end{aligned}
$$

From Theorem 3.18, the two robots do not overlap if and only if

$$
\begin{cases}
a_2 \geq 0 \quad \text{or} \quad 3a_3a_2a_0 - 4a_3a_1^2 + a_2^2a_1 \geq 0,\\
3a_3a_1 - a_2^2 < 0,\\
27a_3^2a_0^2 - 18a_3a_2a_1a_0 + 4a_3a_1^3 + 4a_2^3a_0 - a_2^2a_1^2 \leq 0.
\end{cases} \tag{5.34}
$$

The above four inequalities can be solved by the same method mentioned in Section 4.6.2. As a result, we obtains two solutions $\Delta\theta^- \leq 0$ and $\Delta\theta^+ \geq 0$, which may not exist in some cases. If there exists $\Delta\theta^- \leq 0$, this result is inserted in the subset $\mathcal{C}_{--}$ as follows:

$$\mathcal{C}_{--} \leftarrow \mathcal{C}_{--} \cup \left\{ \begin{bmatrix} \Delta\theta^- & \Delta\theta^- \end{bmatrix}^T \right\} \tag{5.35}$$

Likewise, if there exists $\Delta\theta^+ \geq 0$,

$$\mathcal{C}_{++} \leftarrow \mathcal{C}_{++} \cup \left\{ \begin{bmatrix} \Delta\theta^+ & \Delta\theta^+ \end{bmatrix}^T \right\}. \tag{5.36}$$

Second, the case that the two robots rotating in the opposite direction is considered. The coefficient matrices are represented by

$$\mathbf{M}_{\mathcal{R}_i}(\Delta\theta) = \begin{bmatrix} \mathbf{R}_{\Delta\theta}\mathbf{S}_{\mathcal{R}_i}^{-1}\mathbf{R}_{\Delta\theta}^T & \mathbf{0} \\ \mathbf{0} & -1 \end{bmatrix}, \tag{5.37}$$

$$\mathbf{M}_{\mathcal{R}_j}(\Delta\theta) = \begin{bmatrix} \mathbf{R}_{\Delta\theta}^T\mathbf{S}_{\mathcal{R}_j}^{-1}\mathbf{R}_{\Delta\theta} & -\mathbf{R}_{\Delta\theta}^T\mathbf{S}_{\mathcal{R}_j}^{-1}\mathbf{R}_{\Delta\theta}\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} \\ -\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T\mathbf{R}_{\Delta\theta}^T\mathbf{S}_{\mathcal{R}_j}^{-1}\mathbf{R}_{\Delta\theta} & \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T\mathbf{R}_{\Delta\theta}^T\mathbf{S}_{\mathcal{R}_j}^{-1}\mathbf{R}_{\Delta\theta}\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} - 1 \end{bmatrix}. \tag{5.38}$$

From (5.21) and (5.22), the coefficients of the characteristic polynomial are written as

$$a_3 = -\det\mathbf{S}_{\mathcal{R}_i}^{-1}, \tag{5.39}$$

$$a_2 = \frac{a_{24}t^4 + a_{23}t^3 + a_{22}t^2 + a_{21}t + a_{20}}{t^4 + 2t^2 + 1}, \tag{5.40}$$

$$a_1 = \frac{a_{14}t^4 + a_{13}t^3 + a_{12}t^2 + a_{11}t + a_{10}}{t^4 + 2t^2 + 1}, \tag{5.41}$$

$$a_0 = -\det\mathbf{S}_{\mathcal{R}_j}^{-1}, \tag{5.42}$$

where

$$a_{24} = \det \mathbf{S}_{\mathcal{R}_i}^{-1} \left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{R}_{\frac{\pi}{2}}^T \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} - 1 \right) - \mathrm{tr} \left(\mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{R}_{\frac{\pi}{2}}^T \right), \tag{5.43}$$

$$a_{23} = -2 \det \mathbf{S}_{\mathcal{R}_i}^{-1} \left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} \right) + 4\mathrm{tr} \left(\mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \right), \tag{5.44}$$

$$a_{22} = \det \mathbf{S}_{\mathcal{R}_i}^{-1} \left(\left\| \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} \right\|^2 \mathrm{tr} \mathbf{S}_{\mathcal{R}_j}^{-1} - 2 \right) - 4\mathrm{tr} \left(\mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{S}_{\mathcal{R}_j}^{-1} \right) + 2\mathrm{tr} \left(\mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{R}_{\frac{\pi}{2}}^T \right), \tag{5.45}$$

$$a_{21} = -2 \det \mathbf{S}_{\mathcal{R}_i}^{-1} \left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} \right) - 4\mathrm{tr} \left(\mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \right), \tag{5.46}$$

$$a_{20} = \det \mathbf{S}_{\mathcal{R}_i}^{-1} \left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} - 1 \right) - \mathrm{tr} \left(\mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{R}_{\frac{\pi}{2}}^T \right), \tag{5.47}$$

$$a_{14} = \det \mathbf{S}_{\mathcal{R}_j}^{-1} \left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}}^T \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} - 1 \right) - \mathrm{tr} \left(\mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}}^T \mathbf{S}_{\mathcal{R}_j}^{-1} \right), \tag{5.48}$$

$$a_{13} = 2 \det \mathbf{S}_{\mathcal{R}_j}^{-1} \left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} \right) + 4\mathrm{tr} \left(\mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \right), \tag{5.49}$$

$$a_{12} = \det \mathbf{S}_{\mathcal{R}_j}^{-1} \left(\left\| \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} \right\|^2 \mathrm{tr} \mathbf{S}_{\mathcal{R}_i}^{-1} - 2 \right) - 4\mathrm{tr} \left(\mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{S}_{\mathcal{R}_j}^{-1} \right) + 2\mathrm{tr} \left(\mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{R}_{\frac{\pi}{2}}^T \right), \tag{5.50}$$

$$a_{11} = 2 \det \mathbf{S}_{\mathcal{R}_j}^{-1} \left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} \right) - 4\mathrm{tr} \left(\mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \right), \tag{5.51}$$

$$a_{10} = \det \mathbf{S}_{\mathcal{R}_i}^{-1} \left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} - 1 \right) - \mathrm{tr} \left(\mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{R}_{\frac{\pi}{2}}^T \right). \tag{5.52}$$

Likewise, the interior-disjoint condition of the two robots from (5.34) can be solved by the same method in Section 4.6.2 by putting the above parameters. As a result, two solutions $\Delta\theta^- \leq 0$ and $\Delta\theta^+ \geq 0$ can be obtained, which may not exist. If there exists $\Delta\theta^- \leq 0$,

$$\mathcal{C}_{-+} \leftarrow \mathcal{C}_{-+} \cup \left\{ \begin{bmatrix} \Delta\theta^- & -\Delta\theta^- \end{bmatrix}^T \right\} \tag{5.53}$$

Likewise, if there exists $\Delta\theta^{+} \geq 0$,

$$\mathcal{C}_{+-} \leftarrow \mathcal{C}_{+-} \cup \left\{ \begin{bmatrix} \Delta\theta^{+} & -\Delta\theta^{+} \end{bmatrix}^{T} \right\}. \tag{5.54}$$

The two preceding derivation is related with the case that two ellipses rotating with the same speed touch each other. Finally, the case that an ellipse touches a point that the other already rotate through. Accordingly, the rotation angles at which an ellipse touch the disk whose radius is the major radius of the other are calculated.

First, the rotation angles are derived at which the disk of $\mathcal{R}_i$, which is $D\left(\mathbf{0}, M_{\mathcal{R}_i}\right)$, and the swept region of $\mathcal{R}_j$ on the assumption that $D\left(\mathbf{0}, M_{\mathcal{R}_i}\right) \cap \mathcal{F}_{\mathcal{R}_j} = \emptyset$. The contact rotation angles of robot $\mathcal{R}_j$ are easily calculated by the method in Section 4.6.2 because this problem is to find the collision-free angles for a fixed circle. Let $\Delta\theta_j^{-} \leq 0$ and $\Delta\theta_j^{+} \geq 0$ be the solutions. The contact point $\mathbf{x}_c^{-}$ when the robot $\mathcal{R}_j$ rotates by $\Delta\theta_j^{-}$ is obtained by finding the boundary point closest to the origin. Then

$$\begin{cases} \mathcal{C}_{--} \leftarrow \mathcal{C}_{--} \cup \left\{ \begin{bmatrix} -\pi & \Delta\theta_j^{-} \end{bmatrix}^{T} \right\}, & \text{if } \mathbf{x}_c^{-} \in \mathcal{SA}_i\left(\left[\Delta\theta_j^{-}, 0\right]\right), \\ \mathcal{C}_{+-} \leftarrow \mathcal{C}_{+-} \cup \left\{ \begin{bmatrix} \pi & \Delta\theta_j^{-} \end{bmatrix}^{T} \right\}, & \text{if } \mathbf{x}_c^{-} \in \mathcal{SA}_i\left(\left[0, -\Delta\theta_j^{-}\right]\right). \end{cases} \tag{5.55}$$

Likewise, for the contact point $\mathbf{x}_c^{+}$ associated to $\Delta\theta_j^{+}$, the subsets are updated as follows:

$$\begin{cases} \mathcal{C}_{-+} \leftarrow \mathcal{C}_{-+} \cup \left\{ \begin{bmatrix} -\pi & \Delta\theta_j^{+} \end{bmatrix}^{T} \right\}, & \text{if } \mathbf{x}_c^{+} \in \mathcal{SA}_i\left(\left[-\Delta\theta_j^{+}, 0\right]\right), \\ \mathcal{C}_{++} \leftarrow \mathcal{C}_{++} \cup \left\{ \begin{bmatrix} \pi & \Delta\theta_j^{+} \end{bmatrix}^{T} \right\}, & \text{if } \mathbf{x}_c^{+} \in \mathcal{SA}_i\left(\left[0, \Delta\theta_j^{+}\right]\right). \end{cases} \tag{5.56}$$

The rotation angles at which $D\left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}, M_{\mathcal{R}_j}\right)$ and the swept region of $\mathcal{R}_i$ can also be derived in a similar way.

The moving line segments

The problem that finds the rotation angles at which an ellipse touches the moving line segments can be solved based on the distance between the center to the tangent line. Hence, the derivation is started with the following lemma.

Lemma 5.7 *Suppose that there is an elliptic robot $\mathcal{R}$. The distance from the center to the tangent line that is parallel with a given unit vector $\bar{\mathbf{v}}$ is*

$$d = \bar{\mathbf{v}}^T \mathbf{S}_{\mathcal{R}}^{-1} \bar{\mathbf{v}} \det \mathbf{S}_{\mathcal{R}}, \tag{5.57}$$

where $\mathbf{S}_{\mathcal{R}}$ is the shape matrix of the robot.

Proof. Without loss of generality, suppose that the center of $\mathcal{R}$ is the origin. The boundary point $\mathbf{x}$ is represented by $\mathbf{x} = \mathbf{S}_{\mathcal{R}}^{\frac{1}{2}} \mathbf{u}$ such that $\|\mathbf{u}\| = 1$ since $f(\mathbf{x}) = \mathbf{x}^T \mathbf{S}_{\mathcal{R}}^{-1} \mathbf{x} - 1 = 0$. The normal vector at $\mathbf{x}$ is $\mathbf{n} = \mathbf{S}_{\mathcal{R}}^{-1} \mathbf{x} = \mathbf{S}_{\mathcal{R}}^{-\frac{1}{2}} \mathbf{u}$. Let $\mathbf{x}^*$ be the contact point between the boundary and the tangent line that is parallel with $\bar{\mathbf{v}}$. Then $\bar{\mathbf{v}}^T \mathbf{S}_{\mathcal{R}}^{-\frac{1}{2}} \mathbf{u} = 0$, which leads to

$$\mathbf{u} = \frac{\mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}}^{-\frac{1}{2}} \bar{\mathbf{v}}}{\sqrt{\bar{\mathbf{v}}^T \mathbf{S}_{\mathcal{R}}^{-1} \bar{\mathbf{v}}}} \tag{5.58}$$

From Lemma 4.13, the distance is written as

$$d = \frac{1}{\sqrt{\mathbf{u}^T \mathbf{S}_{\mathcal{R}}^{-1} \mathbf{u}}} = \frac{\bar{\mathbf{v}}^T \mathbf{S}_{\mathcal{R}}^{-1} \bar{\mathbf{v}}}{\sqrt{\bar{\mathbf{v}}^T \mathbf{S}_{\mathcal{R}}^{-\frac{1}{2}} \mathbf{R}_{\frac{\pi}{2}}^T \mathbf{S}_{\mathcal{R}}^{-1} \mathbf{R}_{\frac{\pi}{2}} \mathbf{S}_{\mathcal{R}}^{-\frac{1}{2}} \bar{\mathbf{v}}}} \tag{5.59}$$

Because $\mathbf{S}_{\mathcal{R}}^{-\frac{1}{2}}\mathbf{R}_{\frac{\pi}{2}}\mathbf{S}_{\mathcal{R}}^{-\frac{1}{2}} = \sqrt{\det \mathbf{S}_{\mathcal{R}}^{-1}}\mathbf{R}_{\frac{\pi}{2}}$, this is transformed into

$$d = \frac{\bar{\mathbf{v}}^T \mathbf{S}_{\mathcal{R}}^{-1} \bar{\mathbf{v}} \det \mathbf{S}_{\mathcal{R}}}{\sqrt{\bar{\mathbf{v}}^T \bar{\mathbf{v}}}}. \tag{5.60}$$

Since $\bar{\mathbf{v}}$ is a unit vector, we obtain (5.57). □

In this section, the unit vector $\bar{\mathbf{v}}$ is denoted by

$$\bar{\mathbf{v}} = \left\| \hat{\mathbf{v}}^{new}_{\mathcal{R}_i|\mathcal{R}_j} \right\|^{-1} \hat{\mathbf{v}}^{new}_{\mathcal{R}_i|\mathcal{R}_j}. \tag{5.61}$$

Let $d_i : (-\pi, \pi] \rightarrow \mathbb{R}$ be a function that returns the distance from the center to the tangent line parallel with $\bar{\mathbf{v}}$ when the robot $\mathcal{R}_i$ rotates by a given angle $\Delta\theta_i$, expressed as

$$d_i\left(\Delta\theta_i\right) = \bar{\mathbf{v}}^T \mathbf{R}_{\Delta\theta_i} \mathbf{S}_{\mathcal{R}_i}^{-1} \mathbf{R}_{\Delta\theta_i}^T \bar{\mathbf{v}} \det \mathbf{S}_{\mathcal{R}_i}. \tag{5.62}$$

Likewise, let $d_j : (-\pi, \pi] \rightarrow \mathbb{R}$ be a function of robot $\mathcal{R}_j$, so that

$$d_j\left(\Delta\theta_j\right) = \bar{\mathbf{v}}^T \mathbf{R}_{\Delta\theta_j} \mathbf{S}_{\mathcal{R}_j}^{-1} \mathbf{R}_{\Delta\theta_i}^T \bar{\mathbf{v}} \det \mathbf{S}_{\mathcal{R}_j}. \tag{5.63}$$

Let D be the length of the projection of $\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}$ onto a perpendicular vector of $\bar{\mathbf{v}}$ such that

$$D = \left\| \mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j} - \left(\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}^T \bar{\mathbf{v}} \right) \bar{\mathbf{v}} \right\|. \tag{5.64}$$

Then if an ellipse $\mathcal{R}_1$ touch the line segments which is a part of the swept regions $\mathcal{SA}\left(\Theta_2\right)$ and cannot expand both of the regions in the directions they rotated,

$$D = d_i\left(\Delta\theta_i\right) + d_j\left(\Delta\theta_j\right) \tag{5.65}$$

holds. Due to the reciprocity of rotation, the above equation is solved assuming that $\Delta\theta_i = \Delta\theta_j$ or $\Delta\theta_i = -\Delta\theta_j$. By putting $\cos\Delta\theta_i = \left(1-t^2\right)/\left(1+t^2\right)$ and $\sin\Delta\theta_i = 2t/\left(1+t^2\right)$, (5.65) is transformed into two univariate polynomial of t with degrees of 4. When the polynomials are solved, the expected contact points on the rotated $\mathcal{R}_i$ are calculated. If these points are on the boundary of the other area, this results are added into one of the four subsets.

Next, it is possible to the elliptic robot touches a point the other robot rotates through. This means that $d_i\left(\Delta\theta_i\right) = M_i$ or $d_i\left(\Delta\theta_j\right) = M_j$. This case can be easily derived by Lemma 4.13. When the solutions are obtained, these are inserted into four subsets in a similar way in (5.55) and (5.56).

5.4.2 Updating the Candidates Sets

The neighbors of a given candidate subset are defined as the sets of which either two elements of the subscript is in common. For example, the neighbors of $\mathcal{C}_{-+}$ are $\mathcal{C}_{--}$ and $\mathcal{C}_{++}$. When the limit of rotation in a particular combination of the directions is determined, the neighbor sets are affected. Suppose that a collision between the two robots occurs when $\mathcal{R}_i$ and $\mathcal{R}_j$ rotate by $-\pi < \Delta\theta_i^- \leq 0$ and $0 \leq \Delta\theta_j^+ < \pi$, respectively, and the swept areas cannot expand anymore in those directions. This result has to be considered in $\mathcal{C}_{--}$ because it is associated with the clockwise rotation of the robot $\mathcal{R}_i$. Hence, the rotation angles of robot $\mathcal{R}_j$ is derived at which it contacts with $\mathcal{SA}_i\left(\left[\Delta\theta_i^-, 0\right]\right)$, which is added in $\mathcal{C}_{--}$. In a similar way, the rotation angels of robot $\mathcal{R}_i$ at which it contacts with $\mathcal{SA}_j\left(\left[0, \Delta\theta_j^+\right]\right)$, which is added in $\mathcal{C}_{++}$.

The update process can be executed by the method in Section 4.6. When $\mathcal{C}_{--}$ is updated, the region $\mathcal{SA}_i\left(\left[\Delta\theta_i^-, 0\right]\right)$ is regarded as $\mathcal{SA}_i\left(\left\{\Delta\theta_i^-\right\}\right)$ because the other area is already considered in the preceding step. Hence, this is equivalent to find the collision-free rotation angles of $\mathcal{R}_j$ induced by a moving elliptic obstacle $\mathcal{F}_i\left(\Delta\theta_i^-\right)$ with $-\hat{\mathbf{v}}_{\mathcal{R}_i|\mathcal{R}_j}^{new}$. On the other hand, when $\mathcal{C}_{++}$ is updated, the region $\mathcal{SA}_j\big(\big[0, \Delta\theta_j^+\big]\big)$ is regarded as $\mathcal{SA}_j\big(\big\{\Delta\theta_j^+\big\}\big)$. The problem is solved by finding the collision-free rotation angles of $\mathcal{R}_i$ induced by a moving elliptic obstacle $\mathcal{F}_j\big(\Delta\theta_j^+\big)$ with $\hat{\mathbf{v}}_{\mathcal{R}_i|\mathcal{R}_j}^{new}$.

5.4.3 Calculation of Collision-free Reciprocal Rotation Angles

Suppose that the four subsets are generated as explained in Section 5.4.1. Because all the candidates have to be considered, the set $\mathcal{C}$ is defined as $\mathcal{C} = \mathcal{C}_{--} \cup \mathcal{C}_{-+} \cup \mathcal{C}_{+-} \cup \mathcal{C}_{++}$ at first. In order to establish the priority order in the set $\mathcal{C}$, let $\mathcal{L}_2$ denote a lexicographically ordered set of pairs of positive real numbers between 0 and π, and $\mathcal{E}_2 : \mathbb{R}^2 \to \mathcal{L}_2$ be a function that maps a pair of the rotation angles to a pair with the elements of ascending sorted absolute values of the angles. For example, if $\mathbf{c}_k = \{0.3, -0.2\} \in \mathcal{C}$, then $\mathcal{E}_2(\mathbf{c}_k) = \{0.2, 0.3\}$. As a result, the problem that finds the contact rotation angles pair is described as

$$\underset{\mathbf{c}_k \in \mathcal{C}}{\operatorname{argmin}} \quad \mathcal{E}_2(\mathbf{c}_k)\,. \tag{5.66}$$

Suppose c^* is the determined pair. Then the subset containing c^* is excluded when the entire set is redefined, and its neighbors are updated as mentioned in Section 5.4.2. Next, the contact rotation angles pair is determined by (5.66) again. This process is iterated three times until all the combinations of rotation directions are investigated. The pseudo code of the proposed method is summarized in Algorithm 5.1.

Algorithm 5.1 Calculation of the collision-free reciprocal rotation angles

```
Input: relative position $\mathbf{p}_{\mathcal{R}_i|\mathcal{R}_j}$, relative velocity $\hat{\mathbf{v}}^{new}_{\mathcal{R}_i|\mathcal{R}_j}$, time-to-contact $T_F$
Output: collision-free reciprocal rotation angels $\Theta_i$ and $\Theta_j$
$\Theta_i \leftarrow [-\pi\pi]$; $\Theta_j \leftarrow [-\pi, \pi]$;
$\mathcal{H} \leftarrow \{--, -+, +-, ++\}$;
Calculating the four subsets $\mathcal{C}_{--}$, $\mathcal{C}_{-+}$, $\mathcal{C}_{+-}$, and $\mathcal{C}_{++}$;
while card$(\mathcal{H}) > 0$ do
    $\mathcal{C} \leftarrow \bigcup_{i\in\mathcal{H}} \mathcal{C}_i$;
    $\mathbf{c}^* \leftarrow \text{argmin}_{\mathbf{c}\in\mathcal{C}} \mathcal{E}_2(\mathbf{c})$;
    if $\mathbf{c}^* \in \mathcal{C}_{--}$ then
        $\Theta_1 \leftarrow \Theta_1 \cap [c_1^*, \pi]$; $\Theta_2 \leftarrow \Theta_2 \cap [c_2^*, \pi]$;
        $\mathcal{H} \leftarrow \mathcal{H} \setminus \{--\}$;
    else if $\mathbf{c}^* \in \mathcal{C}_{-+}$ then
        $\Theta_1 \leftarrow \Theta_1 \cap [c_1^*, \pi]$; $\Theta_2 \leftarrow \Theta_2 \cap [-\pi, c_2^*]$;
        $\mathcal{H} \leftarrow \mathcal{H} \setminus \{-+\}$;
    else if $\mathbf{c}^* \in \mathcal{C}_{+-}$ then
        $\Theta_1 \leftarrow \Theta_1 \cap [-\pi, c_1^*]$; $\Theta_2 \leftarrow \Theta_2 \cap [c_2^*, \pi]$;
        $\mathcal{H} \leftarrow \mathcal{H} \setminus \{+-\}$;
    else
        $\Theta_1 \leftarrow \Theta_1 \cap [-\pi, c_1^*]$; $\Theta_2 \leftarrow \Theta_2 \cap [-\pi, c_2^*]$;
        $\mathcal{H} \leftarrow \mathcal{H} \setminus \{++\}$;
    end if
    Updating the neighbors of the set containing $\mathbf{c}^*$
end while
```

5.4.4 An Example

An example is presented where the proposed method is applied to the situation described in Fig. 5.5. Let $\mathcal{R}_1$ be an elliptic robot rotating at the origin and $\mathcal{R}_2$ be another elliptic robot moving at $\mathbf{p}_{\mathcal{R}_2} = \begin{bmatrix}0 & 1.1\end{bmatrix}^T$ with $\mathbf{v}_{\mathcal{R}_2} = \begin{bmatrix}1 & 0\end{bmatrix}^T$. The orientations of the two robots are $\theta_{\mathcal{R}_1} = 70°$ and $\theta_{\mathcal{R}_2} = 60°$. Also, it is assumed that $T_F = 2$. Without rotation, the swept regions are the same presented in Fig. 5.5(a).

According to the derivation in Section 5.4.1, the candidates of the first contact pair of rotation angles can be calculated. These are listed in Table 5.1

Table 5.1 The candidates of the first contact rotation angles

Rotation Directions	$\Delta\theta_i$	$\Delta\theta_j$
$--$	$-42.90°$	$-42.90°$
$++$	$71.96°$	$71.96°$
$++$	$92.90°$	$92.90°$

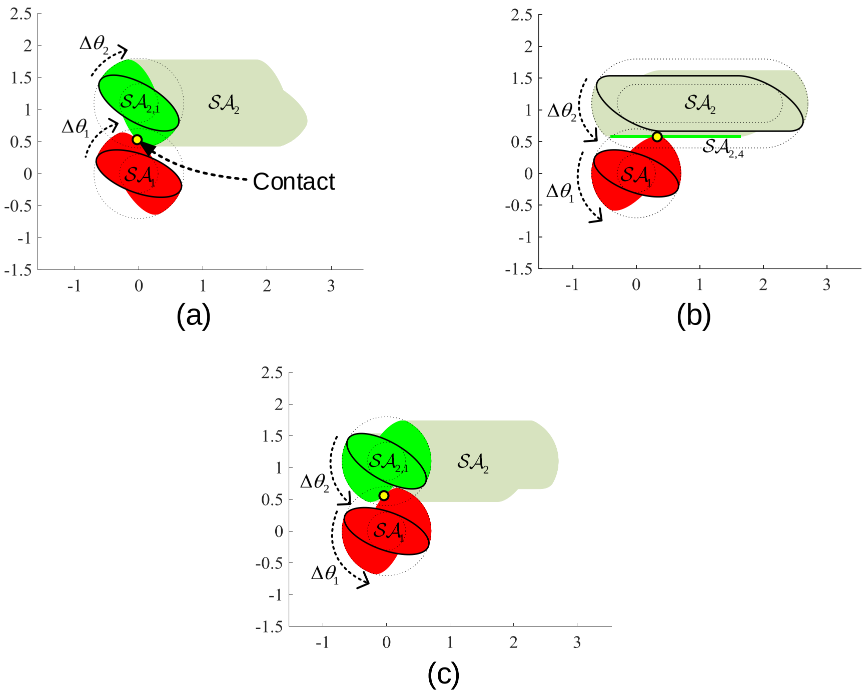


Figure 5.6 The candidates of the first contact rotation angles of the swept areas $\mathcal{SA}_1$ and $\mathcal{SA}_2$: (a) the two areas $\mathcal{SA}_1$ and $\mathcal{SA}_{2,1}$ touch each other when $\Delta\theta_1 = \Delta\theta_2 = -42.90°$; (b) the area $\mathcal{SA}_1$ and the line segment $\mathcal{SA}_{2,4}$ touch each other when $\Delta\theta_1 = \Delta\theta_2 = 71.96°$; (c) the two areas $\mathcal{SA}_1$ and $\mathcal{SA}_{2,1}$ touch each other when $\Delta\theta_1 = \Delta\theta_2 = 92.90°$.

and presented in Fig. 5.6. Among these three candidates, the first contact pair is determined as $\{-42.90°, -42.90°\}$ by (5.66). As a result, the intervals are assigned to $\Theta_1 = [-42.90°, 180°]$ and $\Theta_2 = [-42.90°, 180°]$.

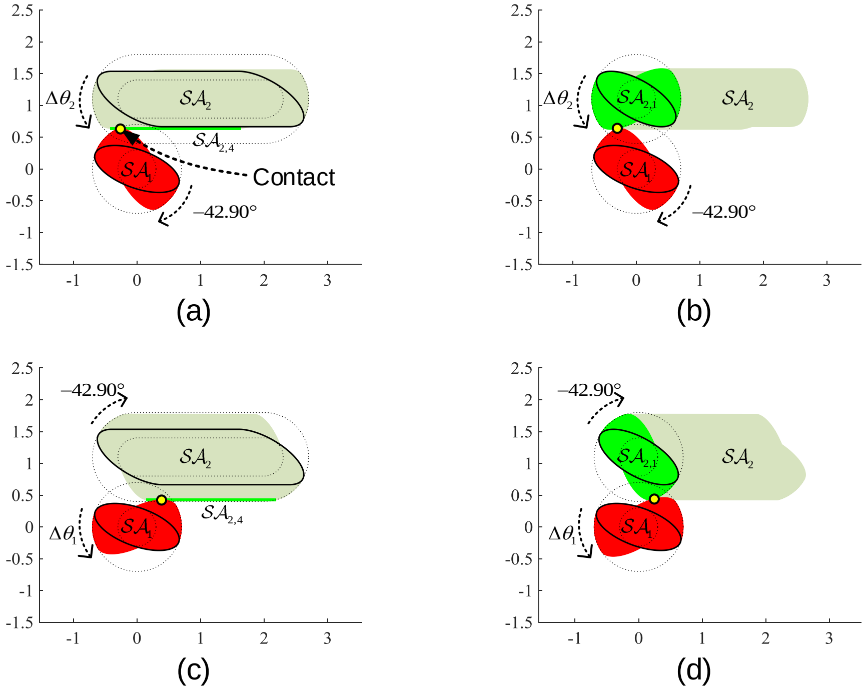


Figure 5.7 The newly added candidates of the second contact rotation angles of the swept areas $\mathcal{SA}_1$ and $\mathcal{SA}_2$: (a) the two areas $\mathcal{SA}_1$ and and the line segment $\mathcal{SA}_{2,4}$ touch each other when $\Delta\theta_2 = 63.75°$; (b) the area $\mathcal{SA}_1$ and $\mathcal{SA}_{2,1}$ touch each other when $\Delta\theta_2 = 66.07°$; (c) the two areas $\mathcal{SA}_1$ and and the line segment $\mathcal{SA}_{2,4}$ touch each other when $\Delta\theta_1 = 48.44°$; (d) the area $\mathcal{SA}_1$ and $\mathcal{SA}_{2,1}$ touch each other when $\Delta\theta_1 = 53.92°$.

Table 5.2 The candidates of the second contact rotation angles

Rotation Directions	$\Delta\theta_i$	$\Delta\theta_j$
$-+$	$-42.90°$	$63.75°$
$-+$	$-42.90°$	$66.07°$
$+-$	$48.44°$	$-42.90°$
$+-$	$53.92°$	$-42.90°$
$++$	$71.96°$	$71.96°$
$++$	$92.90°$	$92.90°$

Because $\{-42.90°, -42.90°\} \in \mathcal{C}_{--}$, its neighbor sets $\mathcal{C}_{-+}$ and $\mathcal{C}_{+-}$ have to be updated. These are presented in Fig. 5.7. Also, the candidate pairs of the second contact angles are listed in Table 5.2. The candidates set $\mathcal{C}$ is the union of $\mathcal{C}_{-+}$, $\mathcal{C}_{+-}$, and $\mathcal{C}_{++}$. Among these candidates, the pair of the second contact angles is determined as $\{48.44°, -42.90°\}$, which is contained in $\mathcal{C}_{+-}$. As a result, the intervals are set to $\Theta_1 = [-42.90°, 48.44°]$ and $\Theta_2 = [-42.90°, 180°]$.

Next, the rest subsets are updated from the determination of the second contact angles. Although the neighbor sets of $\mathcal{C}_{+-}$ are $\mathcal{C}_{--}$ and $\mathcal{C}_{++}$, $\mathcal{C}_{++}$ is only updated since the rotation angles in the clockwise directions are already deter-

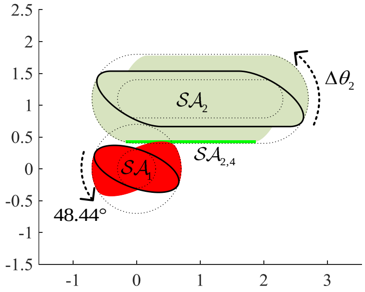


Figure 5.8 The newly added candidate of the third contact rotation angles of the swept areas $\mathcal{SA}_1$ and $\mathcal{SA}_2$. The two areas $\mathcal{SA}_1$ and and the line segment $\mathcal{SA}_{2,4}$ touch each other when $\Delta\theta_2 = 102.90°$.

Table 5.3 The candidates of the third contact rotation angles

Rotation Directions	$\Delta\theta_i$	$\Delta\theta_j$
$-+$	$-42.90°$	$63.75°$
$-+$	$-42.90°$	$66.07°$
$++$	$48.44°$	$102.90°$
$++$	$71.96°$	$71.96°$
$++$	$92.90°$	$92.90°$

mined. The newly added candidate in $\mathcal{C}_{++}$ is presented in Fig. 5.8. Moreover, the candidate pairs of the third contact angles are listed in Table 5.3. The candidates set $\mathcal{C}$ is now the union of $\mathcal{C}_{-+}$ and $\mathcal{C}_{++}$. Among these candidates, the pair of the third contact angles is determined as $\{-42.90°, 63.75°\}$, which is contained in $\mathcal{C}_{-+}$. As a result, the intervals are determined as $\Theta_1 = [-42.90°, 48.44°]$ and $\Theta_2 = [-42.90°, 63.75°]$.

Finally, $\mathcal{C}_{++}$ is only updated again. The newly added candidate in $\mathcal{C}_{++}$ is presented in Fig. 5.9. Moreover, the candidate pairs of the final contact angles are listed in Table 5.4. Among these candidates in $\mathcal{C} = \mathcal{C}_{++}$, the pair of the final contact angles is determined as $\{48.44°, 102.90°\}$. However, this result

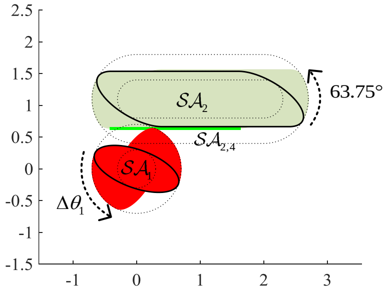


Figure 5.9 The newly added candidate of the fourth contact rotation angles of the swept areas $\mathcal{SA}_1$ and $\mathcal{SA}_2$. The two areas $\mathcal{SA}_1$ and and the line segment $\mathcal{SA}_{2,4}$ touch each other when $\Delta\theta_1 = 82.90°$.

Table 5.4 The candidates of the fourth contact rotation angles

Rotation Directions	$\Delta\theta_i$	$\Delta\theta_j$
++	48.44°	102.90°
++	82.90°	63.75°
++	71.96°	71.96°
++	92.90°	92.90°

does not affect the resultant intervals. Consequently, $\Theta_1 = [-42.90°, 48.44°]$ and $\Theta_2 = [-42.90°, 63.75°]$, which is equivalent to the result presented in Fig. 5.5(e).

5.5 Summary

This chapter has suggested multi-robot collision avoidance algorithm for holonomic elliptic robots. Contrary to obstacles, the motion of other robots is difficult to expect because they continuously sense their surrounding environment and replan their action based on the input. To overcome this problem, this chapter employed the concept of reciprocity. Asa a result, the two steps are modified: the generation of velocity obstacles and the calculation of collision-free rotation angles.

In the first step, the hybrid reciprocal velocity obstacle is adapted in order to prevent the oscillation and the reciprocal dance, by translating the apex of the cone. Next, the method that calculates the collision-free reciprocal rotation angles have been proposed on the basis of the reciprocity of rotation. Because there are four combinations of the rotation directions of two robots, the proposed method defines four candidates set of the end points of the rotation angles. Afterward, the intervals are determined by deciding the endpoints of the rotation about one combination of the directions at a time.

Chapter 6

Implementation and Simulations

6.1 Implementation Setups

The proposed algorithm was implemented in Visual Studio C++ on a PC equipped with Intel Core i7-3770 3.40GHz CPU by adapting the RVO2 library [117]. Because the library was implemented for circular robots and polygonal static obstacles, we extended it to deal with elliptic robots and obstacles. In addition, the Eigen library [52] was utilized to solve the polynomial equations and perform matrix operations.

In this chapter, the parameters were assigned as follows. The robots were ellipses whose major and minor radii were 1m and 0.3m. They detected obstacles and other robots in the range of $\rho = 10$m with a sampling period $\Delta t = 0.2$s and avoided them by considering collisions within $\tau = 5$s. The preferred linear speed was $v_{\mathcal{R}_i}^{pref} = \sqrt{2}/2$m/s. Moreover, their motion was limited by the dynamic constraints: $v_{\mathcal{R}_i}^{max} = 1$m/s, $w_{\mathcal{R}_i}^{max} = 1$rad/s, $a_{\mathcal{R}_i}^{max} = 1\text{m/s}^2$, $\alpha_{\mathcal{R}_i}^{max} = 1\text{rad/s}^2$.

On top of that, $w_{\mathcal{R}_i}^{max}$ was additionally bounded to fairly evaluate and compare the performance. Since the motion of mobile robots was controlled by their wheels, the maximum wheel velocity is set to $v_{\mathcal{R}_i}^{max}$. Then

$$w_{\mathcal{R}_i}^{max} \leq \frac{v_{\mathcal{R}_i}^{max} - \|\mathbf{v}_{\mathcal{R}_i}\|}{M_i}. \tag{6.1}$$

6.2 Obstacle Avoidance

In order to verify the performance of the proposed algorithm for the obstacle avoidance, the following algorithms were compared.

- *Circle* presented in [41] assumed a robot and obstacles were bounded by circles.

- *Ellipse-N* assumed they were bounded by ellipses and robots were only allowed translational motion.

- *Ellipse-H1* presented in [59] assumed they were bounded by ellipses and robots were holonomic. The angular velocity was determined proportional to the change rate of the boundary line of the EBVO with respect to its orientation.

- *Ellipse-H2* is the proposed algorithm, where robots and obstacles were bounded by ellipses and robots are holnomic. However, the angular velocity was determined with which the robot could circumvent obstacles with the minimum deviation in [73].

- *Polygon* presented in [45] assumed that a robot and obstacles were bounded by polygons approximating ellipses and robots were holonomic.

The radius of robots and obstacles in the *Circle* algorithm was denoted by the major radius in the other algorithms, so that $r = 1$m. In the *Polygon* algorithm, the ellipses were approximated with eight piecewise lines by using the method proposed in [12] and the granularity parameter in [45] was set to $\delta = 20$. Theses algorithms were implemented and tested in the following three scenarios.

6.2.1 Line scenario of a robot and an obstacle

An elliptic robot $\mathcal{R}$ and an circular obstacle $\mathcal{O}$ with radius of 0.5m started at the opposite ends and moved along the same line. At the beginning, $\mathcal{R}$ was stalled at $\mathbf{p}_{\mathcal{R}} = (0.0\text{m}, 0.0\text{m})$ with $\theta_{\mathcal{R}} = 45.0\,\text{deg}$, and $\mathcal{O}$ moved with $\mathbf{v}_{\mathcal{O}} = (-0.5\text{m/s}, 0.0\text{m/s})$ at $\mathbf{p}_{\mathcal{O}} = (8.0\text{m}, 0.0\text{m})$. The robot's goal position $\mathbf{p}_{\mathcal{R}}^{goal}$ was equal to the initial position of the obstacle $\mathbf{p}_{\mathcal{O}}$. The above descriptions are presented in Fig. 6.1 and summarized in Table 6.1.

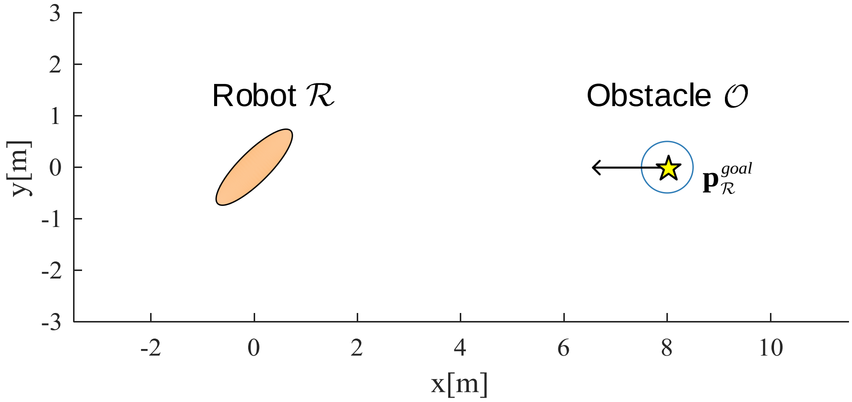


Figure 6.1 Initial states of a robot and an obstacle in the line scenario. The orange ellipse represents the robot $\mathcal{R}$, and the blue circle indicates the obstacle $\mathcal{O}$, which is moving to the left at -0.5m/s. The goal position $\mathbf{p}_{\mathcal{R}}^{goal}$ of the robot is marked as the yellow star, which is equal to the initial position of the obstacle.

Table 6.1 The initial parameters of the robot and the obstacle in the line scenario

	Position[m]	Orientation[°]	Velocity[m/s]	Destination[m]
Robot $\mathcal{R}$	$(0, 0)$	45.0	$(\ \ 0.0, 0.0)$	$(8.0, 0.0)$
Obstacle $\mathcal{O}$	$(8, 0)$		$(-0.5, 0.0)$	

The resultant traces of the robot and the obstacle using the five algorithms are presented in Figs. 6.2–6.6, where the lighter the color of an object is, the older its trace is. The robot in Fig. 6.2 traveled more distance than that in Fig. 6.3. That was because the robot's footprint in Fig. 6.2 was larger than that in Fig. 6.3, so that it had to take a long way around. Meanwhile, the difference between which side the robot passed the obstacle on was trivial. In order to avoid deadlocks due to the perfect symmetry, a small random noise was added to the preferred linear velocity. This random noise determined which side the robot avoided the obstacle on.

In addition, the robots in Figs. 6.4 and 6.5 exploited their rotation to decrease the traveled distance compared to that in Fig. 6.3. The difference was the speed of the convergence in the robot's orientation. *Ellipse-H2* calculated the optimal orientation with a closed form solution, whereas *Ellipse-H1* solved the problem by continuing to find a better approximation than before. Hence, the orientation changed gradually in Fig. 6.4(d), but it varied quickly in Fig. 6.5(d). As a result, the proposed algorithm could cope with a change in the environment more swiftly.

However, the *Polygon* algorithm suffered from the deadlock because the robot could not choose the lateral velocity until $t = 25.2$s in Fig. 6.6. Afterward, the robot broke the deadlock through rotation and detoured the obstacle. However, it took so much time since the robot should be close enough to the

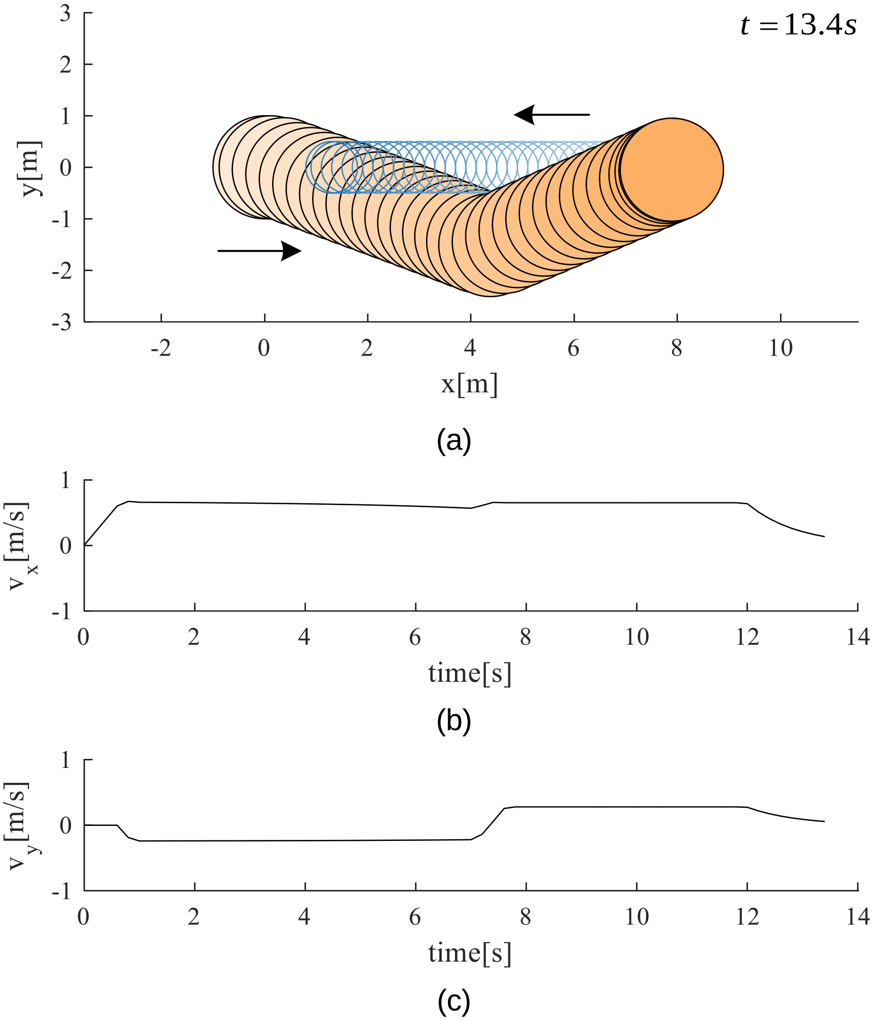
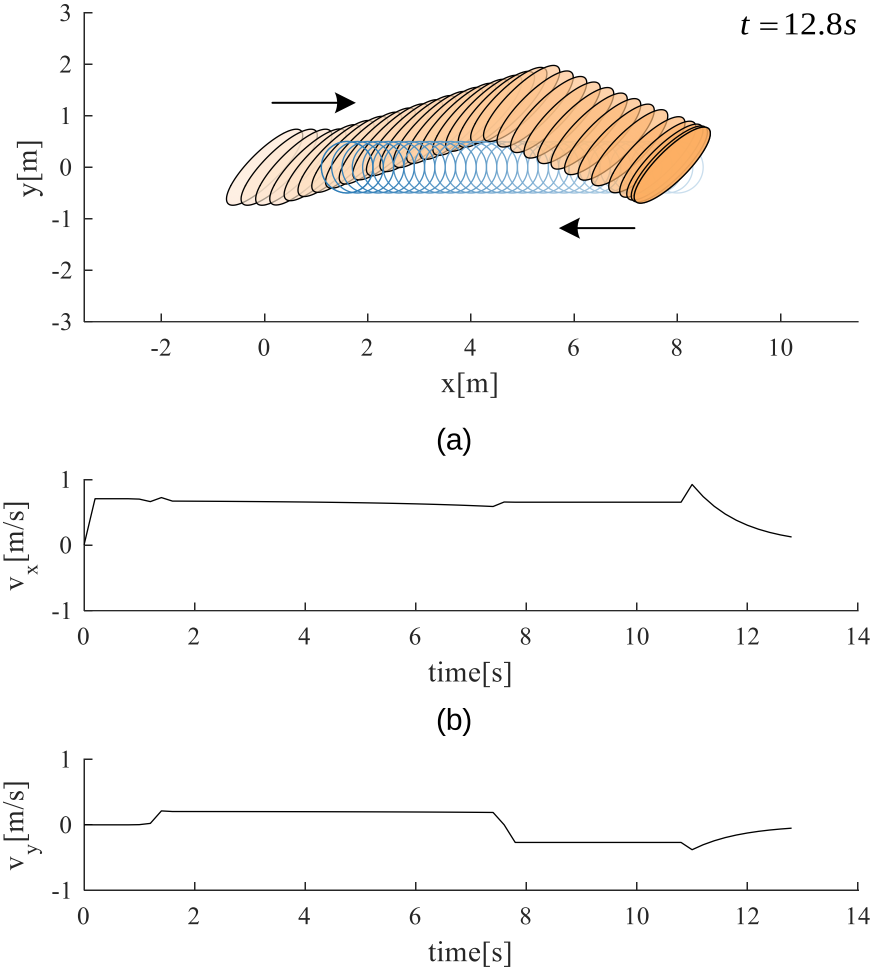


Figure 6.2 The simulation result in the line scenario using the *Circle* algorithm: (a) the resultant path; (b)–(c) the robot's linear velocities with respect to the time. The robot reached its goal at $t = 13.4$s.



(a)

(b)

(c)

Figure 6.3 The simulation result in the line scenario using the *Ellipse-N* algorithm: (a) the resultant path; (b)–(c) the robot's linear velocities with respect to the time. The robot reached its goal at $t = 12.8$s.

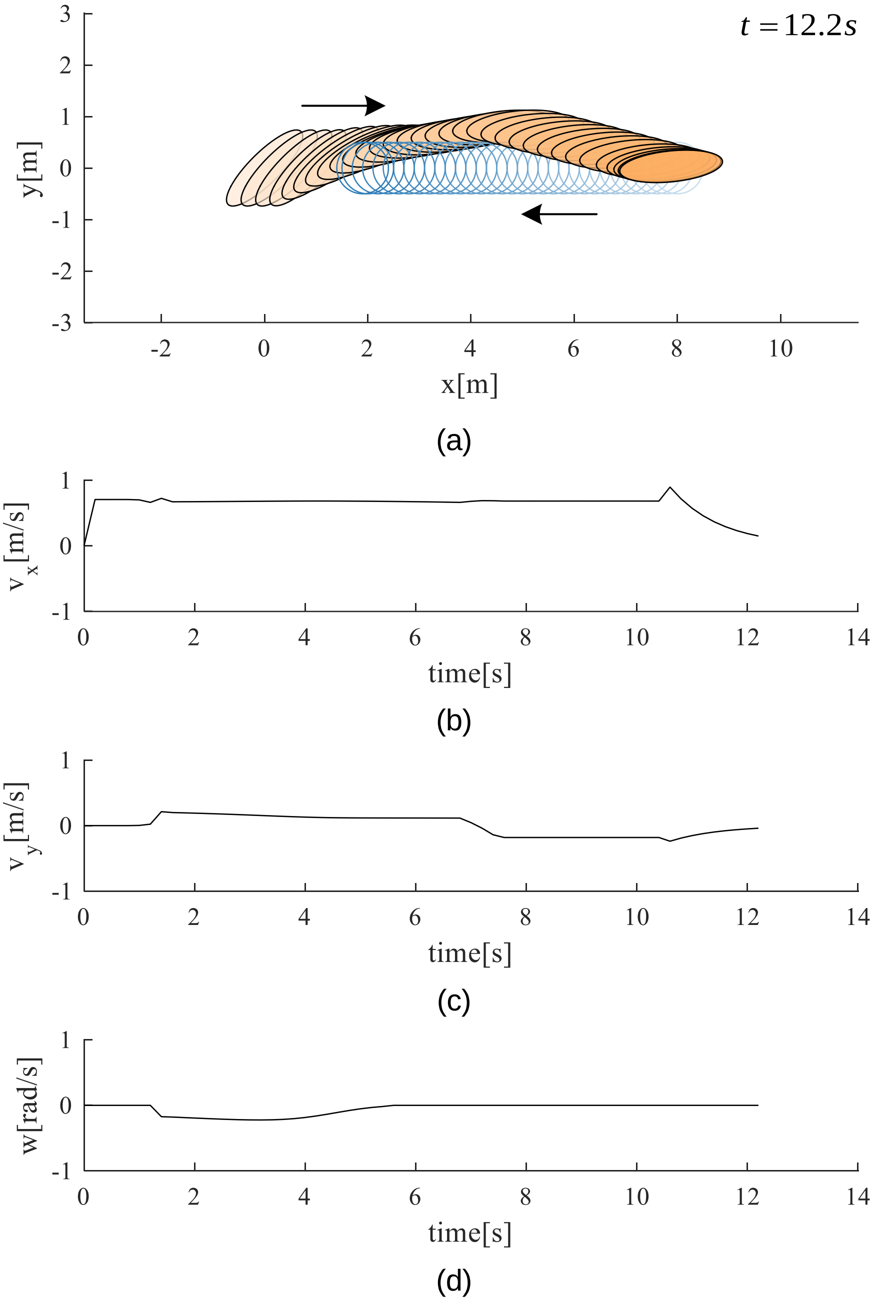
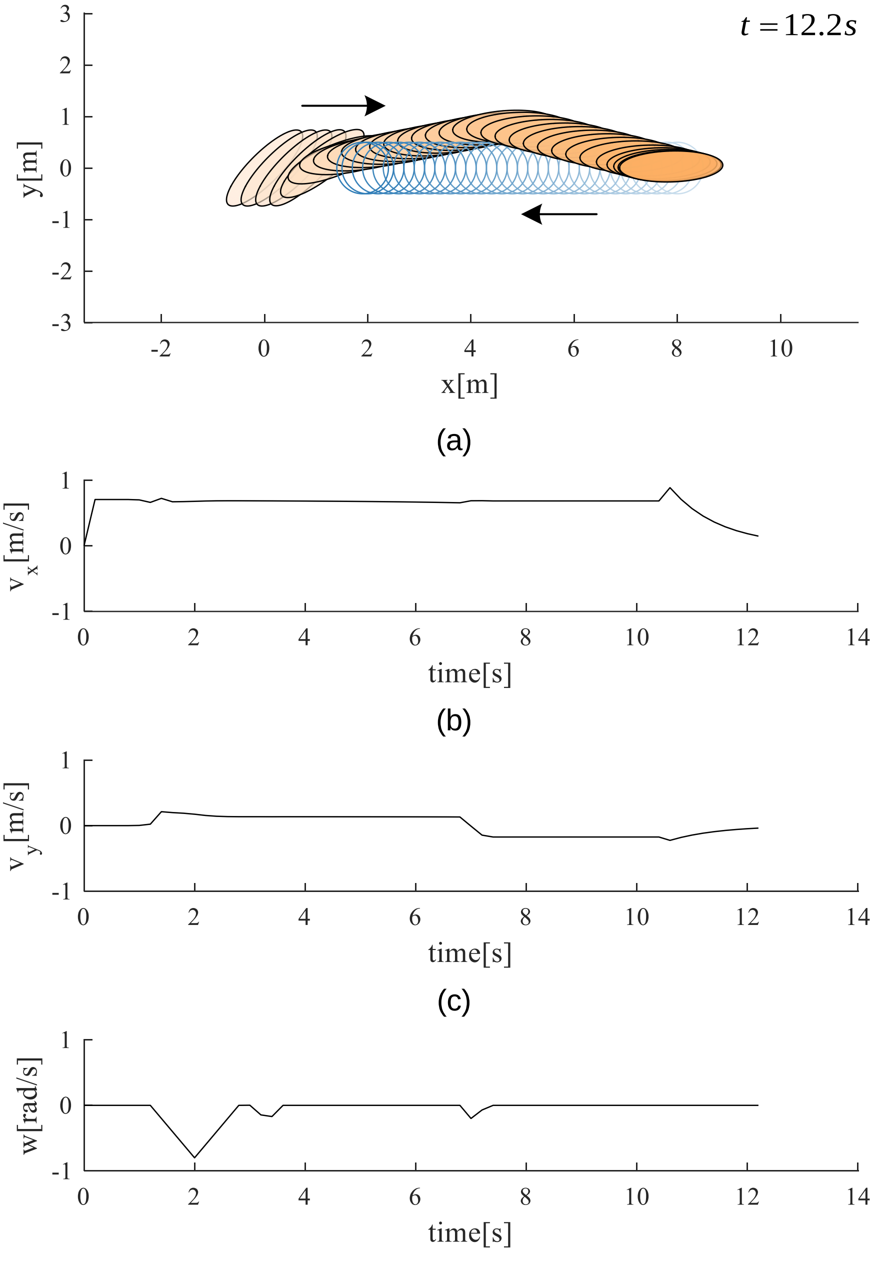


Figure 6.4 The simulation result in the line scenario using the *Ellipse-H1* algorithm: (a) the resultant path; (b)–(d) the robot's linear and angular velocities with respect to the time. The robot reached its goal at $t = 12.2$s.



(a)

(b)

(c)

(d)

Figure 6.5 The simulation result in the line scenario using the *Ellipse-H2* algorithm: (a) the resultant path; (b)–(d) the robot's linear and angular velocities with respect to the time. The robot reached its goal at $t = 12.2$s.

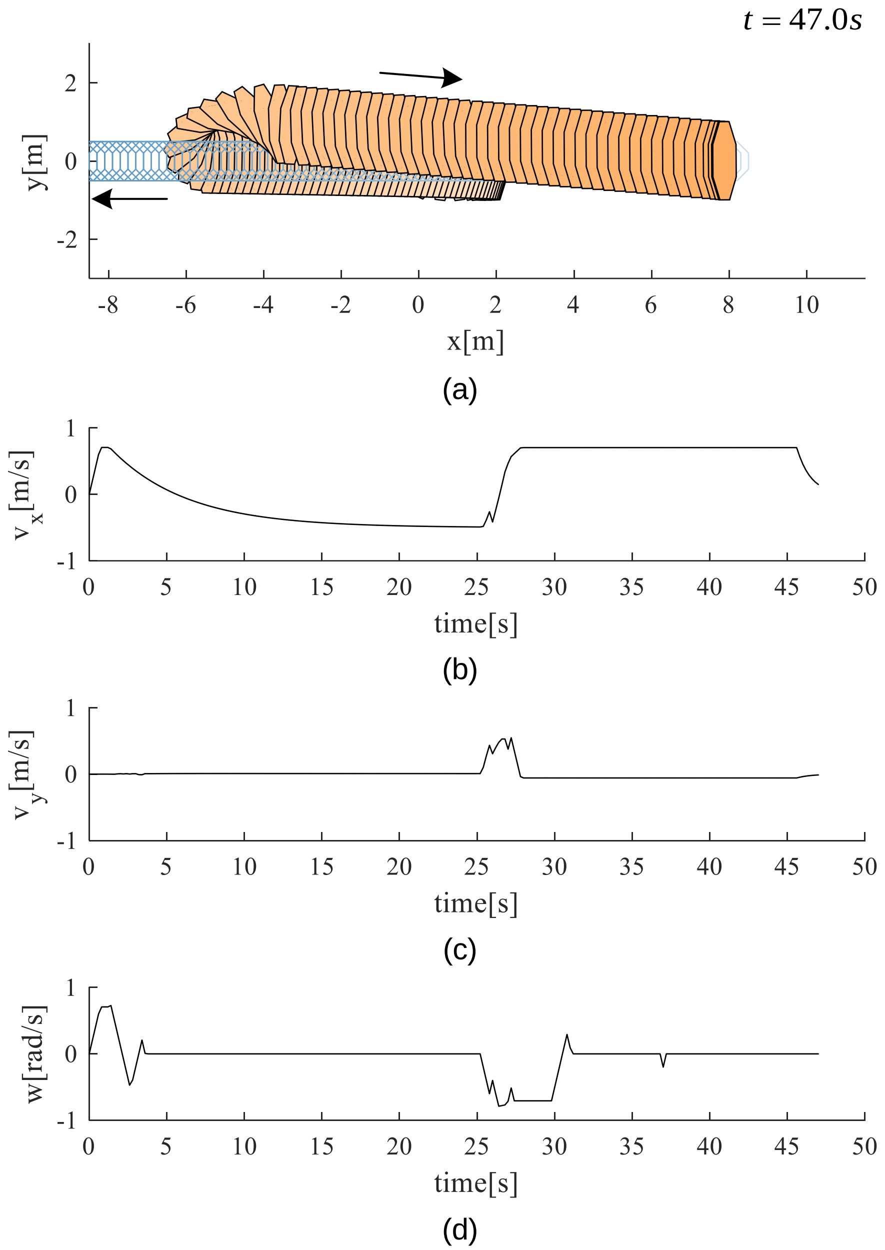


Figure 6.6 The simulation result in the line scenario using the *Polygon* algorithm: (a) the resultant path; (b)–(d) the robot's linear and angular velocities with respect to the time. The robot reached its goal at $t = 47.0$s.

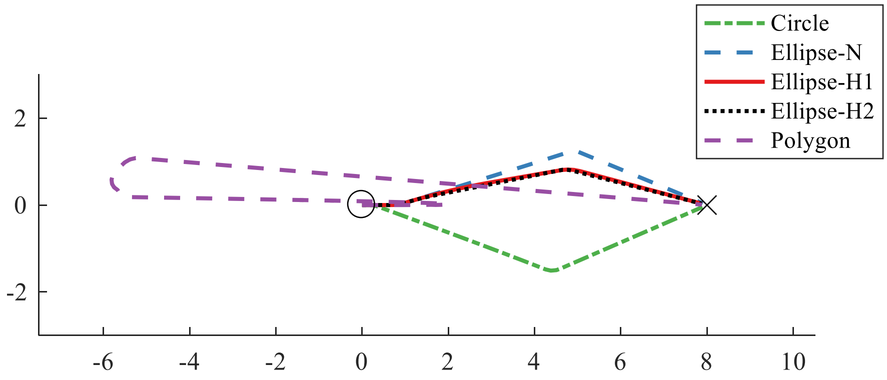


Figure 6.7 The comparison between the paths generated by each of the algorithms in the line scenario. The black circle and x-mark are the start and goal positions. The green dash-dot, blue dashed, red solid, black doted, and violet dashed lines represents the paths of *Circle*, *Ellipse-N*, *Ellipse-H1*, *Ellipse-H2*, and *Polygon*, respectively.

Table 6.2 Simulation results in the line scenario

Algorithm	Travel distance[m]	Travel time[s]	Computation time[ms]
Circle	8.460	13.4	0.002
Ellipse-N	8.322	12.8	0.013
Ellipse-H1	8.068	12.2	0.014
Ellipse-H2	8.064	12.2	0.059
Polygon	23.965	47.0	0.692

obstacle to do so. As a result, the robot arrived at the destination later than when applying other algorithms.

Figure 6.7 compares the paths for each of the algorithms at the same time, and Table 6.2 presents the numerical results of the simulation. The travel distance of the proposed algorithm, *Ellipse-H2*, was almost the same with that of *Ellipse-H1*, but 3.11% shorter than that of *Ellipse-N* and 4.69% shorter than that of *Circle*. Similarly, the travel times of the proposed algorithm and *Ellipse-*

H1 were equal, but theirs are 4.69% shorter than that of *Ellipse-N* and 8.96% shorter than that of *Circle*. Although the computation time of the proposed algorithm was about three times longer than those of *Ellipse-H1* and *Ellipse-N* and thirty times longer than that of *Circle*, the differences were trivial because they were far shorter than the sampling period $\Delta t = 0.2$s.

Meanwhile, the *Polygon* algorithm was performed worse than the others due to the deadlock in terms of the travel distance and time. In addition, it had the longest execution time because of multiple computations of Minkowski sum to calculate the C-obstacle according to the orientation of the robot.

6.2.2 Multiple moving obstacles scenario

An elliptic robot and three elliptic obstacles were moving in the workspace. Similar to the previous scenario, the robot and an obstacle $\mathcal{O}_1$ started at the opposite ends and moved along the same line. At the beginning, the robot was halted at $\mathbf{p}_{\mathcal{R}} = (0.0\text{m}, 0.0\text{m})$ with $\theta_{\mathcal{R}} = -100°$. The obstacle $\mathcal{O}_1$ whose shape matrix was $\mathbf{S}_{\mathcal{O}_1} = \left[\begin{smallmatrix} 0.20 & -0.20 \\ -0.20 & 0.50 \end{smallmatrix}\right]$ moved to the robot with $\mathbf{v}_{\mathcal{O}_1} = (-0.5\text{m/s}, -0.5\text{m/s})$ at $\mathbf{p}_{\mathcal{O}_1} = (5.0\text{m}, 5.0\text{m})$. The robot's goal position $\mathbf{p}_{\mathcal{R}}^{goal}$ was also the same with the initial position of the obstacle $\mathcal{O}_1$. The other two obstacles, $\mathcal{O}_2$ and $\mathcal{O}_3$, were across the robot's path to the goal and had the same shape matrix $\mathbf{S}_{\mathcal{O}_2} = \mathbf{S}_{\mathcal{O}_3} =$

Table 6.3 The initial parameters of the robot and the three obstacles in the multiple moving obstacles scenario

	Position[m]	Shape[m^2]	Velocity[m/s]	Destination[m]
Robot $\mathcal{R}$	$(0.0, 0.0)$	$\left[\begin{smallmatrix} 0.97 & -0.16 \\ -0.16 & 0.12 \end{smallmatrix}\right]$	$(0.0, 0.0)$	$(5.0, 5.0)$
Obstacle $\mathcal{O}_1$	$(5.0, 5.0)$	$\left[\begin{smallmatrix} 0.20 & -0.20 \\ -0.20 & 0.50 \end{smallmatrix}\right]$	$(-0.5, -0.5)$	
Obstacle $\mathcal{O}_2$	$(4.0, 1.0)$	$\left[\begin{smallmatrix} 0.20 & 0.20 \\ 0.20 & 0.50 \end{smallmatrix}\right]$	$(-0.5, 0.5)$	
Obstacle $\mathcal{O}_3$	$(-2.0, 10.0)$	$\left[\begin{smallmatrix} 0.20 & 0.20 \\ 0.20 & 0.50 \end{smallmatrix}\right]$	$(0.5, -0.5)$	

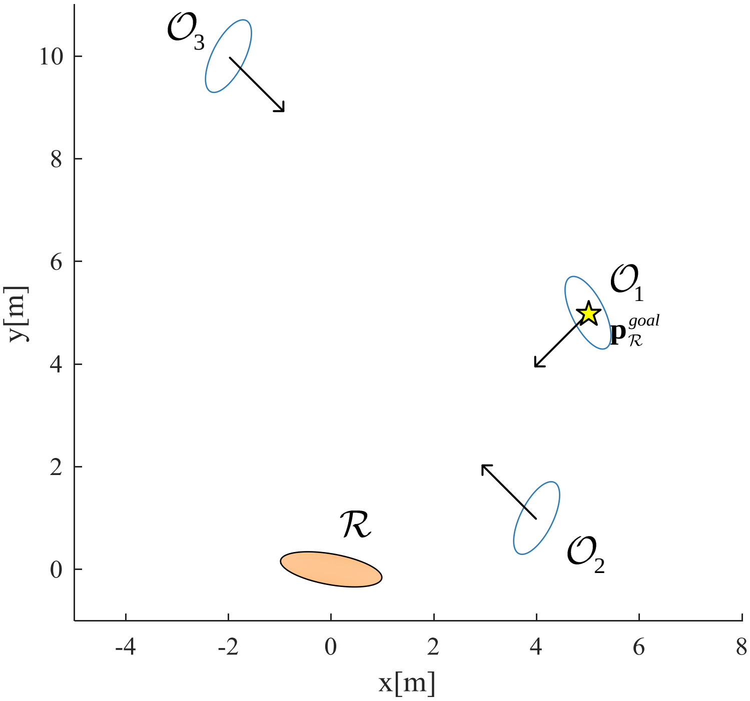
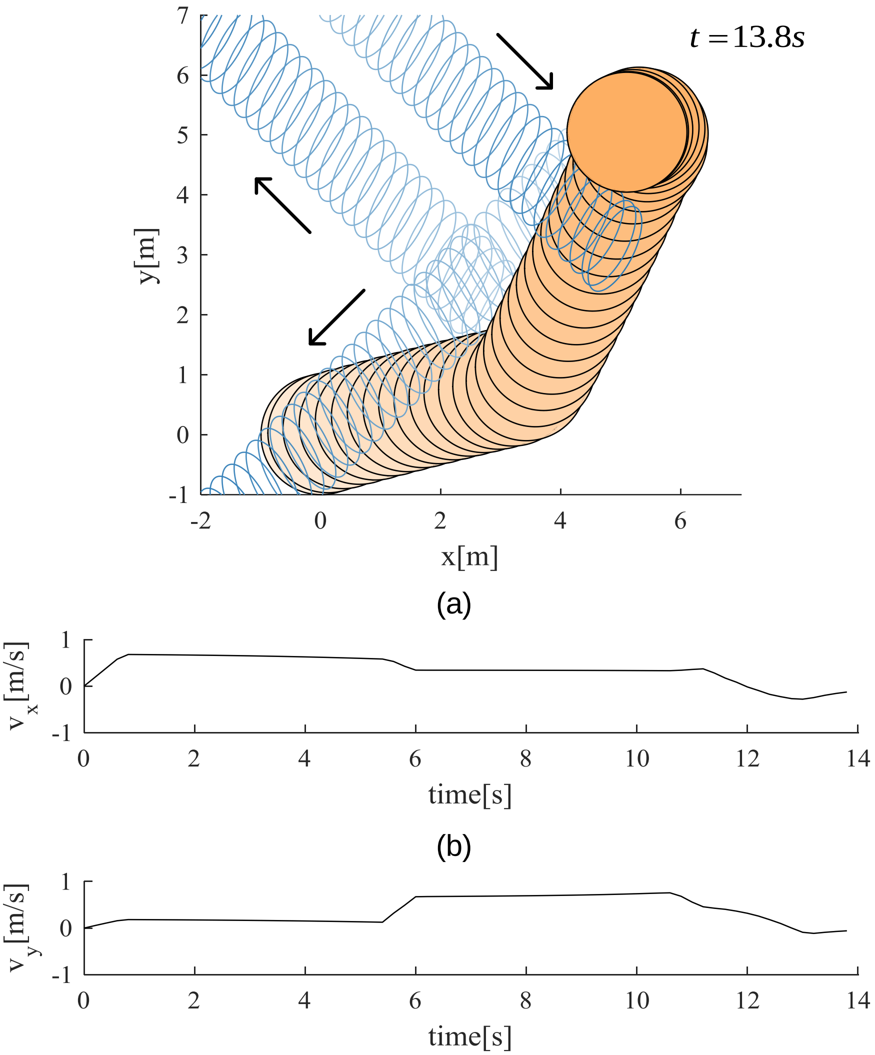


Figure 6.8 Initial states of the robot and the three obstacles in the multiple moving obstacles scenario. The orange ellipse represents the robot $\mathcal{R}$, and the blue circles indicate the obstacles $\mathcal{O}_1$, $\mathcal{O}_2$, and $\mathcal{O}_3$. The goal position $\mathbf{p}_{\mathcal{R}}^{goal}$ is marked as the yellow star, which is the initial position of the obstacle $\mathcal{O}_1$.

$\left[\begin{smallmatrix} 0.20 & 0.20 \\ 0.20 & 0.50 \end{smallmatrix}\right]$. Initially, the obstacle $\mathcal{O}_2$ moved with $\mathbf{v}_{\mathcal{O}_2} = (-0.5\text{m/s}, 0.5\text{m/s})$ at $\mathbf{p}_{\mathcal{O}_2} = (4.0\text{m}, 1.0\text{m})$, and the obstacle $\mathcal{O}_3$ moved with $\mathbf{v}_{\mathcal{O}_3} = (0.5\text{m/s}, -0.5\text{m/s})$ at $\mathbf{p}_{\mathcal{O}_3} = (-2.0\text{m}, 10.0\text{m})$. The above descriptions are summarized in Table 6.3 and presented in Fig. 6.8.

The resultant traces of the robot and the obstacle using the five algorithms are presented in Figs. 6.9–6.13. The overall results were similar to the preceding scenario. It was shown again that the enclosing ellipse was more efficient than the enclosing circle in collision avoidance in Figs. 6.9 and 6.10. Moreover, the

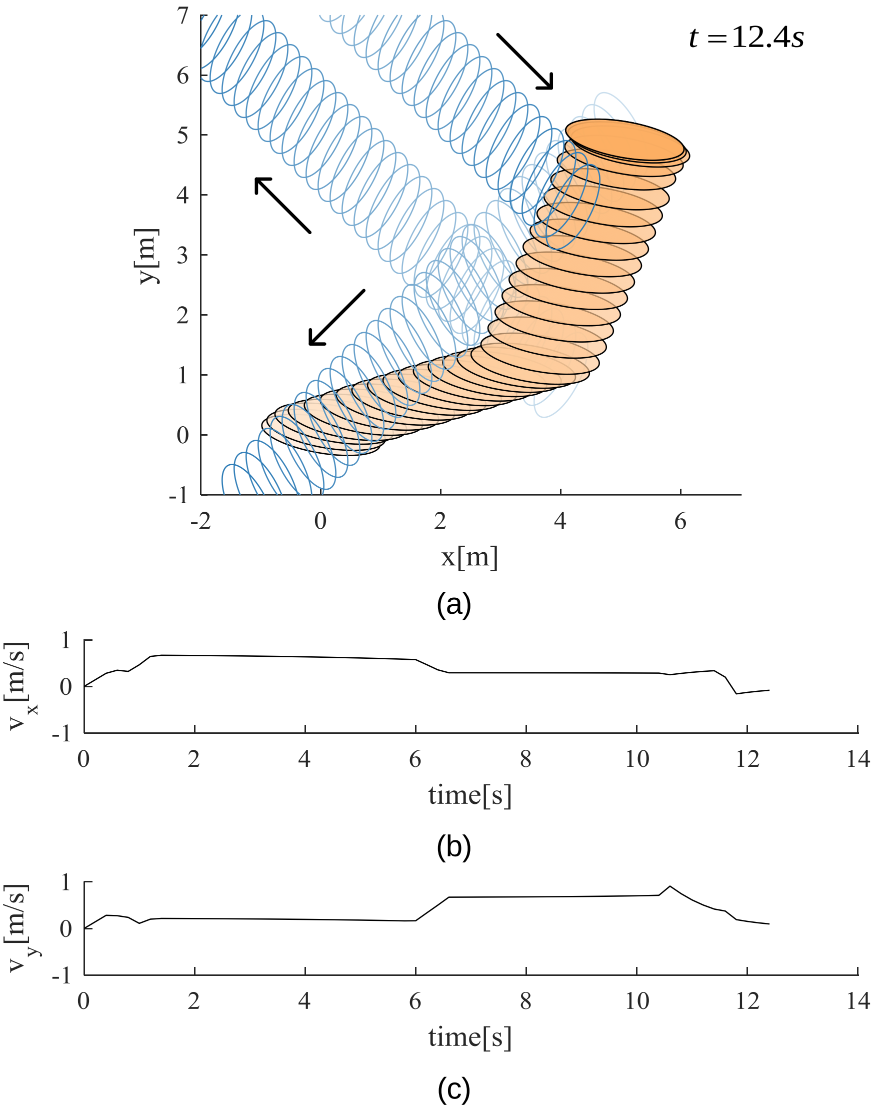


(a)

(b)

(c)

Figure 6.9 The simulation result in the multiple moving obstacles scenario using the *Circle* algorithm: (a) the resultant path; (b)–(c) the robot's linear velocities with respect to the time. The robot reached its goal at $t = 13.8$s.

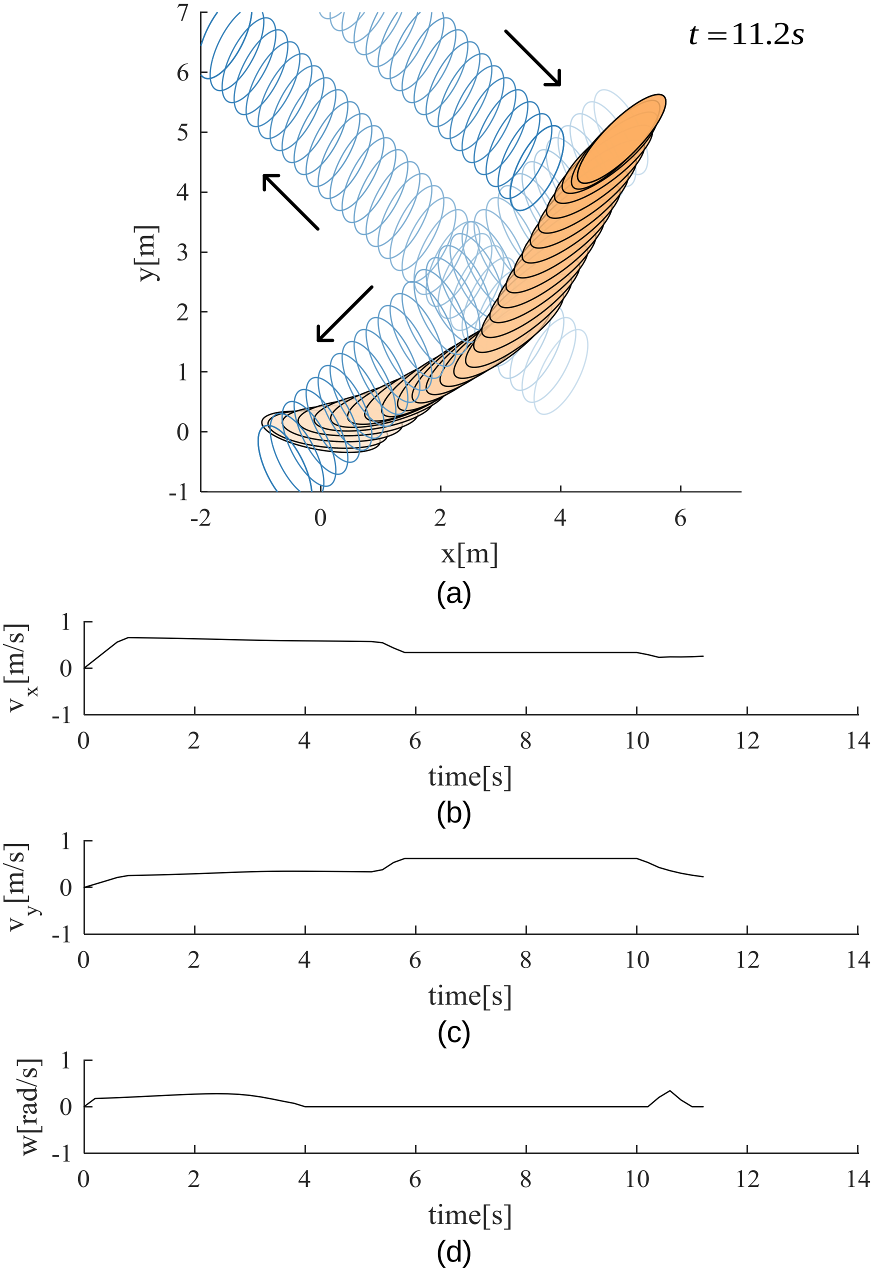


(a)

(b)

(c)

Figure 6.10 The simulation result in the multiple moving obstacles scenario using the *Ellipse-N* algorithm: (a) the resultant path; (b)–(c) the robot's linear velocities with respect to the time. The robot reached its goal at $t = 12.4$s.



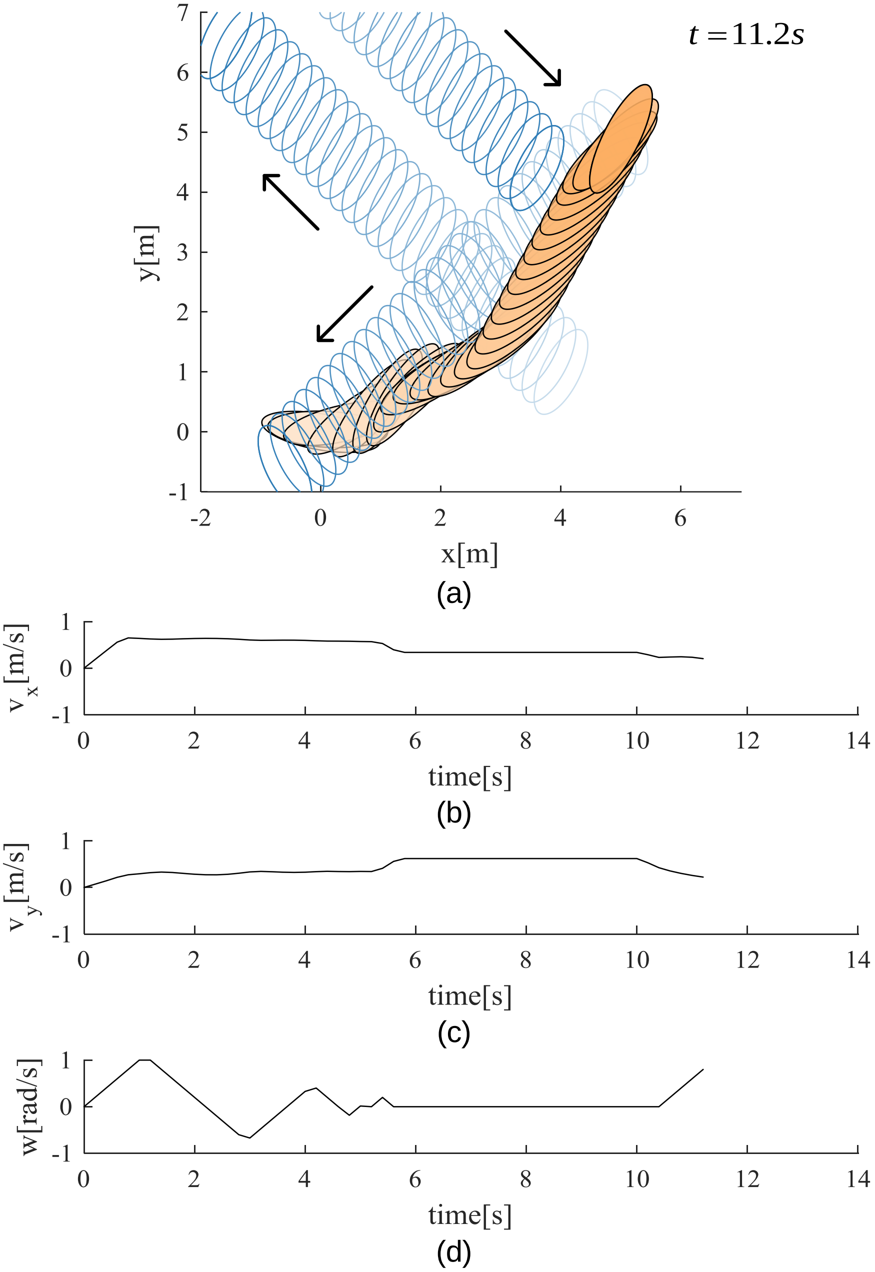
(a)

(b)

(c)

(d)

Figure 6.11 The simulation result in the multiple moving obstacles scenario using the *Ellipse-HR* algorithm: (a) the resultant path; (b)–(d) the robot's linear and angular velocities with respect to the time. The robot reached its goal at $t = 11.2$s.



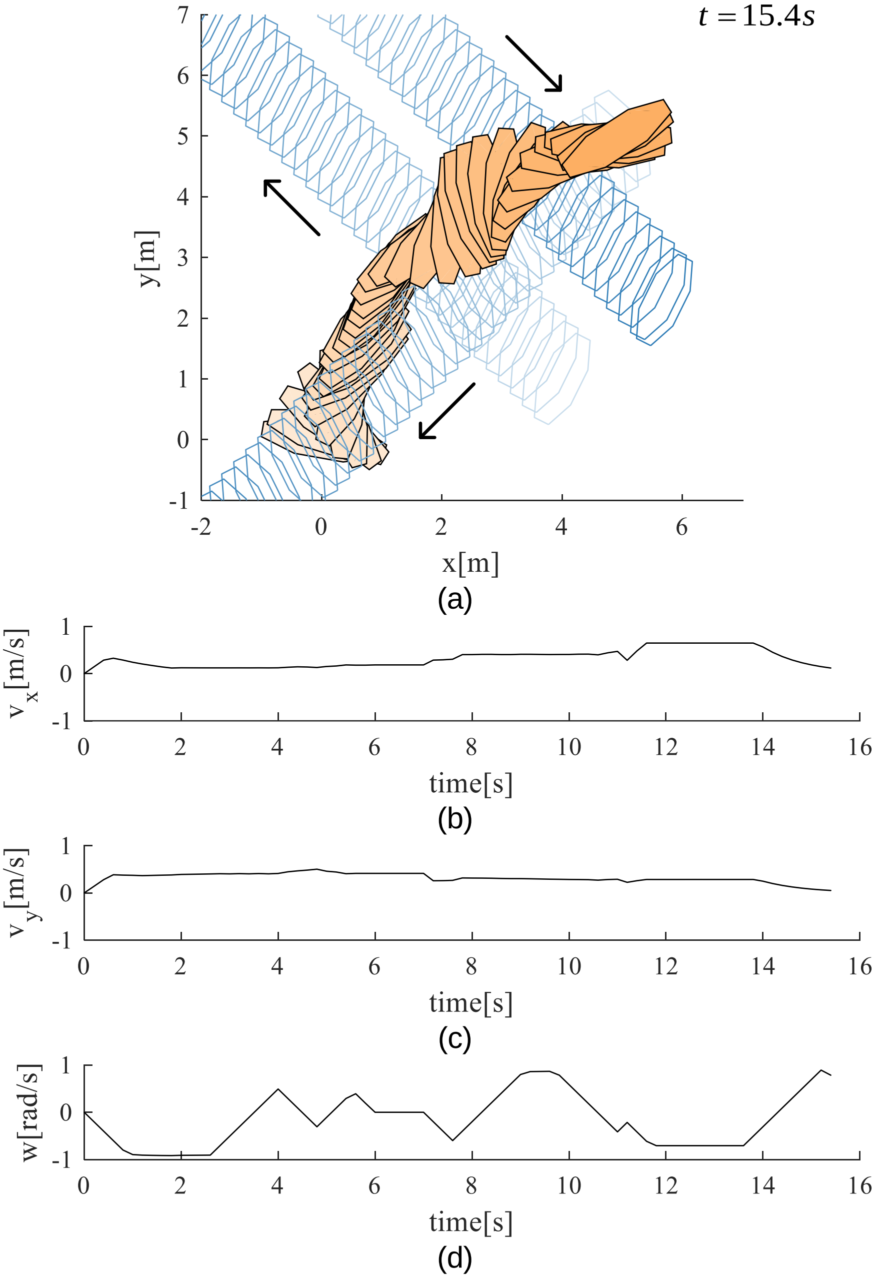
(a)

(b)

(c)

(d)

Figure 6.12 The simulation result in the multiple moving obstacles scenario using the *Ellipse-HO* algorithm: (a) the resultant path; (b)–(d) the robot's linear and angular velocities with respect to the time. The robot reached its goal at $t = 11.2$s.



(a)

(b)

(c)

(d)

Figure 6.13 The simulation result in the multiple moving obstacles scenario using the *Ellipse-HO* algorithm: (a) the resultant path; (b)–(d) the robot's linear and angular velocities with respect to the time. The robot reached its goal at $t = 11.2$s.

robots in Figs. 6.11 and 6.12 moved shorter path than that in Fig. 6.10 due to the rotation. Although the travel times of *Ellipse-H1* and *Ellipse-H2* are equivalent, the proposed algorithm outperformed the method presented in [59] with respect to the travel distance since it struggled to decrease the deviation from the straight path to the goal by changing actively the orientation.

Meanwhile, the characteristic of the motion generated for obstacle avoidance with the *Polygon* algorithm was uncovered in this scenario because the deadlock did not occur contrary to the preceding scenario. First, the robot rotated continuously even when there was no obstacle that blocked the path to its destination. That was because the robot employing the *Polygon* algorithm tried to maximize the distance to obstacles to avoid collisions safely. However, the robot using the proposed algorithm did not rotate in this situation because it rotated to decrease the deviation caused by blocking obstacles. Therefore, the suggested one is more energy-efficient than the *Polygon* algorithm in uncrowded situations. Next, the robot with the *Polygon* algorithm moved slowly when it avoided obstacles compared to those with the others. As a result, the distance traveled by the robot was not long, but it took the longest time to arrive at the destination.

Table 6.4 presents the numerical results of the simulation, and Figure 6.14

Table 6.4 Simulation results in the multiple moving obstacles scenario

Algorithm	Travel distance[m]	Travel time[s]	Computation time[ms]
Circle	8.585	13.8	0.010
Ellipse-N	7.869	12.8	0.031
Ellipse-H1	7.317	11.2	0.033
Ellipse-H2	7.303	11.2	0.080
Polygon	7.414	15.4	2.420

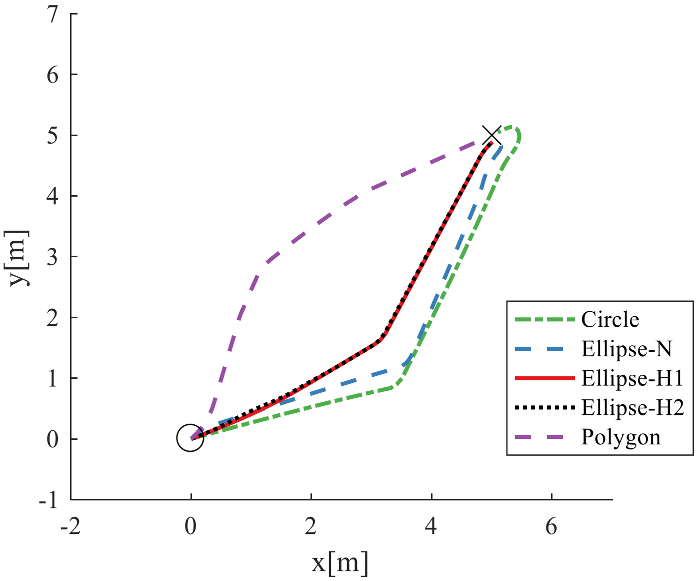


Figure 6.14 The comparison between the paths generated by each of the algorithms in the multiple moving obstacles scenario. The black circle and x-mark are the start and goal positions. The green dash-dot, blue dashed, red solid, black doted, and violet dashed lines represents the paths of *Circle*, *Ellipse-N*, *Ellipse-H1*, *Ellipse-H2*, and *Polygon*, respectively.

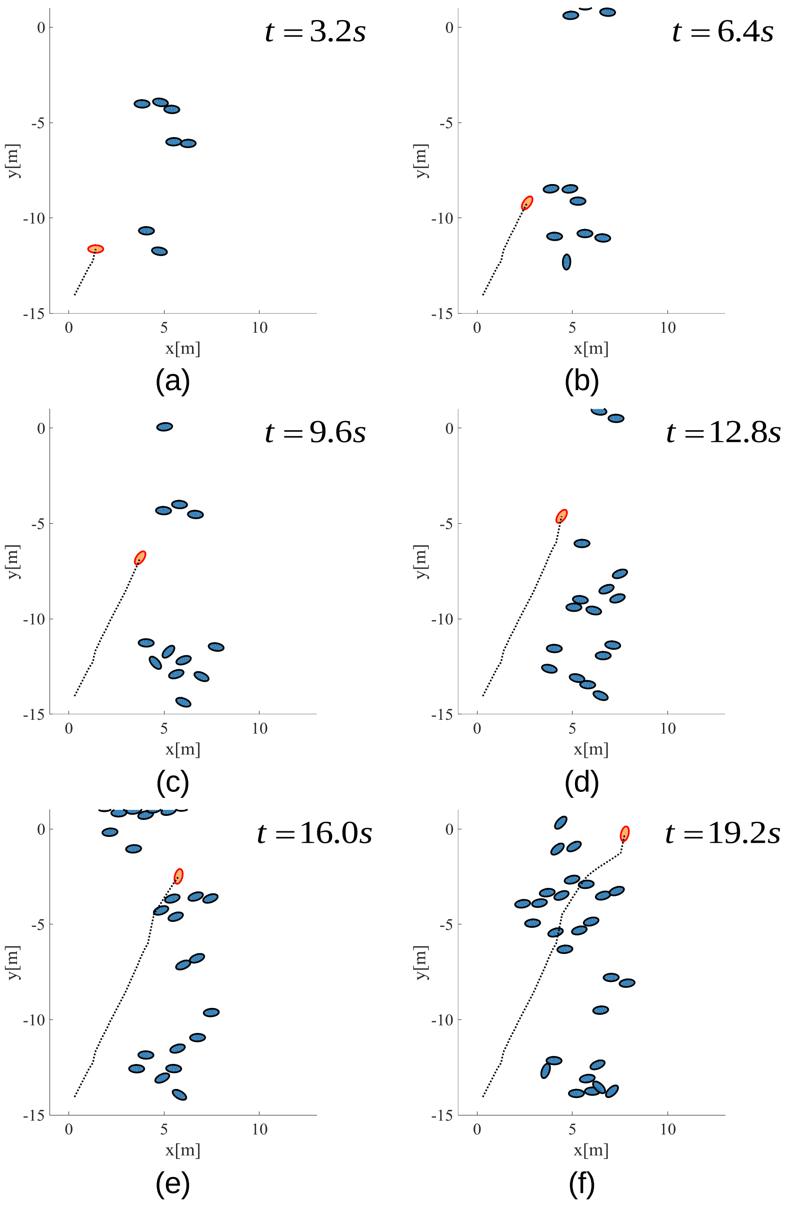
compares the paths for each of the algorithms. The travel distance of the proposed algorithm was 14.93%, 7.20%, 0.19%, and 1.50% shorter than those of *Circle*, *Ellipse-N*, *Ellipse-H1*, and *Polygon*, respectively. The travel times of the proposed algorithm and *Ellipse-H1* were equal, but theirs are 18.84%, 12.50%, and 27.27% shorter than those of *Circle*, *Ellipse-N*, and *Polygon*. Also, all the computation times were still far shorter than the sampling period. However, the computation of the *Polygon* algorithm took at least thirty times longer than the others due to multiple computations of Minkowsiki sum, which causes that it may not be able to operate in real time if there are a number of obstacles in the environment.

6.2.3 Pedestrians avoidance scenario

The proposed algorithm was validated using a real-world dataset. The BIWI Walking Pedestrians dataset was video data of walking students on the campus of ETH Zurich filmed from a bird eye view [86], annotated at 2.5 frames per second. We extracted a sequence about 30 seconds long where the total number of the pedestrians was 53 and the maximum number of them crossing at the same time was 18 and carried out experiments in which a robot avoided the pedestrians in the sequence. Figure 6.15 presents a snapshot of the dataset for this simulation.

In the simulation, each of the pedestrians was represented by an ellipse whose major and minor radii were 40cm and 20cm and major axis was perpendicular to its heading direction. The robot was also supposed to an ellipse having the same size of the pedestrians, but it was given a safe margin of 15cm to compensate the error in assumption that the pedestrians continued their most recent velocity. In addition, the maximum translational and rotational speed limits are assigned to $v_{\mathcal{R}}^{max} = 1.5\text{m/s}$ and $w_{\mathcal{R}}^{max} = 1\text{rad/s}$, but there is no limit

Figure 6.15 A snapshot of the BIWI Walking Pedestrians dataset [86].



(a)

(b)

(c)

(d)

(e)

(f)

Figure 6.16 An example of robot trajectory in the pedestrians avoidance simulation using the *Ellipse-HO* algorithm: (a) $t = 3.2$s; (b) $t = 6.4$s; (c) $t = 9.6$s; (d) $t = 12.8$s; (e) $t = 16.0$s; (f) $t = 19.2$s.

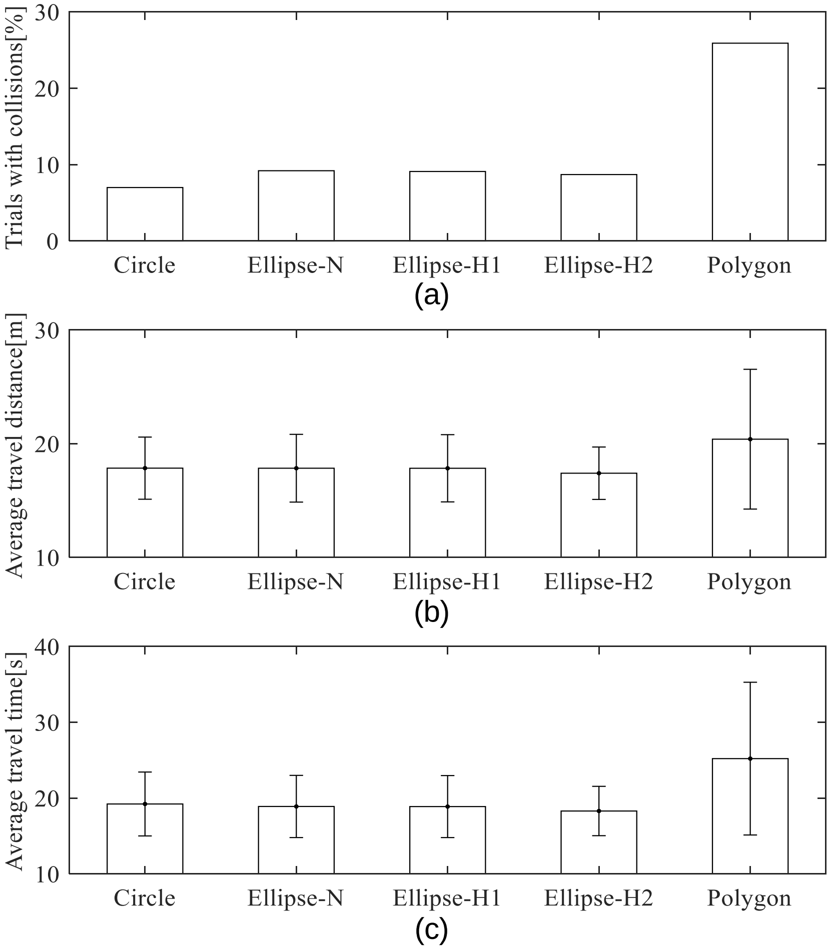


Figure 6.17 Simulation results in the Pedestrians avoidance scenario. The simulation results of the four algorithms for 1,000 runs were compared with respect to (a) the percentage of the trials with collision, (b) the average travel distance, and (c) the average travel time. The distances and times were calculated for the 667 runs where the robot successfully reached the goal with all the algorithms and were represented by bar plots with error bars of one standard deviation.

on the accelerations. The robot was initially located at one side of the walkway and moved to the opposite side while avoiding the pedestrians. The algorithms were tested in 1,000 repeated runs, where the initial and goal position of the robot were randomly chosen. Figure 6.16 shows an example of robot trajectory in the simulation using the proposed algorithm, where the robot with rotation successfully avoided a number of pedestrians.

The simulation results are presented in Fig. 6.17. Figure 6.17(a) indicates the percentage of runs where the robot failed to avoid collisions with the pedestrians. The *Circle* algorithm showed the highest success rate because the robot was most loosely bounded. Among the *Ellipse* variants, the proposed algorithm was more effective than the others in decreasing the collision probability since it exploited the robot's rotation to efficiently avoid the obstacles. However, the highest percentage of trials with collisions, 25.9%, was reported when using the *Polygon* algorithm due to the deadlock induced by the polygonal robot and the low speed at which the robot moved when it avoided obstacles. The longer the robot arrived at its destination, the more obstacles it encountered, increasing the likelihood of collisions.

In Figs. 6.17(b) and (c), the average travel distances and times were calculated for 667 runs where the robot successfully reached the goal using all the algorithms for fairness. As a result, the travel distance of the proposed algorithm was 2.50%, 2.46%, 2.44%, and 14.67% shorter than those of *Circle*, *Ellipse-N*, *Ellipse-H1*, and *Polygon*, respectively. Also, the travel time of the proposed algorithm was 4.83%, 3.17%, 3.10%, and 27.42% shorter than those of the others. Moreover, it is found that the performance of the *Polygon* algorithm was unstable and highly dependent on the environment in that the standard deviation of the average travel distance and time were about 2.5 times larger than the *Circle* and the *Ellipse* variants. On the other hand, the proposed algorithm showed the most stable performance in the simulation.

6.3 Multi-Robot Collision Avoidance

In order to verify the performance of the proposed algorithm for the multi-robot collision avoidance, the following algorithms were compared.

- *Circle* assumed that robots were bounded by circles, which was presented in [104].
- *Ellipse-N* assumed that robots were bounded by ellipses and allowed only translational motion.
- *Ellipse-H* was the proposed algorithm, where robots were bounded by ellipses and were holonomic.
- *Polygon* assumed that robots were bounded by polygons approximating ellipses and were holonomic, which was presented in [45].

Similar to the preceding section, the radius of robots in *Circle* algorithm is denoted by the major radius in the other algorithms, so that $r = 1$m. Also, the ellipses were approximated with eight piecewise lines by using the method proposed in [12] and the granularity parameter in [45] was set to $\delta = 20$ in the *Polygon* algorithm. However, the maximum rotational acceleration was not assigned to all the algorithms in order to make the robots change their orientation quickly according to the surrounding environment. Theses algorithms were implemented and tested in the following two scenarios.

6.3.1 Chicken scenario

Two elliptic robot $\mathcal{R}_1$ and $\mathcal{R}_2$ started at the opposite ends and moved along the same line. At the beginning, the two robots were stalled at $\mathbf{p}_{\mathcal{R}_1} = (6.0\text{m}, 0.0\text{m})$ with $\theta_{\mathcal{R}_1} = 100\,\text{deg}$ and at $\mathbf{p}_{\mathcal{R}_2} = (-6.0\text{m}, 0.0\text{m})$ with $\theta_{\mathcal{R}_2} = -80\,\text{deg}$. The objective was to exchange their positions, so that the goal position of one robot was equal to the initial position of the other robot. The above descriptions are presented in Fig. 6.18 and summarized in Table 6.5.

The resultant trajectories of the robots using the four algorithms are presented in Figs. 6.19–6.22. When comparing the performance of the algorithms of *Circle* and *Ellipse-N*, it is noticed that the former outperformed the latter with respect to both the travel time and distance contrary to expectations. Because

Table 6.5 The initial parameters of two robots in the chicken scenario

	Position[m]	Orientation[°]	Destination[m]
Robot $\mathcal{R}_1$	$(\ \ 6.0, 0.0)$	100	$(-6.0, 0.0)$
Robot $\mathcal{R}_2$	$(-6.0, 0.0)$	-80	$(\ \ 6.0, 0.0)$

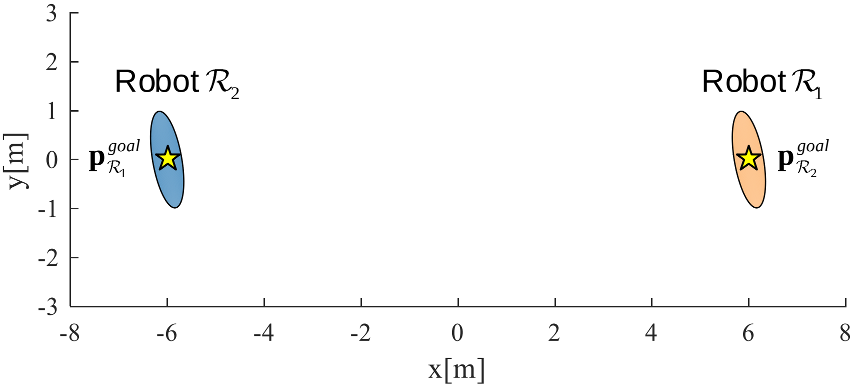


Figure 6.18 The initial and goal positions of two robots in the chicken scenario.

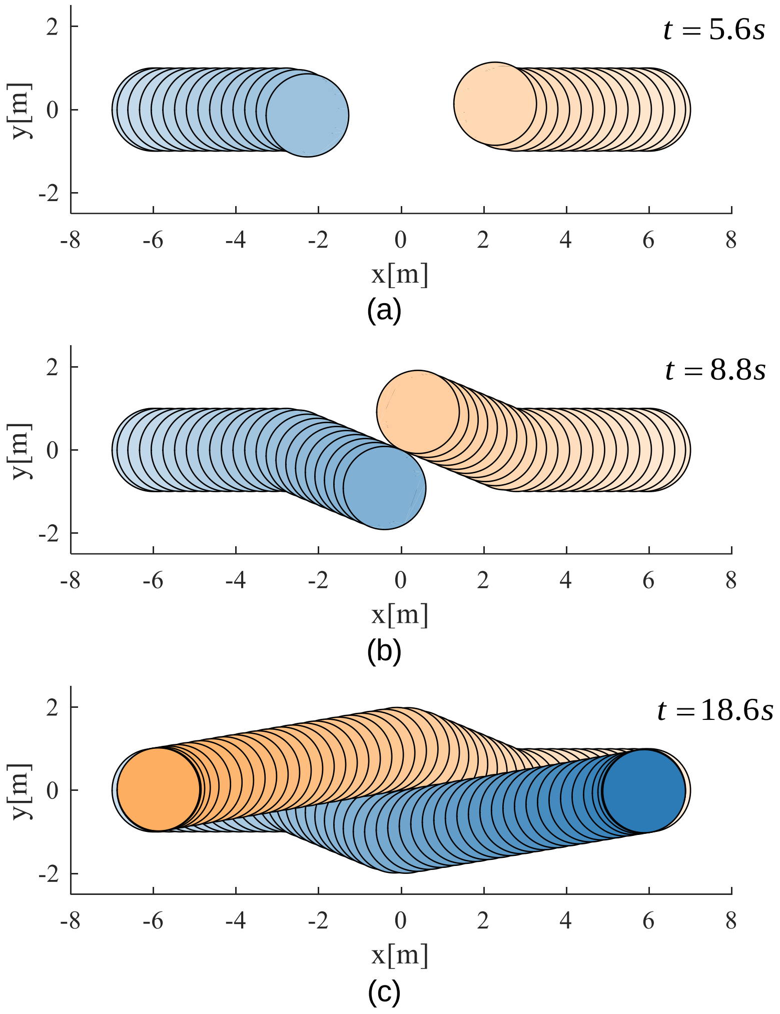
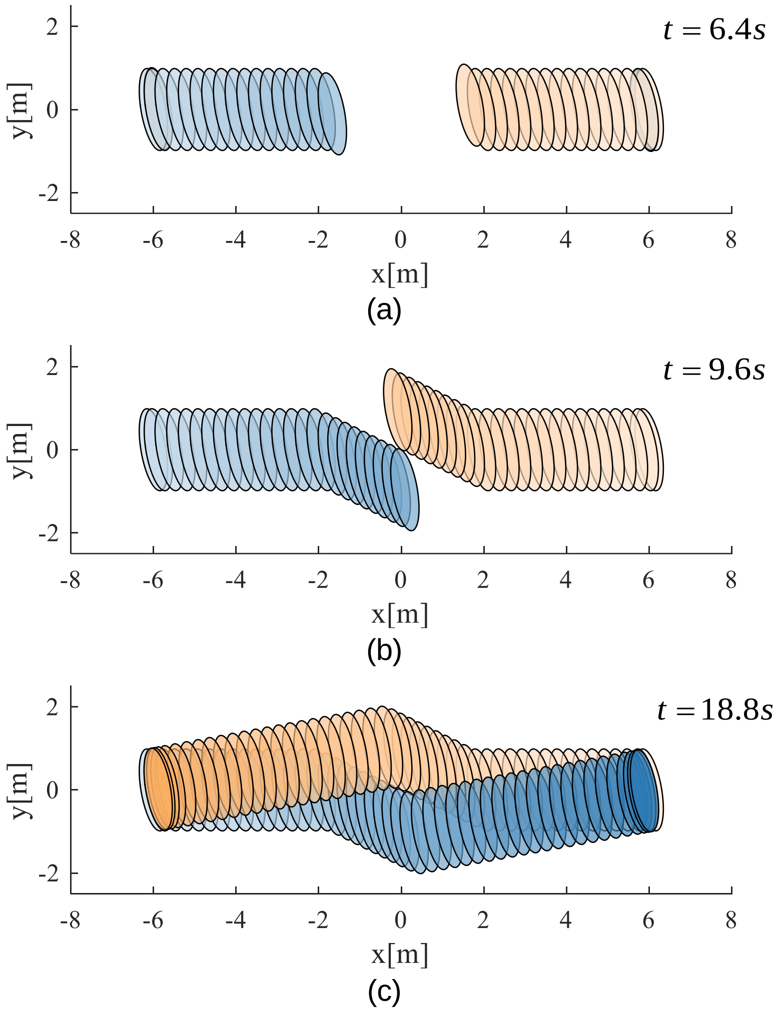


Figure 6.19 The simulation result in the chicken scenario using the *Circle* algorithm: (a) the two robots started to have a velocity component in the y-direction at $t = 5.6$s; (b) they passed by each other in the middle at $t = 8.8$s; (c) they reached their goals at $t = 18.6$s.

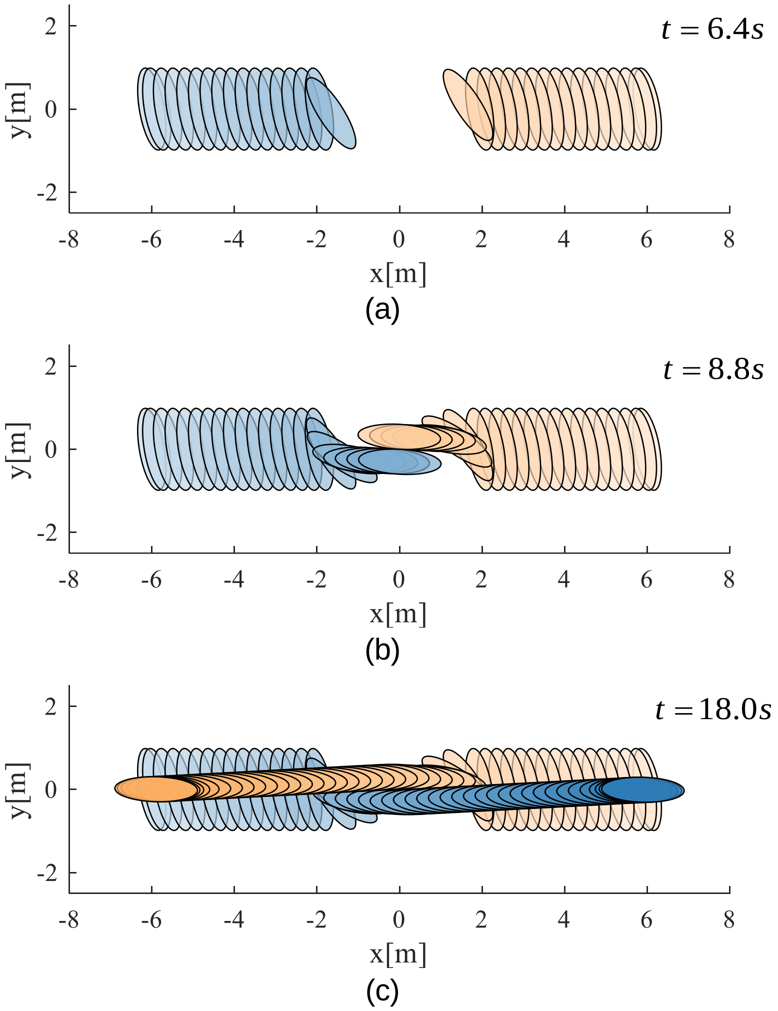


(a)

(b)

(c)

Figure 6.20 The simulation result in the chicken scenario using the *Ellipse-N* algorithm: (a) the two robots started to have a velocity component in the y-direction at $t = 6.4$s; (b) they passed by each other in the middle at $t = 9.6$s; (c) they reached their goals at $t = 18.8$s.

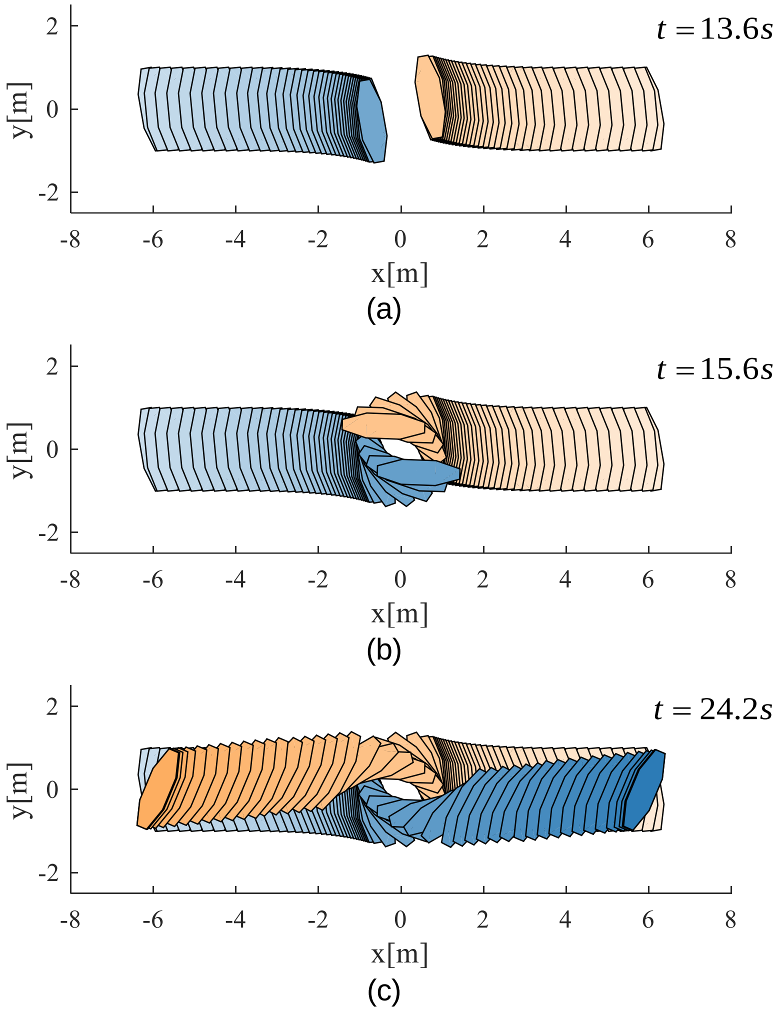


(a)

(b)

(c)

Figure 6.21 The simulation result in the chicken scenario using the *Ellipse-H* algorithm: (a) the two robots started to rotate at $t = 6.4$s; (b) they passed by each other in the middle at $t = 8.8$s; (c) they reached their goals at $t = 18.0$s.



(a)

(b)

(c)

Figure 6.22 The simulation result in the chicken scenario using the *Polygon* algorithm: (a) the two robots started to rotate at $t = 13.6$s; (b) they passed by each other in the middle at $t = 15.6$s; (c) they reached their goals at $t = 24.2$s.

the elliptic robots were enclosed by the circular robots, the *Ellipse-N* algorithm was predicted to show better performance than the *Circle* one. However, the distance between the robots in the former was larger than the latter, the elliptic robots started to take an avoidance maneuver later than the circular robots due to the time horizon τ. Actually, the elliptic robots started to have a velocity component in the y-direction at $t = 6.4$s, while the circular robots did at $t = 5.6$s, as shown in Figs. 6.19(a) and 6.20(a). The robots using the *Ellipse-H* algorithm also started to rotate at $t = 6.4$s, as presented in Fig. 6.21(a).

In addition, it was shown the proposed algorithm decreased the travel time and distance by efficiently adjusting their orientation. By aligning their orientation with the direction of the relative velocity, they could avoid each other as if they were circular robots with radius equal to their minor radius. Compared to the other approaches, Figure 6.21(c) shows that the robots utilized the only small area in workspace to avoid each other in the proposed algorithm, so that they were expected to pass safely in even narrow passages.

Although the *Polygon* algorithm also decreased the travel distance compared to other algorithms that assumed the robots did not rotate, but it took the longest time for the robots to reach their destination. That was because the robots had encountered the deadlock until they started to rotate at $t = 13.6$s. The most significant difference between the proposed method and the *Polygon* algorithm is that a robot with the former rotated quickly when it could avoid other robots more efficiently through rotation, while that with the latter rotated only after it fell a deadlock situation. In addition, the computation time of the *Polygon* algorithm was much longer than that of the proposed one since it required Minkowski sum to be computed multiple times.

Table 6.6 Simulation results in the chicken scenario

Algorithm	Travel distance[m]	Travel time[s]	Computation time[ms]
Circle	12.152	18.6	0.001
Ellipse-N	12.208	18.8	0.018
Ellipse-H	11.941	18.0	0.216
Polygon	12.031	24.0	21.250

Table 6.6 presents the numerical results of the simulation. The average travel distance per robot of the proposed algorithm was 1.74%, 2.19%, and 0.75% shorter than those of *Circle*, *Ellipse-N*, and *Polygon*. In addition, the robots utilizing the *Ellipse-H* algorithm reached their goal 3.23%, 4.26%, and 25.00% faster than those employing the others. Finally, although the *Ellipse-H* algorithm took about 20 and 200 times longer than the *Circle* and *Ellipse-N* algorithms, it was still insignificant because it was still far shorter than the sampling period $\Delta t = 0.2$s. In addition, it was about 100 times faster than that of the *Polygon* method.

6.3.2 Circle scenario

Nineteen elliptic robot $\mathcal{R}_i$ for $i = 1, \cdots, 19$ were distributed equally on a circle with radius of 15.0m, and their goal was to reach their opposite position on the circle, as shown in Fig. 6.23. In order to avoid the perfect symmetry, the orientations of the robots were not assigned as perpendicular or parallel to their direction to the goal. The initial positions, orientations, and goal positions of the robots are summarized in Table 6.7.

The resultant trajectories of the robots using the four algorithms are presented in Figs. 6.24–6.27. In the circle scenario, congestion occurred in the

Table 6.7 The initial parameters of the robots in the circle scenario

	Position[m]	Orientation[°]	Destination[m]
Robot $\mathcal{R}_1$	(15.00, 0.00)	95.00	(−15.00, 0.00)
Robot $\mathcal{R}_2$	(14.19, 4.87)	113.95	(−14.19, −4.87)
Robot $\mathcal{R}_3$	(11.84, 9.21)	132.89	(−11.84, −9.21)
Robot $\mathcal{R}_4$	(8.20, 12.56)	151.84	(−8.20, −12.56)
Robot $\mathcal{R}_5$	(3.68, 14.54)	170.79	(−3.68, −14.54)
Robot $\mathcal{R}_6$	(−1.24, 14.95)	−170.26	(1.24, −14.95)
Robot $\mathcal{R}_7$	(−6.03, 13.74)	−151.32	(6.03, −13.74)
Robot $\mathcal{R}_8$	(−10.16, 11.04)	−132.37	(10.16, −11.04)
Robot $\mathcal{R}_9$	(−13.19, 7.14)	−113.42	(13.19, −7.14)
Robot $\mathcal{R}_{10}$	(−14.80, 2.47)	−94.47	(14.80, −2.47)
Robot $\mathcal{R}_{11}$	(−14.80, −2.47)	−75.53	(−14.80, 2.47)
Robot $\mathcal{R}_{12}$	(−13.19, −7.14)	−56.58	(13.19, 7.14)
Robot $\mathcal{R}_{13}$	(−10.16, −11.04)	−37.63	(10.16, 11.04)
Robot $\mathcal{R}_{14}$	(−6.03, −13.74)	−18.68	(6.03, 13.74)
Robot $\mathcal{R}_{15}$	(−1.24, −14.95)	0.26	(1.24, 14.95)
Robot $\mathcal{R}_{16}$	(3.68, −14.54)	19.21	(−3.68, 14.54)
Robot $\mathcal{R}_{17}$	(8.20, −12.56)	38.16	(−8.20, 12.56)
Robot $\mathcal{R}_{18}$	(11.84, −9.21)	57.11	(−11.84, 9.21)
Robot $\mathcal{R}_{19}$	(14.19, −4.87)	76.05	(−14.19, 4.87)

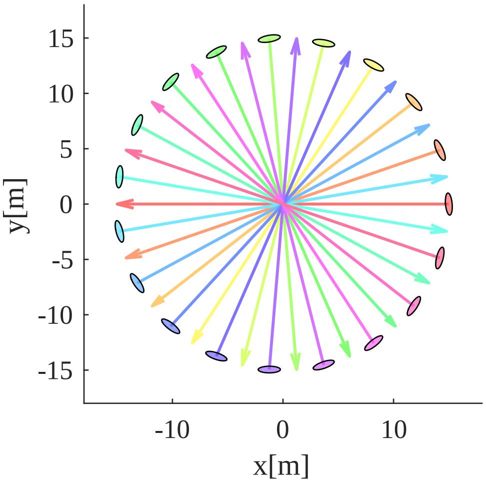
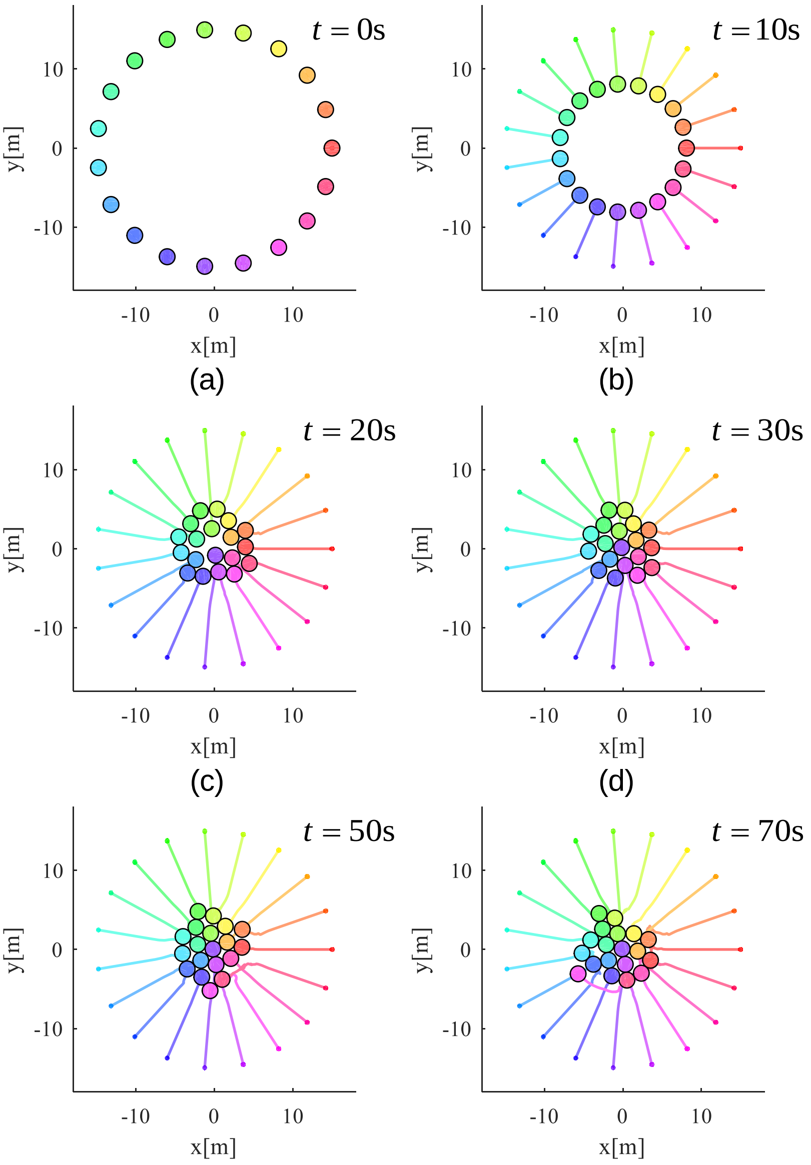


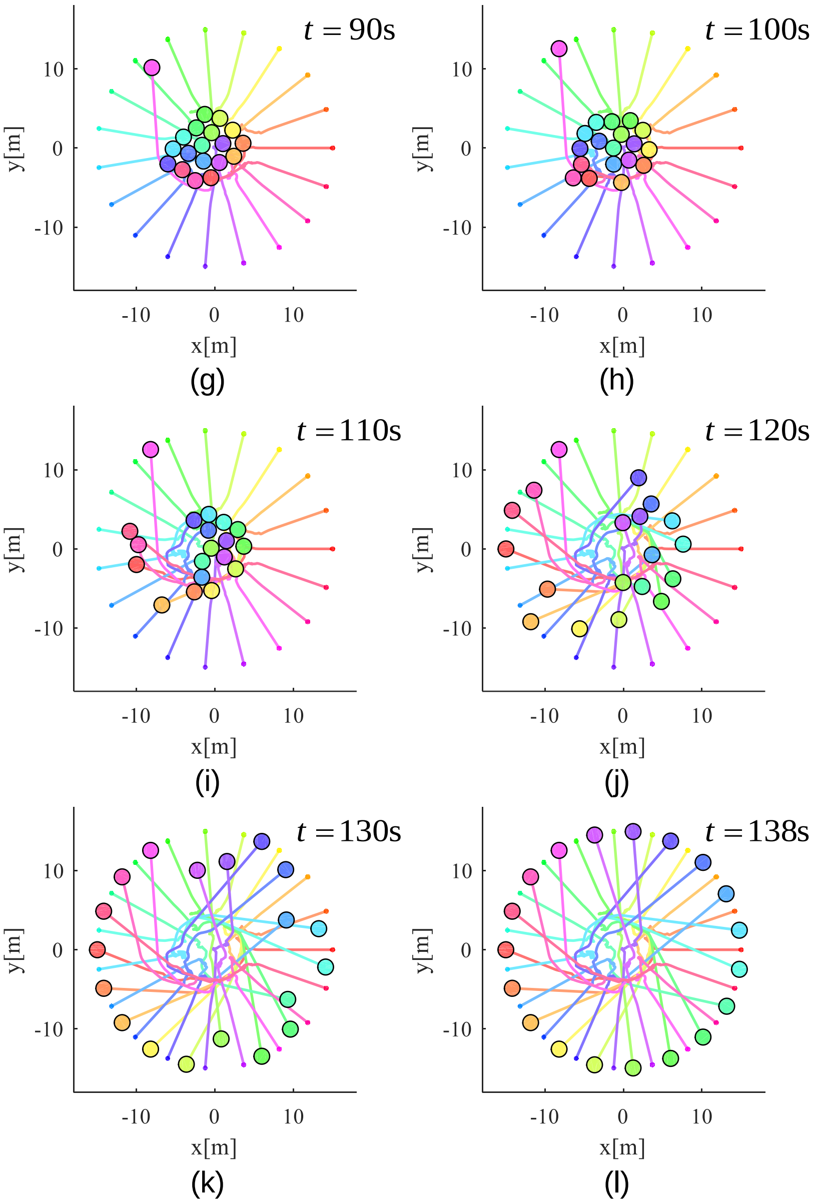
Figure 6.23 The initial positions of 19 robots and their straight path to their goal in the circle scenario. The ellipses represent the robots, and the arrows indicate the straight paths to their goal.

center because all the robots gathered at the origin to reach the destination. The bigger the size of the robots was, the more often the congestion occurred. That was because the free space that the robots were allowed to move without collisions was reduced if their size increased. For instance, congestion lasted for 90s when using the *Circle* algorithm, while it lasted for 50s when the *Ellipse-N* method was utilized, as shown in Figs. 6.24 and 6.25. As a result, the robots with the *Ellipse-N* could arrive at their destination about 40 seconds earlier than those with the *Circle*.

However, the possibility of the congestion can be reduced through the rotation of the robots. As described in Section 4.7.1, the robots can avoid other robots or obstacles as if smaller ones avert collisions by changing their orientation. The *Ellipse-H* and *Polygon* algorithms were tested to achieve this goal, and the results are presented in Figs. 6.26 and 6.27.



(a) (b) (c) (d) (e) (f)



(g)

(h)

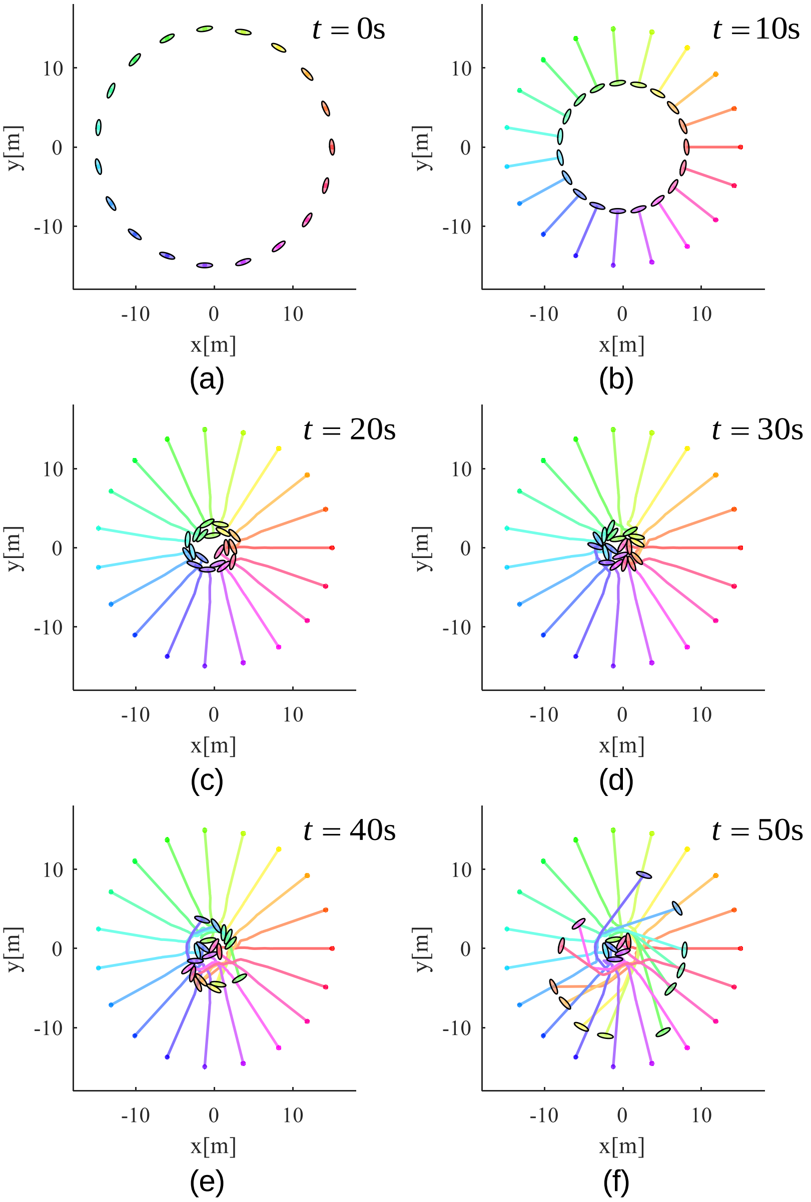
(i)

(j)

(k)

(l)

Figure 6.24 The trajectories of the robots in the circle scenario using *Circle* algorithm: (a) $t = 0$s; (b) $t = 10$s; (c) $t = 20$s; (d) $t = 30$s; (e) $t = 50$s; (f) $t = 70$s; (g) $t = 80$s; (h) $t = 100$s; (i) $t = 110$s; (j) $t = 120$s; (k) $t = 130$s; (l) $t = 138$s. There was congestion near the origin from $t = 20$s to $t = 110$s, but eventually all the robots arrived at their goal at $t = 138$s.



(a) (b) (c) (d) (e) (f)

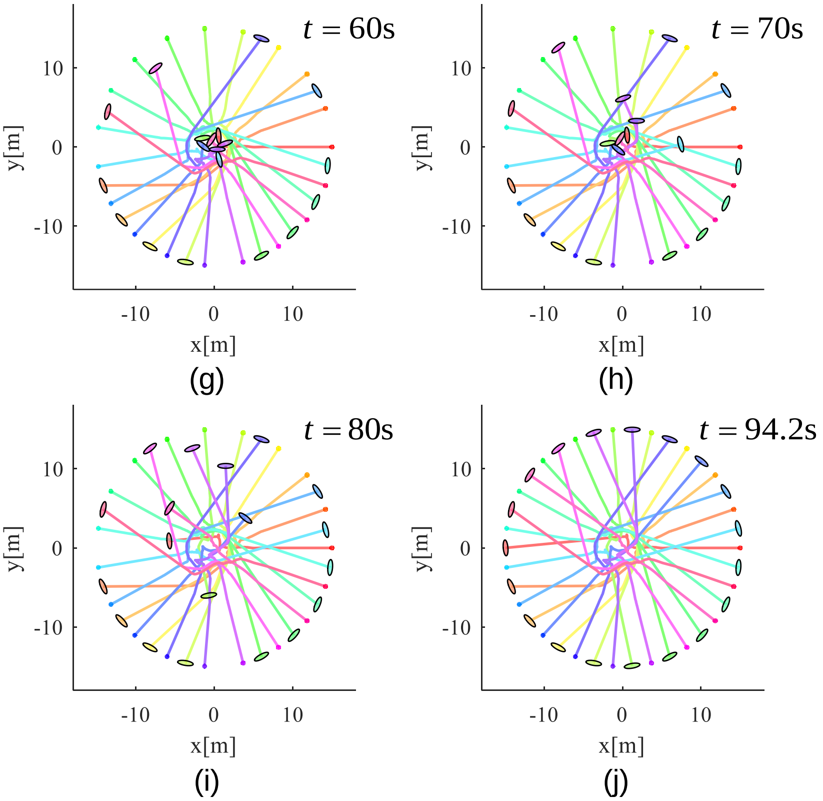


Figure 6.25 The trajectories of the robots in the circle scenario using *Ellipse-N* algorithm: (a) $t = 0$s; (b) $t = 10$s; (c) $t = 20$s; (d) $t = 30$s; (e) $t = 40$s; (f) $t = 50$s; (g) $t = 60$s; (h) $t = 70$s; (i) $t = 80$s; (j) $t = 94.2$s. There was congestion near the origin from $t = 20$s to $t = 70$s, but eventually all the robots arrived at their goal at $t = 94.2$s.

When using the proposed algorithm, the robots rotated from $t = 10$s and $t = 20$s to adjust their orientation suitable for avoiding collisions. As a result, there was no congestion in the center and the robots arrived at the goal the fastest among the four algorithms. They moved at an average speed of 0.643m/s, which corresponded to about 91% of the preferred speed. he *Polygon* algorithm also decreased the average travel time and distance. Nevertheless, since the robots with the *Polygon* method moved slowly compared to those with the

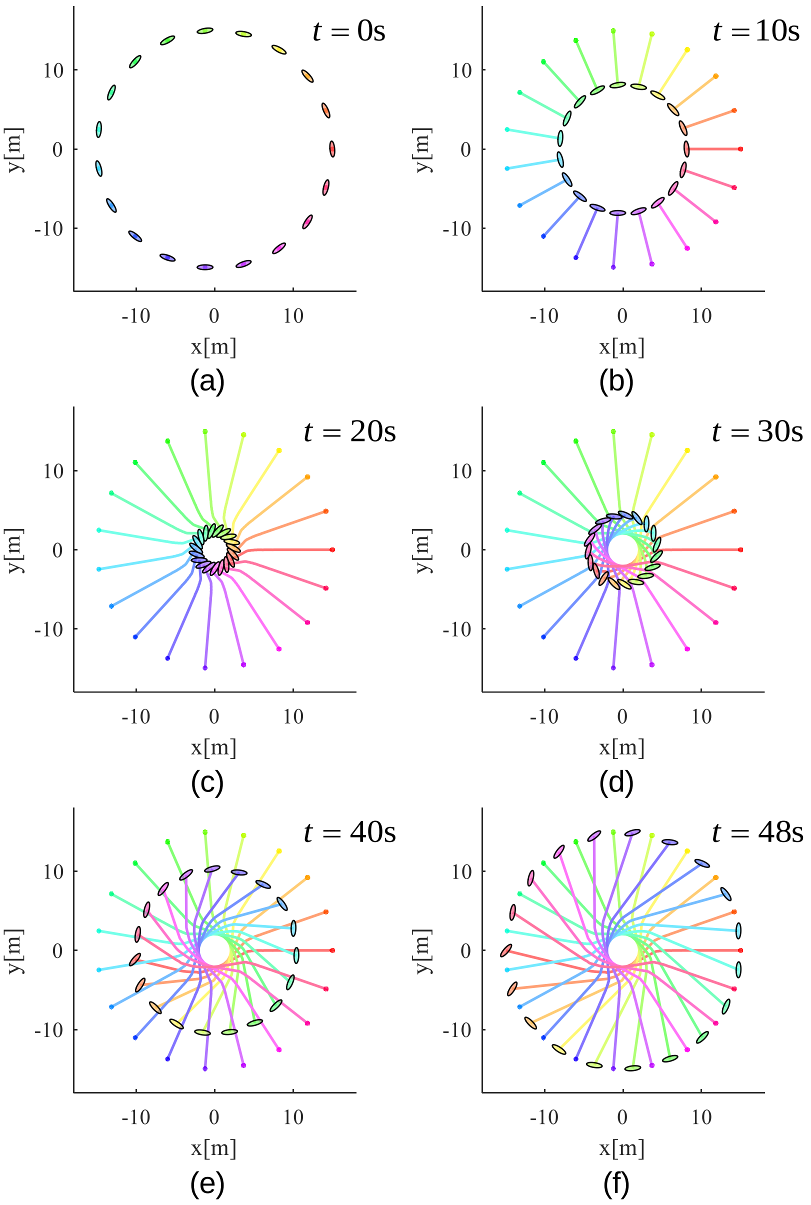


Figure 6.26 The trajectories of the robots in the circle scenario using *Ellipse-H* algorithm:: (a) t = 0s; (b) t = 10s; (c) t = 20s; (d) t = 30s; (e) t = 40s; (f) t = 48s. There was no congestion near the origin because all the robots changed their orientation adequately, so that they arrived at their goal at t = 48s.

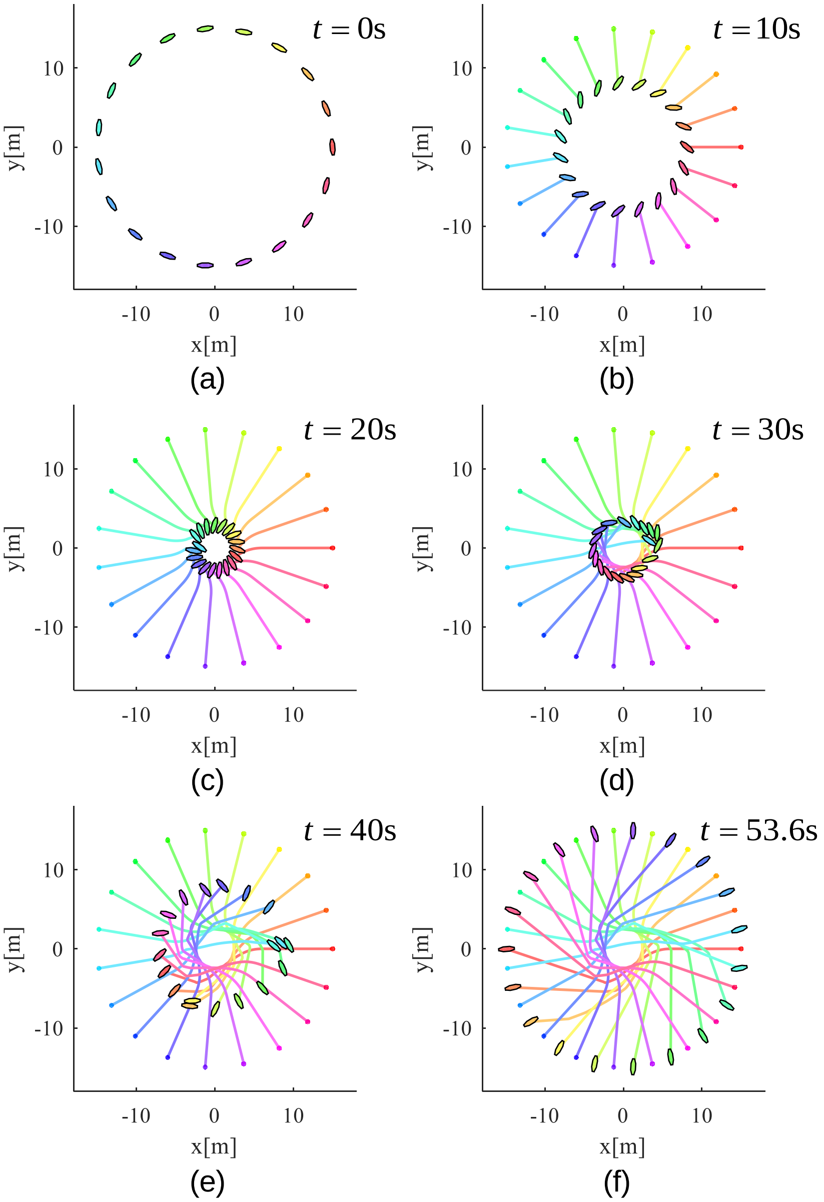


Figure 6.27 The trajectories of the robots in the circle scenario using *Polygon* algorithm:: (a) $t = 0$s; (b) $t = 10$s; (c) $t = 20$s; (d) $t = 30$s; (e) $t = 40$s; (f) $t = 53.6$s. There was no congestion near the origin because all the robots changed their orientation adequately, so that they arrived at their goal at $t = 53.6$s.

Table 6.8 Simulation results in the circle scenario

Algorithm	Travel distance[m]	Travel time[s]	Computation time[ms]
Circle	36.566	138.0	0.125
Ellipse-N	31.911	94.2	0.401
Ellipse-H	30.862	48.0	2.165
Polygon	31.883	53.6	155.981

others when they had to avoid collisions, the travel time was reported 5.6s later than that of the proposed algorithm. They moved at an average speed of 0.595m/s, which corresponded to about 84% of the preferred speed.

Furthermore, the numerical results of the circle scenario are summarized in Table 6.8. The average travel distance per robot of the proposed algorithm was 15.60%, 3.29%, and 3.20% shorter than those of the *Circle*, *Ellipse-N*, *Polygon* algorithms. The travel time of the *Ellipse-H* algorithm was also 65.22%, 49.04%, and 10.45% less than those of the others. Because the computation times of the *Circle* algorithm and the *Ellipse* variants were far shorter than the sampling period, less than 3%, the algorithms utilized just a little part of the computational resources. On the other hand, the computation time of the *Polygon* algorithm accounted for about three fourths of the sampling time. It meant that the robots could not perform more complicated tasks than just moving, which was not desirable.

Chapter 7

Conclusion

This dissertation has proposed a velocity-based algorithm for local collision avoidance of elliptic robots in three steps. First, the collision detection between two ellipses was investigated. Next, the velocity-based local collision avoidance method for an elliptic robot in dynamic environments with moving elliptic obstacles was suggested. Finally, this method was extended for multi-robot collision avoidance.

In the literature on the velocity-based local navigation, sets of velocities of a robot that induce collisions were represented in the velocity space, and it reached its goal position without collision by selecting its velocity outside of these regions. In order to simplify the computation of the forbidden velocities, robots and obstacles used to be approximated by simple geometric objects. The minimum area bounding ellipse, known as the Löwner ellipse [62], is a good candidate since it can enclose them more tightly than the bounding circle and it requires a smaller amount of memory than the convex hull to store a computer model. However, the collision detection between two ellipses has been known to be computationally expensive.

Therefore, we investigated two methods for the collision detection between two ellipses in Chapter 3, which were complementary to each other. The first method was to utilize the configuration space framework [77]. The configuration space obstacle (C-obstacle) of an elliptic obstacle with respect to an elliptic robot was defined, and its boundary was identified. In addition, it was shown that an elliptic robot could be regarded as a circular robot with radius equal to its minor radius by changing its orientation. Nevertheless, it was hard to figure out whether a point robot was contained in the C-obstacle or not because the locus of the boundary was presented in the form of a parametric equation. In order to tackle this problem, an algebraic condition for the interior-disjoint of two ellipses was derived by following the proofs in [23] and [36]. It was proven that this condition was obtained from the separation condition by replacing three of the four strict inequalities to their corresponding non-strict ones. This result could facilitate the collision avoidance of an elliptic robot by representing the position and orientation as a function of time.

Next, we presented a velocity-based obstacle avoidance algorithm for an elliptic robot in Chapter 4, where the robot changed its orientation to increase the efficiency of the collision avoidance. This method was composed into two parts: linear and angular motion planning. In the first part, the ellipse-based velocity obstacle (EBVO) that was a set of linear velocities of an elliptic robot that would induce a collsion with obstacles within a finite horizon was derived. Also, the conservative approximation of the EBVO was suggested for the computational efficiency. Afterward, a strategy for selecting the new linear velocity closest to the preferred linear velocity and outside the approximated EBVO was presented. In the second part, the problem that determined the new angular velocity was considered. If the new linear velocity was selected to avoid obstacles that blocked the shortest path to the destination, there was an obstacle that the robot grazed when it maintained the chosen velocity. If the robot efficiently avoided the first contact obstacle, it could detour obstacles with the minimum deviation from the straight path to the goal. Hence, the time to contact was calculated when the robot touched the first contact obstacle, and the collision-free rotation angles were calculated within the time to collision. Next, the preferred angular velocities were calculated that enabled the robot to detour the first contact obstacle efficiently. Finally, a strategy for selecting the new angular velocity, which was similar with that of the first part, was presented.

Lastly, we extended the obstacle avoidance method for multi-robot collision avoidance. Contrary to passive obstacles, the motion of robots was difficult to expect because they continuously sensed their surrounding environment and replaned their action based on the input. Hence, we assumed that all the robots in the environment used the same collision avoidance algorithm, so that one robot could predict the next velocities of other robots. Because the predictive motion of other robots were different with obstacles, the condition of collisions changed in both linear and angular motion planning. In order to account for reciprocity between robots in linear motion planning, the concept of hybrid reciprocal velocity obstacles (HRVO) was adopted. Also, the collision-free reciprocal rotation angles were calculated on the assumption that if one robot rotated, then the other robot might rotate equally or equally opposite in the angular motion planning.

Both of the collision avoidance methods were evaluated and compared with other algorithms in simulations for various scenarios in Chapter 6. The obstacle avoidance was tested for three scenarios, increasing the number of obstacles from 1 to 53. Especially in the last scenario, a real pedestrian dataset was utilized and 1,000 trials were conducted for randomly chosen initial and goal positions of the robot. The multi-robot collision avoidance was experimented through chicken and circular scenarios. In the chicken scenario, there were two robots starting at the opposite ends and moved along the same line to analyze characteristics of the motion of the robots. On the other hand, in the circle scenario, there were 19 robots distributed equally on a circle and moving to their opposite position in order to see if they could avoid congestion efficiently. As a result, the proposed algorithm decreased the travel time and distance through the additional computation when compared with the conventional methods that approximated robots and obstacles with circles, with ellipses without rotation. Moreover, the proposed method had a computation time that was dozens of times faster than the algorithm modeling robots as rotating polygons, with the better results in terms of the travel time and distance.

The future work of this dissertation lies in three aspects. First, the proposed algorithm for multi-robot collision avoidance can be extended to those based on the optimal reciprocal collision avoidance (ORCA) [116]. Since the HRVO generally has a fundamental limitation that it does not provide a sufficient condition for collision avoidance, increasing the number of robots can lead to defects in the system.. Moreover, the expected relative velocity used to calculate the collision-free reciprocal rotation angles may be invalid in that case. Hence, the proposed algorithm needs to be established on the ORCA for the robustness.

Next, the proposed algorithm was suggested for holonomic elliptic robots. Therefore, it is planned to study the kinematic constraints of non-holonomic robots to expand the application area of the proposed algorithm. In a similar way to the method studied in [9], the movement of non-holonomic robots should be calculated numerically through a Runge-Kutta integration because the velocity of robots changes with time.

Finally, it is necessary to conduct research about predicting human motion trajectories so that robots navigate around humans without collisions. In order to estimate a person's movement exactly, human-robot interaction must be considered. For instance, if a person does not see a robot, the person keeps the direction in which he or she has moved. Otherwise, he or she tries to avoid a collision with the robot. Therefore, the proposed algorithm, which assumed that dynamic obstacles will continue to maintain their own velocity, should be extended for human avoidance based on the human-robot interaction.

Bibliography

[1] Y. Abe and Y. Matsuo, "Collision avoidance method for multiple autonomous mobile agents by implicit cooperation," in *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, Maui, Hawaii, USA, 2001, pp. 1207–1212.

[2] J. Alonso-Mora, A. Breitenmoser, P. Beardsley, and R. Siegwart, "Reciprocal collision avoidance for multiple car-like robots," in *Proceedings of IEEE International Conference on Robotics and Automation*, Saint Paul, MN, USA, 2012, pp. 360–366.

[3] J. Alonso-Mora, A. Breitenmoser, M. Rufli, P. Beardsley, and R. Siegwart, "Optimal reciprocal collision avoidance for multiple non-holonomic robots," in *Distributed Autonomous Robotic Systems*, ser. Springer Tracts in Advanced Robotics, 2012, vol. 83, pp. 203–216.

[4] J. Alonso-Mora, A. Breitenmoser, M. Rufli, R. Siegwart, and P. Beardsley, "Multi-robot system for artistic pattern formation," in *Proceedings of IEEE International Conference on Robotics and Automation*, Shanghai, China, 2011, pp. 4512–4517.

[5] ——, “Image and animation display with multiple mobile robots,” *International Journal of Robotics Research*, vol. 31, no. 6, pp. 753–773, 2012.

[6] J. Alonso-Mora, M. Schoch, A. Breitenmoser, R. Siegwart, and P. Beardsley, “Object and animation display with multiple aerial vehicles,” in *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, Vilamoura, Algarve, Portugal, 2012, pp. 1078–1083.

[7] J. Alonso-Mora, T. Naegeli, R. Siegwart, and P. Beardsley, “Collision avoidance for aerial vehicles in multi-agent scenarios,” *Autonomous Robots*, vol. 39, no. 1, pp. 101–121, 2015.

[8] D. Bareiss and J. van den Berg, “Reciprocal collision avoidance for robots with linear dynamics using LQR-Obstacles,” in *Proceedings of IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, 2013, pp. 3847–3853.

[9] ——, “Generalized reciprocal collision avoidance,” *International Journal of Robotics Research*, vol. 34, no. 12, pp. 1501–1514, 2015.

[10] F. Behrend, “Über die kleinste umbeschriebene und die größte einbeschriebene ellipse eines konvexen bereichs,” *Mathematische Annalen*, vol. 115, no. 1, pp. 379–411, 1938.

[11] A. Bera and D. Manocha, “Realtime multilevel crowd tracking using reciprocal velocity obstacles,” in *Proceedings of International Conference on Pattern Recognition*, Stockholm, Sweden, 2014, pp. 4164–4169.

[12] A. Best, S. Narang, and D. Manocha, “Real-time reciprocal collision avoidance with elliptical agents,” in *Proceedings of IEEE International*

Conference on Robotics and Automation, Stockholm, Sweden, 2016, pp. 298–305.

[13] S. Bischoff and L. Kobbelt, "Ellipsoid decomposition of 3D-models," in *Proceedings of International Symposium on 3D Data Processing Visualization and Transmission*, Padova, Italy, 2002, pp. 480–488.

[14] V. J. Blue and J. L. Adler, "Emergent fundamental pedestrian flows from cellular automata microsimulation," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1644, pp. 29–36, 1998.

[15] ——, "Cellular automata microsimulation of bidirectional pedestrian flows," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1678, pp. 135–141, 1999.

[16] C. Burstedde, K. Klauck, A. Schadschneider, and J. Zittartz, "Simulation of pedestrian dynamics using a two-dimensional cellular automaton," *Physica A: Statistical Mechanics and its Applications*, vol. 295, no. 3-4, pp. 507–525, 2001.

[17] H. Busemann, "The foundations of minkowskian geometry," *Commentarii Mathematici Helvetici*, vol. 24, no. 1, pp. 156–187, 1950.

[18] A. Chakravarthy and D. Ghose, "Obstacle avoidance in a dynamic environment: a collision cone approach," *IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans*, vol. 28, no. 5, pp. 562–574, 1998.

[19] ——, "Generalization of the collision cone approach for motion safety in 3-D environments," *Autonomous Robots*, vol. 32, no. 3, pp. 243–266, Dec. 2011.

[20] B. Chazelle and J. Matousek, "On linear-time deterministic algorithms for optimization problems in fixed dimension," in *Proceedings of the Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, Austin, TX, USA, 1993, pp. 281–290.

[21] Y.-K. Choi, J.-W. Chang, W. Wang, M.-S. Kim, and G. Elber, "Continuous Collision Detection for Ellipsoids," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 2, pp. 311–325, 2009.

[22] Y.-K. Choi, W. Wang, and M.-S. Kim, "Exact collision detection of two moving ellipsoids under rational motions," in *Proceedings of IEEE International Conference on Robotics and Automation*, Taipei, Taiwan, 2003, pp. 349–354.

[23] Y.-K. Choi, W. Wang, Y. Liu, and M.-S. Kim, "Continuous Collision Detection for Two Moving Elliptic Disks," *IEEE Transactions on Robotics and Automation*, vol. 22, no. 2, pp. 213–224, 2006.

[24] H. Choset, K. M. Lynch, S. Hutchinson, G. A. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementation*, ser. A Bradford book. MIT Press, 2005.

[25] M. Chraibi, A. Schadschneider, and A. Seyfried, "On force-based modeling of pedestrian dynamics," in *Modeling, Simulation and Visual Analysis of Crowds: A Multidisciplinary Perspective*, ser. The International Series

in Video Computing, S. Ali, K. Nishino, D. Manocha, and M. Shah, Eds. New York, NY, USA: Springer New York, 2013, vol. 11, pp. 23–41.

[26] M. Chraibi, A. Seyfried, and A. Schadschneider, “Generalized centrifugal-force model for pedestrian dynamics,” *Physical Review E*, vol. 82, p. 046111, 2010.

[27] W.-J. Cong, H.-W. Liu, F. Ye, and S.-S. Zhou, “Rank-two update algorithms for the minimum volume enclosing ellipsoid problem,” *Computational Optimization and Applications*, vol. 51, no. 1, pp. 241–257, 2012.

[28] V. Coppola and J. Woodburn, “Determination of close approaches based on ellipsoidal threat volumes,” *Advances in the Astronautical Sciences*, vol. 102 II, pp. 1013–1023, 1999.

[29] M. Cremer and J. Ludwig, “A fast simulation model for traffic flow on the basis of boolean operations,” *Mathematics and Computers in Simulation*, vol. 28, no. 4, pp. 297–303, 1986.

[30] B. Damas and J. Santos-Victor, “Avoiding moving obstacles: The forbidden velocity map,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, MO, USA, 2009, pp. 4393–4398.

[31] S. Dinas and J. M. Bañón, “A literature review of bounding volumes hierarchy focused on collision detection,” *Ingeniería Y competitividad*, vol. 17, no. 1, pp. 49–62, 2015.

[32] A. Donev, S. Torquato, and F. H. Stillinger, “Neighbor list collision-driven molecular dynamics simulation for nonspherical hard particles. I. Algo-

rithmic details," *Journal of Computational Physics*, vol. 202, no. 2, pp. 737–764, 2005.

[33] ——, "Neighbor list collision-driven molecular dynamics simulation for nonspherical hard particles. II. Applications to ellipses and ellipsoids," *Journal of Computational Physics*, vol. 202, no. 2, pp. 765–793, 2005.

[34] D. Eberly, "Distance Between Ellipses in 2D." [Online]. Available: http://geometrictools.com/

[35] ——, "Intersection of Ellipses." [Online]. Available: http://geometrictools.com/

[36] F. Etayo, L. González-Vega, and N. del Rio, "A new approach to characterizing the relative position of two ellipses depending on one parameter," *Computer Aided Geometric Design*, vol. 23, no. 4, pp. 324–350, 2006.

[37] R. Farouki, C. Neff, and M. O'Connor, "Automatic parsing of degenerate quadric-surface intersections," *ACM Transactions on Graphics*, vol. 8, no. 3, pp. 174–203, 1989.

[38] F. Feurtey, "Simulating the collision avoidance behavior of pedestrians," Master's thesis, The University of Tokyo, 2000.

[39] P. Fiorini and D. Botturi, "Introducing service robotics to the pharmaceutical industry," *Intelligent Service Robotics*, vol. 1, no. 4, pp. 267–280, 2008.

[40] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using the relative velocity paradigm," in *Proceedings of IEEE International*

Conference on Robotics and Automation, Atlanta, GA, USA, 1993, pp. 560–566.

[41] ——, “Motion planning in dynamic environments using velocity obstacles,” *International Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, 1998.

[42] T. Fraichard and H. Asama, “Inevitable collision states - a step towards safer robots,” *Advanced Robotics*, vol. 18, no. 10, pp. 1001–1024, 2004.

[43] C. Fulgenzi, A. Spalanzani, and C. Laugier, “Dynamic obstacle avoidance in uncertain environment combining pvos and occupancy grid,” in *Proceedings of IEEE International Conference on Robotics and Automation*, Roma, Italy, 2007, pp. 1610–1616.

[44] O. Gal, Z. Shiller, and E. Rimon, “Efficient and safe on-line motion planning in dynamic environments,” in *Proceedings of IEEE International Conference on Robotics and Automation*, Kobe, Japan, 2009, pp. 88–93.

[45] A. Giese, D. Latypov, and N. M. Amato, “Reciprocally-Rotating Velocity Obstacles,” in *Proceedings of IEEE International Conference on Robotics and Automation*, Hong Kong, 2014, pp. 3234–3241.

[46] P. J. Gmytrasiewicz, “A decision-theoretic model of coordination and communication in autonomous systems,” Ph.D. dissertation, The University of Michigan, 1992.

[47] C. Goerzen, Z. Kong, and B. Mettler, “A survey of motion planning algorithms from the perspective of autonomous uav guidance,” *Journal of*

Intelligent and Robotic Systems: Theory and Applications, vol. 57, no. 1-4, pp. 65–100, 2010.

[48] E. Goffman and P. Manning, *Relations in public: Microstudies of the public order.* New York: Basic Books, 1971.

[49] L. González-Vega, H. Lombardi, T. Recio, and M.-F. Roy, "Sturm-Habicht Sequence," in *Proceedings of ACM-SIGSAM International Symposium on Symbolic and Algebraic Computation*, New York, NY, USA, 1989, pp. 136–146.

[50] L. González-Vega, T. Recio, H. Lombardi, and M.-F. Roy, "Sturm—Habicht Sequences, Determinants and Real Roots of Univariate Polynomials," in *Quantifier Elimination and Cylindrical Algebraic Decomposition*, B. F. Caviness and J. R. Johnson, Eds. Vienna: Springer Vienna, 1998, pp. 300–316.

[51] L. Gonzalez-Vega and E. Mainar, "Solving the separation problem for two ellipsoids involving only the evaluation of six polynomials," in *Proceedings of Milestones in Computer Algebra*, Scarborough, Tobago, 2008, pp. 201–208.

[52] G. Guennebaud, B. Jacob *et al.*, "Eigen v3," http://eigen.tuxfamily.org, 2010.

[53] S. J. Guy, J. Chhugani, C. Kim, N. Satish, M. Lin, D. Manocha, and P. Dubey, "ClearPath: Highly Parallel Collision Avoidance for Multi-Agent Simulation," in *Proceedings of ACM SIGGRAPH/Eurographics*

Symposium on Computer Animation, New Orleans, LA, USA, 2009, pp. 177–187.

[54] D. Helbing, I. Farkas, and T. Vicsek, "Simulating dynamical features of escape panic," *Nature*, vol. 407, no. 6803, pp. 487–490, 2000.

[55] D. Helbing and P. Molnár, "Social force model for pedestrian dynamics," *Physical Review E*, vol. 51, no. 5, pp. 4282–4286, 1995.

[56] D.-E. Hyun, S.-H. Yoon, M.-S. Kim, and B. Jüttler, "Modeling and deformation of arms and legs based on ellipsoidal sweeping," in *Proceedings of Pacific Conference on Computer Graphics and Applications*, Canmore, Canada, 2003, pp. 204–212.

[57] K. Ioannidis, G. C. Sirakoulis, and I. Andreadis, "A PATH PLANNING METHOD BASED ON CELLULAR AUTOMATA FOR COOPERATIVE ROBOTS," *Applied Artificial Intelligence*, vol. 25, no. 8, pp. 721–745, 2011.

[58] D. Jacka, "High-level control of agent-based crowds by means of general constraints," Ph.D. dissertation, University of Cape Town, 2009.

[59] J. D. Jeon and B. H. Lee, "Ellipse-based velocity obstacles for local navigation of holonomic mobile robot," *Electronics Letters*, vol. 50, no. 18, pp. 1279–1281, 2014.

[60] ——, "Wheel velocity obstacles for differential drive robot navigation," *Journal of Automation and Control Engineering*, vol. 3, no. 5, pp. 347–353, 2015.

[61] X. Jia, Y.-K. Choi, B. Mourrain, and W. Wang, "An algebraic approach to continuous collision detection for ellipsoids," *Computer Aided Geometric Design*, vol. 28, no. 3, pp. 164–176, 2011.

[62] F. John, "Extremum Problems with Inequalities as Subsidiary Conditions," in *Traces and Emergence of Nonlinear Programming*, G. Giorgi and T. H. Kjeldsen, Eds. Springer Basel, 2014, pp. 197–215.

[63] M.-Y. Ju, J.-S. Liu, S.-P. Shiang, Y.-R. Chien, K.-S. Hwang, and W.-C. Lee, "A novel collision detection method based on enclosed ellipsoid," in *Proceedings of IEEE International Conference on Robotics and Automation*, Seoul, South Korea, 2001, pp. 2897–2902.

[64] I. Karamouzas and S. J. Guy, "Prioritized group navigation with Formation Velocity Obstacles," in *Proceedings of IEEE International Conference on Robotics and Automation*, Seattle, WA, USA, 2015, pp. 5983–5989.

[65] L. G. Khachiyan, "Rounding of polytopes in the real number model of computation," *Mathematics of Operations Research*, vol. 21, no. 2, pp. 307–320, 1996.

[66] S. Kim, S. J. Guy, W. Liu, D. Wilkie, R. W. H. Lau, M. C. Lin, and D. Manocha, "BRVO: Predicting pedestrian trajectories using velocity-space reasoning," *International Journal of Robotics Research*, vol. 34, no. 2, pp. 201–217, 2015.

[67] A. Kirchner, H. Klüpfel, K. Nishinari, A. Schadschneider, and M. Schreckenberg, "Discretization effects and the influence of walking speed in cel-

lular automata models for pedestrian dynamics," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2004, no. 10, pp. 1–21, 2004.

[68] B. Kluge and E. Prassler, "Reflective navigation: Individual behaviors and group behaviors," in *Proceedings of IEEE International Conference on Robotics and Automation*, New Orleans, LA, USA, 2004, pp. 4172–4177.

[69] P. Kumar and E. Yildirim, "Minimum-volume enclosing ellipsoids and core sets," *Journal of Optimization Theory and Applications*, vol. 126, no. 1, pp. 1–21, 2005.

[70] Y. Kuwata, M. T. Wolf, D. Zarzhitsky, and T. L. Huntsberger, "Safe Maritime Autonomous Navigation With COLREGS, Using Velocity Obstacles," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, pp. 110–119, 2014.

[71] E. Lalish and K. Morgansen, "Distributed reactive collision avoidance," *Autonomous Robots*, vol. 32, no. 3, pp. 207–226, 2012.

[72] F. Large, C. Laugier, and Z. Shiller, "Navigation among moving obstacles using the NLVO: Principles and applications to intelligent vehicles," *Autonomous Robots*, vol. 19, no. 2, pp. 159–171, 2005.

[73] B. H. Lee, J. D. Jeon, and J. H. Oh, "Velocity obstacle based local collision avoidance for a holonomic elliptic robot," *Autonomous Robots*, in press.

[74] C. Lennerz and E. Schomer, "Efficient distance computation for quadratic curves and surfaces," in *Proceedings of Geometric Modeling and Processing*, Saitama, Japan, 2002, pp. 60–69.

[75] J. Z. Levin, “Mathematical models for determining the intersections of quadric surfaces,” *Computer Graphics and Image Processing*, vol. 11, no. 1, pp. 73–87, 1979.

[76] X. Lin and T.-T. Ng, “Contact detection algorithms for three-dimensional ellipsoids in discrete element modelling,” *International Journal for Numerical and Analytical Methods in Geomechanics*, vol. 19, no. 9, pp. 653–659, 1995.

[77] T. Lozano-Pérez, “Spatial Planning: A Configuration Space Approach,” *IEEE Transactions on Computers*, vol. C-32, no. 2, pp. 108–120, 1983.

[78] L. Lu, Y.-K. Choi, W. Wang, and M.-S. Kim, “Variational 3d shape segmentation for bounding volume computation,” *Computer Graphics Forum*, vol. 26, no. 3, pp. 329–338, 2007.

[79] R. A. Metoyer and J. K. Hodgins, “Reactive pedestrian path following from examples,” *The Visual Computer*, vol. 20, no. 10, pp. 635–649, 2004.

[80] C. B. Moler, “ROOTS - Of Polynomials, That Is,” *The MathWorks Newsletter*, vol. 5, no. 1, pp. 8–9, 1991.

[81] C. B. Moler and G. W. Stewart, “An Algorithm for Generalized Matrix Eigenvalue Problems,” *SIAM Journal on Numerical Analysis*, vol. 10, no. 2, pp. 241–256, 1973.

[82] S. Narang and A. Best, “Comp 790 Research: Density Modeling and Control via Intention Filters.” [Online]. Available: http://www.cs.unc.edu/ best/comp790/

[83] H. Ouadfel and L. Rothenburg, “An algorithm for detecting inter-ellipsoid contacts,” *Computers and Geotechnics*, vol. 24, no. 4, pp. 245–263, 1999.

[84] E. Owen and L. Montano, “Motion planning in dynamic environments using the velocity space,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Edmonton, AB, Canada, 2005, pp. 997–1002.

[85] ——, “A robocentric motion planner for dynamic environments using the velocity space,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China, 2006, pp. 4368–4374.

[86] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool, “You’ll never walk alone: Modeling social behavior for multi-target tracking,” in *Proceedings of IEEE International Conference on Computer Vision*, Kyoto, Japan, 2009, pp. 261–268.

[87] J. W. Perram and M. S. Wertheim, “Statistical mechanics of hard ellipsoids. I. overlap algorithm and the contact function,” *Journal of Computational Physics*, vol. 58, no. 3, pp. 409–416, 1985.

[88] J. Perram, J. Rasmussen, E. Præstgaard, and J. Lebowitz, “Ellipsoid contact potential: Theory and relation to overlap potentials,” *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, vol. 54, no. 6, pp. 6565–6572, 1996.

[89] M. J. Post, “Minimum spanning ellipsoids,” in *Proceedings of the Sixteenth Annual ACM Symposium on Theory of Computing*, New York, NY, USA, 1984, pp. 108–116.

[90] E. Prassler, J. Scholz, and P. Fiorini, “Navigating a robotic wheelchair in a railway station during rush hour,” *International Journal of Robotics Research*, vol. 18, no. 7, pp. 711–727, 1999.

[91] ——, “A robotic wheelchair for crowded public environments,” *IEEE Robotics and Automation Magazine*, vol. 8, no. 1, pp. 38–45, 2001.

[92] E. Prassler, J. Scholz, M. Strobel, and P. Fiorini, “Maid: A robotic wheelchair operating in public environments,” in *Sensor Based Intelligent Robots*, ser. Lecture Notes in Computer Science, H. I. Christensen, H. Bunke, and H. Noltemeier, Eds. Springer Berlin Heidelberg, 1999, vol. 1724, pp. 68–95.

[93] C. Reinholtz, T. Alberi, D. Anderson, A. Bacha, C. Bauman, S. Cacciola, P. Currier, A. Dalton, J. Farmer, R. Faruque, M. Fleming, S. Frash, G. Gothing, J. Hurdus, S. Kimmel, C. Sharkey, A. Taylor, C. Terwelp, D. V. Covern, M. Webster, and A. Wicks, “Victor Tango,” *DARPA Urban Challenge Technical Paper*, 2007.

[94] E. Rimon and S. P. Boyd, “Obstacle Collision Detection Using Best Ellipsoid Fit,” *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 18, no. 2, pp. 105–126, 1997.

[95] L. Rothenburg and R. J. Bathurst, “Numerical simulation of idealized granular assemblies with plane elliptical particles,” *Computers and Geotechnics*, vol. 11, no. 4, pp. 315–329, 1991.

[96] M. Rufli, J. Alonso-Mora, and R. Siegwart, "Reciprocal collision avoidance with motion continuity constraints," *IEEE Transactions on Robotics and Automation*, vol. 29, no. 4, pp. 899–912, 2013.

[97] H.-P. Schröcker, "Uniqueness results for minimal enclosing ellipsoids," *Computer Aided Geometric Design*, vol. 25, no. 9, pp. 756–762, 2008.

[98] S.-P. Shiang, J.-S. Liu, and Y.-R. Chien, "Estimate of minimum distance between convex polyhedra based on enclosed ellipsoids," in *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, Takamatsu, Japan, 2000, pp. 739–744.

[99] Z. Shiller, F. Large, and S. Sekhavat, "Motion planning in dynamic environments: Obstacles moving along arbitrary trajectories," in *Proceedings of IEEE International Conference on Robotics and Automation*, Seoul, South Korea, 2001, pp. 3716–3721.

[100] B. W. Silverman and D. M. Titterington, "Minimum covering ellipses," *SIAM Journal on Scientific and Statistical Computing*, vol. 1, no. 4, pp. 401–409, 1980.

[101] P. Simari and K. Singh, "Extraction and remeshing of ellipsoidal representations from mesh data," in *Proceedings of Graphics Interface*, Victoria, BC, Canada, 2005, pp. 161–168.

[102] J. Snape, J. van den Berg, S. J. Guy, and D. Manocha, "Independent navigation of multiple mobile robots with hybrid reciprocal velocity obstacles," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, MO, USA, 2009, pp. 5917–5922.

[103] ——, “Smooth and collision-free navigation for multiple robots under differential-drive constraints,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, 2010, pp. 4584–4589.

[104] ——, “The hybrid reciprocal velocity obstacle,” *IEEE Transactions on Robotics and Automation*, vol. 27, no. 4, pp. 696–706, 2011.

[105] K.-A. Sohn, B. Juttler, M.-S. Kim, and W. Wang, “Computing distances between surfaces using line geometry,” in *Proceedings of the 10th Pacific Conference on Computer Graphics and Applications*, Beijing, China, 2002, pp. 236–245.

[106] P. Sun and R. M. Freund, “Computation of minimum-volume covering ellipsoids,” *Operations Research*, vol. 52, no. 5, pp. 690–706, 2004.

[107] U. A. Syed and F. Kunwar, “Cellular Automata Based Real-time Path-planning for Mobile Robots,” *International Journal of Advanced Robotic Systems*, vol. 11, no. 1, pp. 1–15, 2014.

[108] M. Takahashi, T. Suzuki, T. Matsumura, and A. Yorozu, “Dynamic Obstacle Avoidance with Simultaneous Translational and Rotational Motion Control for Autonomous Mobile Robot,” in *Informatics in Control, Automation and Robotics*, ser. Lecture Notes in Electrical Engineering, J.-L. Ferrier, A. Bernard, O. Gusikhin, and K. Madani, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, vol. 174, pp. 51–64.

[109] J. Ting, "A robust algorithm for ellipse-based discrete element modelling of granular materials," *Computers and Geotechnics*, vol. 13, no. 3, pp. 175–186, 1992.

[110] J. M. Ting, M. Khwaja, L. R. Meachum, and J. D. Rowell, "Ellipse-based discrete element model for granular materials," *International Journal for Numerical and Analytical Methods in Geomechanics*, vol. 17, no. 9, pp. 603–623, 1993.

[111] D. M. Titterington, "Optimal design: some geometrical aspects of D optimality," *Biometrika*, vol. 62, no. 2, pp. 313–320, 1975.

[112] ——, "Estimation of correlation coefficients by ellipsoidal trimming," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 27, no. 3, pp. 227–234, 1978.

[113] M. J. Todd and E. A. Yildirim, "On Khachiyan's algorithm for the computation of minimum-volume enclosing ellipsoids," *Discrete Applied Mathematics*, vol. 155, no. 13, pp. 1731–1744, 2007.

[114] L. Tychonievich, D. Zaret, J. Mantegna, R. Evans, E. Muehle, and S. Martin, "A maneuvering-board approach to path planning with moving obstacles," in *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, San Francisco, CA, USA, 1989, pp. 1017–1021.

[115] J. van den Berg, J. Snape, S. J. Guy, and D. Manocha, "Reciprocal collision avoidance with acceleration-velocity obstacles," in *Proceedings of IEEE International Conference on Robotics and Automation*, Shanghai, China, 2011, pp. 3475–3482.

[116] J. van den Berg, S. J. Guy, M. Lin, and D. Manocha, "Reciprocal n-body collision avoidance," in *Robotics Research*, ser. Springer Tracts in Advanced Robotics, C. Pradalier, R. Siegwart, and G. Hirzinger, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, vol. 70, no. STAR, pp. 3–19.

[117] J. van den Berg, S. J. Guy, J. Snape, M. C. Lin, and D. Manocha, "RVO2 Library: Reciprocal collision avoidance for real-time multi-agent simulation," http://gamma.cs.unc.edu/RVO2/, 2011.

[118] J. van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *Proceedings of IEEE International Conference on Robotics and Automation*, Pasadena, CA, USA, 2008, pp. 1928–1935.

[119] W. Wang, Y.-K. Choi, B. Chan, M.-S. Kim, and J. Wang, "Efficient Collision Detection for Moving Ellipsoids Using Separating Planes," in *Geometric Modelling*, S. Hahmann, G. Brunnett, G. Farin, and R. Goldman, Eds. Vienna: Springer Vienna, 2004, pp. 235–246.

[120] W. Wang, J. Wang, and M.-S. Kim, "An algebraic condition for the separation of two ellipsoids," *Computer Aided Geometric Design*, vol. 18, no. 6, pp. 531–539, 2001.

[121] E. Welzl, "Smallest enclosing disks (balls and ellipsoids)," in *New Results and New Trends in Computer Science*, ser. Lecture Notes in Computer Science, H. Maurer, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1991, pp. 359–370.

[122] Wikipedia, “Velocity obstacle.” [Online]. Available: https://en.wikipedia.org/wiki/Velocity_obstacle

[123] I. Wilf and Y. Manor, “Quadric-surface intersection curves: shape and structure,” *Computer-Aided Design*, vol. 25, no. 10, pp. 633–643, 1993.

[124] D. Wilkie, J. van den Berg, and D. Manocha, “Generalized velocity obstacles,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, MO, USA, 2009, pp. 5573–5578.

[125] C.-J. Wu, “On the representation and collision detection of robots,” *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 16, no. 2, pp. 151–168, 1996.

[126] K. Yamamoto, S. Kokubo, and K. Nishinari, “Simulation for pedestrian dynamics by real-coded cellular automata (RCA),” *Physica A: Statistical Mechanics and its Applications*, vol. 379, no. 2, pp. 654–660, 2007.

[127] W. J. Yu, R. Chen, L. Y. Dong, and S. Q. Dai, “Centrifugal force model for pedestrian dynamics,” *Physical Review E*, vol. 72, no. 2, p. 026112, 2005.

초록

다중 로봇의 무충돌 경로 계획은 전역 경로 계획과 지역 경로 계획으로 나뉘어 연구됐다. 전역 경로 계획이 전체 환경 정보를 활용하여 목표 지점까지의 최단 경로를 생성하는 것이라면, 지역 경로 계획은 전역 경로 계획에서 생성한 경로를 주변 동적 장애물과 로봇의 이동 제한 조건을 고려하여 조정하는 것이다. 지역 경로 계획 방법에서는 효율적인 연산을 위해 로봇과 장애물을 간단한 기하 도형으로 근사해왔는데, 이 중 원은 충돌 감지가 비교적 간단하다는 점 때문에 많이 사용되었다. 하지만 긴 로봇이나 장애물을 원으로 근사할 시 공간 낭비가 심해 로봇이 장애물과의 충돌을 회피하기 위해 더 먼 거리를 돌아간다는 단점이 있었다.

본 학위 논문에서는 비등방 로봇을 타원으로 근사하고 타원 로봇의 지역 경로 계획 문제를 해결한다. 타원은 다른 도형에 비해 2차원 평면상에서 로봇과 장애물을 더 효율적으로 근사하지만, 충돌을 감지하는 것은 더 복잡하다. 따라서 어떤 조건에서 두 타원 사이에 충돌이 발생하는지 알기 위해 형태 공간 방법과 대수적인 접근법을 활용한다. 첫 번째 방법에서는 타원 로봇의 형태 공간 내에서 타원 장애물에 대응하는 형태 장애물의 경계를 매개변수를 활용해 유도한다. 이를 통해 두 타원이 충돌하는 상대 위치 및 방위 조건의 전체 형태를 쉽게 파악하지만, 특정 위치 및 방위에 있는 두 타원의 위치 관계를 즉시 판단할 수 없다는 한계점이 있다. 두 번째 방법에서는 충돌 조건을 네 부등식으로 나타내어 충돌 여부를 빠르게 판단하지만, 전체 영역을 파악할 수 없다는 단점이 있다. 두 방법은 상호보완적이므로 충돌 회피 문제를 해결할 때 상황에 따라 더 적합한 것을 사용한다.

다음으로 타원 로봇이 동적 장애물과 충돌하지 않고 목적지에 도달케 하는 속도 기반의 충돌 회피 방법을 제안한다. 이 방법은 충돌 회피 문제를 속도 공간에서의

기하학적 최적화 문제로 변환하여 해결하는 것으로, 본 학위 논문에서는 실시간 제어를 위해 로봇의 선형 운동과 회전 운동을 순차적으로 나누어 결정한다. 선형 제어 단계에서는 일정 시간 이내에 장애물과 충돌이 발생하는 로봇의 모든 선속도 집합을 타원 기반의 속도 장애물이라 정의하고, 이 영역 밖에서 목표 지점으로 가장 빠르게 향하는 선속도를 선택한다. 회전 제어 단계에서는 로봇이 장애물과의 충돌을 회피할 때 회전을 통해 더 짧은 거리를 우회하도록 한다. 이를 위해 로봇이 장애물과 충돌하지 않는 회전 각도를 구하고, 이 안에서 장애물에 대한 상대 속도와 타원의 장축이 평행하도록 회전시켜 장애물을 효율적으로 회피하도록 한다.

마지막으로 장애물 회피 알고리즘을 다개체 로봇 충돌 회피가 가능하도록 확장한다. 이때 모든 로봇에 같은 충돌 회피 알고리즘이 적용되었다는 상호성 개념을 적용하여 통신 없이 다른 로봇의 향후 움직임을 예측한다. 선형 운동 제어 단계에는 혼합 상호 속도 장애물 개념을 적용해 앞에서 구한 타원 기반의 속도 장애물을 단순히 평행이동시켜 일정 시간 이내에 로봇 간 충돌이 발생하게 하는 선속도의 집합을 유도하고, 이 영역 밖의 선속도를 선택한다. 회전 운동 제어 단계에서는 한 로봇이 회전하면 다른 로봇도 동일한 각도로 회전한다는 가정하에 충돌하지 않는 회전각의 범위를 구하고, 이 안의 범위에서 로봇의 각속도를 선택한다.

그리고 제안한 알고리즘은 여러 시뮬레이션을 통해 원으로 근사하는 기존 방법과 타원으로 근사하지만 회전을 고려하지 않은 방법, 회전 가능한 다각형으로 근사하는 방법에 비해 이동 시간과 거리 면에서 뛰어남을 입증하였다. 동적 장애물 회피와 관련해서는 보행자 데이터셋을 활용해 1,000번의 몬테카를로 시뮬레이션을 통해 알고리즘을 검증했다. 그리고 다중 로봇 충돌 회피 방법에서는 로봇이 초기에 원 형태로 배치해 있고 그 반대편으로 이동하도록 지정하여 원의 중심에서 모든 로봇이 만나는 시나리오를 통해 제안한 알고리즘이 뛰어남을 보였다.

주요어: 지역 경로 계획, 타원, 속도 장애물, 상호 충돌 회피, 다중 로봇 시스템

학번: 2011-20924