PURSUIT-EVASION AND DIFFERENTIAL GAMES UNDER UNCERTAINTIES

A Thesis
Presented to
The Academic Faculty

by

Wei Sun

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Aerospace Engineering

Georgia Institute of Technology
August 2017

PURSUIT-EVASION AND DIFFERENTIAL GAMES UNDER UNCERTAINTIES

Approved by:

Professor Panagiotis Tsiotras
Committee Chair, Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Professor Evangelos A. Theodorou,
Co-Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Professor Eric Feron
School of Aerospace Engineering
Georgia Institute of Technology

Professor Eric N. Johnson
School of Aerospace Engineering
Georgia Institute of Technology

Professor Anthony J. Yezzi
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Date Approved: 17 May 2017

To my parents and my sister.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my advisor Dr. P. Tsiotras for his support and guidance throughout my PhD studies. I would also like to extend my appreciation to my co-advisor Dr. E. A. Theodorou and other committee members: Dr. E. M. Feron and Dr. A. Yezzi for their true interest in evaluating my work.

I owe the greatest debt of gratitude to my parents and my sister for their never ending support during my graduate studies. This work is dedicated to them as a token of appreciation.

Lastly, but by no means least, I wish to thank my labmates Dr. Efstathios Bakolas, Dr. Nuno Filipe, Dr. Oktay Arslan, Mr. Ioannis Exarchos, Mr. Spyridon Zafeiropoulos, Mr. Florian Hauer, Mr. Changxi You among others who have given me many inspirations during my years at Georgia Tech. I would like to extend my thanks to Dr. Andrea L'Afflitto, Dr. Haocao Li, Mr. Tanmay Rajpurohit, Mr. Yunpeng Pan, who have been talented and considerate friends with me. Furthermore, the help in my living and studying by Frank Jiang and his family is gratefully appreciated.

TABLE OF CONTENTS

LIST OF FIGURES

SUMMARY

Differential games involve multi-person decision making under conflicts in the context of dynamical systems. It has found many applications in a large range of areas, including aeronautics, biology, ecology, economics, engineering, management science, operations research, etc. Often, the decisions made by the players that join the differential game are susceptible to uncertainties that are pervasive in realistic differential game scenarios. Uncertainties entering the system can be divided into three main categories, namely, external/environmental uncertainties, internal/dynamical uncertainties and observation uncertainties.

The research effort in this dissertation pursues two main objectives. First, we provide analytical and numerical methods to deal with environmental and dynamical uncertainties. In particular, we solve pursuit-evasion games under different forms of external flow fields to demonstrate how to cope with differential games under environmental uncertainties.

Pursuit-evasion problems under external flow fields are generally hard to solve. We start by focusing on finding the strategy of the pursuers in the problem of pursuit and evasion between a set of targets and a team of pursuers distributed in the plane subject to a time-varying flow field. The objective of the pursuers is to intercept the moving targets which, however, are not affected by the presence of the environmental disturbance. We first solve the multiple-pursuers-one-target problem by assigning only one pursuer to chase a single target at every instant of time, based on a generalized Voronoi partition of the plane where "closeness" is measured by minimum time-to-intercept. We then apply the scheme to multiple-target problems where the pursuer assignment changes dynamically based on this partition during the pursuit.

Next, we address the differential game of pursuit and evasion between two players in the presence of a spatial flow field, which is approximated by a time-invariant affine function of the state. By utilizing standard techniques from differential game theory, we characterize the regions of initial conditions that lead to capture, and we analytically derive the optimal strategies of the pursuer and the evader within the respective capture regions of the pursuers.

Pursuit-evasion games between multiple pursuers and one evader under general spatiotemporal flow fields are also dealt with through a reachable set analysis and by utilizing the numerical level set method. Conditions for the game to be terminated are given in terms of reachable set inclusions. Level set equations are defined and solved in order to generate the forward reachable sets of the pursuers and the evader. The time-optimal trajectories and the corresponding optimal strategies are subsequently retrieved from these level sets. We apply this reachability-based scheme to deal with pursuit-evasion of multiple agents both in 2-dimensional and 3-dimensional spaces. We also extend this scheme in a probablistic setting to deal with a two-player pursuit-evasion game in the presence of stochastic environmental disturbances.

The second objective of this research is to present an efficient algorithm to solve general two player differential game problems and extend it to a stochastic formulation in order to tackle differential games under dynamical uncertainties. In particular, we propose a Game-Theoretic Differential Dynamic Programming (GT-DDP) algorithm in continuous time by providing a set of backward differential equations for the value function expansion without assuming closeness of the initial nominal control to the optimal control solution, and derive the update law for the controls. The effect of the game-theoretic formulation in the feed-forward and feedback parts of the control policies is analyzed by applying different control gains to a system affected by additive noise. A stochastic version of GT-DDP algorithm is then derived to solve a differential game under dynamical uncertainties modeled by a state-dependent Gaussian noise.

We present the update law for the minimizing and maximizing controls for both players and provide a set of backward differential equations for the second-order value function approximation. We find the extra terms in the backward propagation equations that arise from the stochastic assumption compared with the original GT-DDP and present the corresponding SGT-DDP algorithm.

CHAPTER I

INTRODUCTION

1.1 Motivation and Goals

Differential games originate from game theory, which was introduced by von Neumann and Morgenstern [151] in the mid '40s. Differential games involve multi-person decision making under conflicts in the context of dynamical systems. Since the pioneer work of Isaacs in differential games [62], the theory has been developed from its early focus on military pursuit-evasion problems and expanded towards a variety of areas, including aeronautics, biology, ecology, economics, engineering, management science, operations research, political science, psychology, to name just a few [2, 17, 38, 44, 67, 97, 106, 157, 161].

Each player involved in a differential game makes a decision based on its own interests and those different interests may lead to conflict and/or cooperation. An outcome is induced from the combined decisions of the players. What makes the problem interesting is that the decision made by each player could affect the outcome of all players. In this process, both the decision-making and the outcome of the game may alter due to the change in the dynamics or the information each individual have. The dynamics or the information each individual have can be easily affected by uncertainties in the game. To be specific, the uncertainties that enter the system can be divided into three main categories, namely, external/environmental uncertainties, endogenous/dynamical uncertainties, and observation uncertainties. Owing to the pervasive nature of uncertainties in realistic differential game scenarios, efforts should be undertaken in dealing with differential games under uncertainties.

In this work, we focus on the modeling and analysis of differential games subject

to environmental and dynamical uncertainties. In particular, we single out pursuit-evasion games to demonstrate how to deal with differential games under environmental uncertainties.

Pursuit-evasion games is a well-established branch of differential games with various applications in both military and civilian areas. It involves two sides with conflicting interests, where one (or a group) of pursuers aim to catch one (or a group) of evaders who try to avoid capture. Despite the plethora of work in this area [17, 28, 47, 50, 52, 57, 62, 97, 113, 130, 134], few approaches have taken into consideration how dynamic environmental conditions may affect the outcome of the game. On the other hand, due to the rapid advancement of technology, autonomous vehicles with various sizes have been developed for a wide range of tasks and many of them are susceptible to external environmental disturbances. For instance, in a pursuit-evasion game, when either the pursuer or the evader (or both) is a small autonomous underwater vehicle (AUV) or small unmanned aerial vehicle (UAV), the presence of dynamic sea currents or winds may significantly affect the vehicle motion. As a result, during the pursuit-evasion of these vehicles, their optimal behaviors may be greatly affected by the existence of external dynamically changing ambient weather and wind conditions. Therefore, the decision mechanisms for the pursuer and the evader should be explored such that they can act efficiently despite the presence of environmental disturbances.

It is also very common in practice for dynamical systems to have dynamical uncertainties, since most of the dynamic models used nowadays are only simplifications or approximations of the real systems, e.g., manipulators, aircraft and biped robots [20, 45, 138, 153]. In order to deal with this problem, we need new tools to obtain a dynamical model that takes into account the dynamical uncertainties. After a relatively realistic model is achieved, an efficient method is required to solve the follow-up differential game problem. In particular, we utilize the Differential Dynamic

Programming (DDP) [63] technique and derive a game-theoretic version of DDP that solves the Hamilton-Jacobi-Isaacs equation associated with a differential game problem. DDP is a well-known trajectory optimization method that iteratively finds a locally optimal control policy starting from a nominal control and state trajectory. Since its introduction in [63], there has been a plethora of variations and applications of DDP within the controls and robotics communities. Starting with a differential game-theoretic formulation and application to bipedal locomotion [105], [104] to receding horizon [144], and stochastic control formulations [147, 148], DDP has become one of the standard methods for trajectory optimization with a broad range of applications [1, 5, 6, 43, 145, 146, 148]. Our work focus on the derivation and application of continuous-time Game-Theoretic DDP (GT-DDP). This differential game-theoretic or min-max formulation is closely related to the H^∞ control theory [40, 41, 49, 71, 119, 149, 160]. The H^∞ control theory aims to achieve robustness of systems against model uncertainty. The basic idea is to keep the sensitivity γ of the feedback control loop against a disturbance input small enough such that any disturbance subject to modeling error can be suppressed if the gain of mapping from the state error to the disturbance is bounded by $1/\gamma$ in terms of H^∞ norm. It has been shown [16] that the H^∞ control problem can be recast as a min-max problem subject to the Hamilton-Jacobi-Isaacs (HJI) equation of a value function. In this min-max problem, the objective is to obtain a control that minimizes a given performance index under worst possible disturbances or parameter variations. Therefore, the GT-DDP algorithm we developed can also be utilized to solve H^∞ control problems. H^∞ optimization has also been extended in the form of robust H^∞ control to deal with uncertain systems [65, 72, 154, 156].

In the next three sections, we focus on some of the recent advances in various research topics relevant to the scope of this dissertation. In particular, we review the major contributions in the literature on pursuit-evasion problems involving teams of

autonomous vehicles. Subsequently, we review the differential dynamic programming method in dealing with optimal control problems. Given the vast body of work related to these popular research topics available in the literature, our literature review is not meant to be exhaustive, but rather indicative of some recent trends in the fields related to the scope of this dissertation.

1.2 Task Assignment Problem of Spatially Distributed Group of Autonomous Agents and Voronoi Diagram Partitions

A significant body of work in the field of multi-agent systems deals with distributed control/task assignment problems. This class of problems can be traced back to steering behavior of autonomous mobile agents studied by Reynolds in [127, 128], which discuss various motion coordination strategies of multi-agent systems such as flocking, leader following, containment, motion alignment and separation, etc. In general, motion coordination problems attempt to reach some global objective through coordination of a group of autonomous vehicles in the context of a system theoretic framework. The rendezvous problem for multi-agents problems, where the goal for the participants is to meet with each other at some common spot, have been examined extensively in [59, 64, 103, 110, 126]. The dynamic routing problem that solves automatic planning of optimal multi-vehicle routes to perform tasks generated by an exogenous process have been studied in [24, 118]. In [78], the performance of a multi-agent risk-sensitive tracking system is evaluated by formulating the tracking problem as an infinite horizon linear exponential quadratic Gaussian problem. UAV applications on motion coordination have been investigated in [89, 94, 95]. Multi-vehicle control problems in the presence of a flow field have also been studied in [54, 114, 120].

An increased amount of attention has been directed to multi-agent systems to

develop more general schemes to address broader classes of task assignment and distributed control problems. For example, potential function methods and Lyapunov-based analysis have been utilized to deal with formation control and other consensus problems for multi-agent systems, a survey of which can be found in [55]. In [4] a game-theoretical formulation is utilized to address vehicle-target assignment problems for teams of non-cooperative autonomous vehicles where collective team objectives are achieved through optimization of utility functions. A scheme to deal with target allocation problems for a teams of UAVs based on mixed integer-linear programming techniques is introduced in [73]. Although the aforementioned approaches apply to various problems with respect to multi-agent systems, the interactions of the environment with the agents is not taken into consideration, which places restrictions on the applicability of these approaches in many real-world applications. In addition, these general task assignment schemes usually come with a heavy computational cost that restrain them from applications that require real-time decision making.

Another tool that is widely utilized in task assignment problems for multi-agent systems is the so-called Voronoi diagram and its generalized forms. The concept of Voronoi Diagram is discussed in [152]. A Voronoi diagram is a spatial partition of a topological space formed by a set of generators with respect to some prescribed distance metric. Each generator is associated with an element of the partition, which is known as the Voronoi cell, such that each point within the Voronoi cell of a particular generator is closer to this generator than any other generators in terms of the prescribed distance metric. A detailed discussion of Voronoi diagrams with different types of distance metric can be found in [109] and the references therein. It has found a large number of applications in fields related to mult-agent systems. For instance, coverage control of a multi-agent network system is studied in [30], where a locational optimization problem is solved with centroidal Voronoi partitions. A partitioning problem with respect to the minimum time-to-reach of the agents is proposed

in [9]. It was subsequently applied to a pursuer assignment problem in [14]. The study of dynamic Voronoi diagram problems that deal with Voronoi-like partitioning problems where the generators are moving rather than being stationary can be found in [35, 36, 129]. In contrast to the standard Voronoi partitioning problem, where all generators are stationary, a sequence of time-evolving Voronoi diagrams is obtained as the solution of the dynamic partitioning problem. In particular, the authors in [35] deal with a pursuer assignment problem between a set of pursuers and a set of evaders. The evaders move in straight lines and are assumed to be slower than the pursuers. The task is to find a rule that assignes each pursuer to an evader such that the evaders can be captured in minimum time through the dynamic Voronoi diagram.

1.3 Multiplayer Pursuit-Evasion Problem

A discussion on the historical background of pursuit-evasion problems can be found in [107]. Two player pursuit-evasion differential games were originally studied by Isaacs in his seminal book [62], which extended the theory of zero-sum games in classical game theory [90] to problems in the context of dynamical systems. The main idea is to associate the solution of the differential game to that of a Hamilton-Jacobi-Isaacs (HJI) equation. Pursuit-evasion between an agile pursuer and an evader with a curvature constraint was studied in Isaacs' Homicidal Chauffeur game [62]. A reversed version of the Homicidal Chauffeur game, where the evader is agile and the player has a curvature constraint, was recently studied in [46, 47]. Another similar game, called the Game of Two Cars [97], focuses on two players, both having a finite maximum turning radius. A general result for the pursuit-evasion problem with curvature constraints was presented in [28]. Pursuit-evasion games with quadratic costs and linear dynamics for the players is addressed in [57] by employing standard linear optimal control techniques.

Stochastic pursuit-evasion games have also received attention over the recent years.

A Markov pursuit-evasion game in the discrete time formulation is presented in [56]. The Markov chain approximation method is also utilized in [76] to deal with stochastic differential games. A linear quadratic pursuit-evasion game is studied in [7]. A stochastic version of the Homicidal Chauffeur game, a pursuit-evasion game between an agile pursuer and an evader having a finite maximum turning radius, is addressed in [113]. An analytical form of the value function in a two-player stochastic pursuit-evasion game is found in [80].

Besides the HJI equation approach, another approach researchers have used when dealing with pursuit-evasion problems is based on reachable set analysis [27, 52, 102]. According to this approach, the reachable state space of the pursuers and the evaders is utilized to find the optimal controls of the pursuer and/or the evader. Reachability set analysis has been applied in performing missile/sensor trade-offs in homing guidance [131], in obtaining escape strategy under pursuit [158], and in finding pursuer control under control constraints [26].

Some of the aforementioned problems have been extended to the case of multiple players. Specifically, multiplayer pursuit-evasion games with quadratic costs and linear dynamics are presented in [137], where the theory of nonzero-sum games in classical game theory is extended to solve such problems. A multiplayer extension of the classical Homicidal Chauffeur game [62] is discussed in [21] where a chain formation of faster, yet less maneuverable, pursuers are utilized to ensure capture of a single slower but agile evader. The group pursuit problems, pursuit-evasion involving multiple players, are, in general, difficult problems to solve due to their complexity [18, 19, 122]. Their solution is also based on the information the pursuers and the targets/evaders have about each other, resulting in either cooperative or non-cooperative strategies [14, 35, 75, 116, 150, 157].

A large number of group pursuit problems focus on the case where a group of

pursuers go after a single evader. In particular, conditions for target interception between multiple pursuers and one evader in the simplest form is studied by Pshenichnyi in [123]. The result is extended to simultaneous k-capture in [19] and ϵ-capture in [70] where the capture occurs when some pursuer is within ϵ distance of the evader instead of coinciding with the evader. Conditions for guaranteed evasion and the corresponding evading strategies are studied in [25] and [61]. Pursuit-evasion between one evader and countably many pursuers is investigated in [60]. In a relay pursuit, generalized Voronoi diagrams are utilized to assign active pursuers dynamically. Capture of the evader is achieved through a relay of the pursuers in a multi-pursuer/one-evader problem [9,14]. The idea has also been applied in [141] to deal with known environmental disturbances and in [39] for cooperative relay tracking of targets. Some results exist for cases with more general dynamics for the agents (pursuers/evaders), but extra assumptions are made for the problem to be tractable. Pursuit-evasion between a group of pursuers and one evader with linear time-varying dynamics is studied in [122], and later on generalized to the case of multiple evaders in [15]. Group pursuit of a target under the so-called "soft" capture, where capture occurs only when at least one pursuer and the target have identical orientation, velocity and acceleration, is investigated in [121] under linear time-invariant dynamics.

Despite the plethora of work in this area, few approaches have taken into consideration how dynamic environmental conditions may affect the outcome of the game. For instance, when either the pursuers or the evaders (or both) are small autonomous underwater vehicles (AUV) or small unmanned aerial vehicles (UAV), the presence of time-varying or spatially-varying sea currents or winds, respectively, may significantly affect the vehicle's motion. As a result, during pursuit-evasion, the optimal behavior of the players, as is determined by the solution of a differential game, may be greatly affected by the existence of an external dynamically changing flow field.

Some optimal control problem formulations have taken into account the effect

of an external flow field. For example, in [96] the authors address the problem of optimal guidance to a specified position of a Dubins vehicle [91] under the influence of an external flow. The minimum-time guidance problem for an isotropic rocket in the presence of wind has been studied in [12]. The problem of minimizing the expected time to steer a Dubins vehicle to a target set in a stochastic wind field has also been discussed in [3]. However, the same level of attention in the literature has not been devoted to pursuit-evasion games with two (or more) competing agents under the influence of external disturbances (e.g., winds or currents). Such problem will be addressed later in this dissertation.

1.4 Differential Dynamic Programming

Differential Dynamic Programming (DDP) is a trajectory optimization method that iteratively finds a locally optimal control policy. The method starts from a nominal control and state trajectory and iterates on the backward propagation of the value function and the forward propagation of state dynamics to update the control and state trajectory. It utilizes locally-quadratic models of the dynamics and value functions. The backward evolution law of the value function is found through an expansion of the Hamilton-Jacobi-Bellman equation associated with the optimal control problem. Since its introduction in 1966 by Mayne [93], and the subsequent discussion in [63] by Jacobson, several variations of DDP have been derived and applied extensively to deterministic and stochastic systems in robotics, autonomous systems and computational neuroscience. In particular, in [148] a discrete time DDP algorithm is derived for nonlinear stochastic systems with state and control multiplicative noise, and then applied to biomechanical models. The resulting algorithm, known as iterative Linear Quadratic Gaussian (iLQG) control, relies on first order expansion of the dynamics. Second-order expansion of stochastic dynamical systems with state and control multiplicative noise is considered in [147]. The resulting algorithm, known as

Stochastic Differential Dynamic Programming (SDDP), is a generalization of iLQG. In [104], a discrete differential game problem is investigated through the extension of the original DDP method and the proposed algorithm is applied to bipedal locomotion. DDP has also been implemented in a receding horizon fashion reminiscent of model predictive control to account for dimensionality issues [144]. The resulting discrete time receding horizon DDP is applied for helicopter acrobatic maneuvers in [1]. In [5] random sampling techniques are proposed to improve the scalability of DDP. In [43] an infinite horizon version of discrete time DDP is derived, and in [146], DDP is applied to deterministic nonlinear systems with control limits and subsequently implemented to control humanoid robot in simulation. DDP has also been used with machine learning methods to deal with optimal control problems with learned dynamics [101, 115].

While DDP was initially derived for continuous-time problems, the bulk of the previous work on applications of DDP has mainly focused on discretep-time formulations of continuous-time optimal control problems. The key idea in the aforementioned discrete-time formulations is to first discretize the dynamics and then use Dynamic Programming (DP) to derive the backward propagation equations for the zeroth, first and second order approximation terms of the value function. Thus, instead of first optimizing to find the optimal control and then discretizing the solution so that it can be applied to a real physical system, in discrete-time DDP discretization is performed first, which is followed by an optimization step to find the optimal control.

1.5 Comments on the Structure of the Dissertation

The contents of this dissertation are divided into four parts. The first part of this dissertation (Chapter 1) contains a literature review on the following topics: 1) Task assignment problems of spatially distributed group of autonomous agents and Voronoi

Diagram Partitions, 2) Multiplayer pursuit-evasion problems and 3) Differential Dynamic Programming.

The second part of the disseration, comprised of Chapters 2-6, focuses on differential games under external disturbances. In particular, in Chapter 2 we address the sequential pursuit problem between multiple pursuers and evaders in the presence of environmental disturbances. In Chapter 3, we consider the differential game of pursuit and evasion between two players on a plane under an external flow field. It is assumed that the flow field is approximated by a time-invariant, spatially-affine function. The goal is to find the region of initial conditions of both players that leads to capture when both players act optimally, and derive the corresponding optimal strategies of the two players when capture is guaranteed. In Chapter 4, we address a multiple-pursuer-one-evader game in an external dynamic flow field without a prespecified structure. Due to the generality of the external flow, Issacs' approach does not readily yield feasible results. Instead, we adopt a reachability-based approach and find the optimal trajectories of the players through the evolution of their reachable sets. In Chapter 5, We apply the reachability-based approach to a multiplayer pursuit-evasion game under general flow field in a 3-dimensional space. In Chapter 6 this approach is extend in a probabilistic setting to deal with pursuit-evasion games in the presence of stochastic environmental disturbances.

The third part of this dissertation (Chapters 7 and 8) deals with differential games under dynamical uncertainties. In Chapter 7, we present the min-max Differential Dynamic Programming (GT-DDP) algorithm to solve general differential game problems. The stochastic extension of the GT-DDP algorithm is derived to solve differential game problems whose dynamical uncertainties are modeled by state dependent additive Gaussian noise. Finally, in Chapter 9, we conclude this disseration with a summary of the contributions and we provide a discussion on future extensions of the results presented in this research.

1.6 Chapter Description

Next, we give a short description of each chapter of this dissertation.

- **A Sequential Pursuer-Target Assignment Problem Under External Disturbances**

  Pursuit-evasion problems under external flow fields are, in general, hard to solve. In Chapter 2, we focus on finding the strategy of the pursuers in the problem of pursuit and evasion between a set of targets and a team of pursuers distributed in the plane subject to an environmental disturbance (e.g., wind, sea current). The objective of the pursuers is to intercept the moving targets which, however, are not affected by the presence of the flow field disturbance. We first solve the multiple-pursuers-one-target problem by assigning only one pursuer to chase the single target at every instant of time, based on a Voronoi-like partition of the plane. During the pursuit, the pursuer assignment changes dynamically based on this partition. We present an algorithm to efficiently update this Voronoi-like partition on-line. We then deal with the original problem by assigning the "closest" pursuer in the sense of time-to-capture to each of the targets and keep the sequential pursuit until all targets have been captured. Simulations are included to illustrate the theoretical results.

- **Pursuit-Evasion Game in Linear Flow Fields**

  In Chapter 3, we address the differential game of pursuit and evasion between two players in the presence of an external flow field. It is assumed that the two players move on the plane using fixed but different speeds, and they are both agile. That is, they steer by choosing at each instant their direction of travel and abrupt heading changes are allowed. The external flow field is approximated by a time-invariant affine function. By utilizing standard techniques from differential game theory, we characterize the regions of initial conditions that lead

to capture, as well as the regions that result in evasion when the two players act optimally. We derive the optimal strategies of the pursuer and the evader within the capture regions. Finally, we present numerical simulations of the resulting pursuer and evader trajectories for several values of the parameters of the external flow field.

- **Pursuit-Evasion Game Under General Flow Fields**

  In Chapter 4, a reachability-based approach is adopted to deal with the pursuit-evasion differential game between one evader and multiple pursuers in the presence of dynamic environmental disturbances (e.g., winds, sea currents). Conditions for the game to be terminated are given in terms of reachable set inclusions. Level set equations are defined and solved to generate the forward reachable sets of the pursuers and the evader. The time-optimal trajectories and the corresponding optimal strategies are subsequently retrieved from these level sets. The pursuers are divided into active pursuers, guards, and redundant pursuers according to their respective roles in the pursuit-evasion game. The proposed scheme is implemented on problems with both simple and realistic time-dependent flow fields, with and without obstacles.

- **Pursuit-Evasion Game Under 3-Dimensional Flow Fields**

  In Chapter 5 we deal with a pursuit-evasion differential game between multiple pursuers and evaders in the 3-dimensional (3D) space under dynamic environmental disturbances (e.g., winds, sea currents). We first recast the problem as a pursuer assignment problem through a tessellation of the 3D space with generalized Voronoi diagrams. Within each partition, the problem is reduced into a multiple-pursuers/one-evader game. This problem is then addressed through a reachability-based approach. We give conditions for the game to be terminated in terms of reachable set inclusions. The reachable sets of the pursuers and the

evader are obtained by solving their corresponding level set equations through the narrow band level set method. The time-optimal trajectories and corresponding optimal strategies can be retrieved afterwards. The proposed scheme is implemented on problems with both simple and realistic and complicated flow fields.

- **Pursuit-Evasion Game Under Stochastic Flow Fields**

  In Chapter 6, we address a two-player pursuit-evasion differential game in the presence of stochastic environmental disturbances through an extension of the reachability-based approach. We utilize the moment expansion method to describe the system with its mean and covariance propagation, which allows us to construct mean and augmented reachable set through error ellipse of 2-dimensional Gaussian process. We give conditions for the game to be terminated in terms of augmented reachable set inclusions in a probabilistic setting. Level set equations are defined and solved to generate the mean reachable sets of the pursuer and the evader with respect to the mean dynamics. We then implement the proposed scheme to a problem with a stochastic flow field.

- **Game Theoretic Continuous Time Differential Dynamic Programming**

  In Chapter 7, we derive a Game Theoretic Differential Dynamic Programming (GT-DDP) algorithm in continuous time to solve a general differential game problem. We provide a set of backward differential equations for the value function expansion without assuming closeness of the initial nominal control to the optimal control solution, and derive the update law for the controls. We introduce the GT-DDP algorithm and analyze the effect of the game theoretic formulation in the feed-forward and feedback parts of the control policies. Furthermore, we investigate the performance of GT-DDP through simulations

on an inverted pendulum with conflicting controls and we apply the control gains on a stochastic system to demonstrate the effect of the design of the cost function to the feed-forward and feedback parts of the control policies.

- **Stochastic Game Theoretic Continuous Time Differential Dynamic Programming**

  A Stochastic Game Theoretic Differential Dynamic Programming (SGT-DDP) algorithm is derived in Chapter 8 to solve a differential game under stochastic dynamics. We present the update law for the minimizing and maximizing controls for both players and provide a set of backward differential equations for the second order value function approximation. We find the extra terms in the backward propagation equations that arise from the stochastic assumption compared with the original GT-DDP. We present the SGT-DDP algorithm and analyze how the design of the cost function affects the feed-forward and feedback parts of the control policies under the game theoretic formulation. The performance of SGT-DDP is then investigated through simulations on three examples, namely, a first order nonlinear system, the inverted pendulum and the cart pole problems with conflicting controls. We conclude with some possible future extensions.

- **Conclusions and Future Research**

  In Chapter 9 we summarize our research efforts and highlight possible directions for future research. Our emphasis is on exploring the possibility of applying the reachable set analysis as a means to address problems involving teams of spatially distributed autonomous vehicles under stochastic environmental disturbances. It would also be of interest to combine machine learning methods and SGT-DDP algorithm efficiently to deal with differential game problems under dynamical uncertainties.

CHAPTER II

A SEQUENTIAL PURSUER-TARGET ASSIGNMENT PROBLEM UNDER EXTERNAL DISTURBANCES

2.1 Introduction

Consider a scenario where a group of helicopters or small UAVs in a wind field are trying to capture some vehicles moving on the ground, or a team of small marine or underwater vehicles attempting to reach some ships which are large enough so that the sea currents do not significantly affect their motions. Given such a group of pursuers, we want to find a pursuit strategy to intercept the targets in minimum time. Problems of this nature fall under the general class of group pursuit problems. In order to solve this problem, in this Chapter we first focus on the multiple-pursuer-one-target problem and propose a *sequential* pursuit strategy. By sequential (or relay) pursuit we mean that for each target, only one pursuer is assigned to chase the target at every instant of time. In addition to simplifying significantly the group pursuit problem, a relay pursuit strategy may be desirable in cases where the power or energy/fuel consumption of the agents is an important factor, when the agents also play a dual role as guardians protecting a certain area, or in order to account for possible deceptive strategies of an opponent. Then in the case of multiple targets, we assign one pursuer to each of the target according to some reasonable criteria until all the targets are captured.

For the multiple-pursuer-one-target problem, in contrast to most standard pursuit-evasion problem formulations [14, 61, 66, 70, 74, 121, 123], where the effect of the environment is not taken into consideration, in our problem setup (only) the pursuers are affected by known exogenous environmental conditions (e.g., the winds or currents).

Furthermore, as with all pursuit games, the solution of this problem depends on the knowledge each pursuer has about the current and future position of the target. Note that if the whole target trajectory is given a priori, the problem can be solved using optimal control theory. In this Chapter it will be assumed that each pursuer has a stroboscopic view of the target position. That is, each pursuer knows the current position of the target but not its future position nor its velocity. Our objective is to find the optimal assignment of which pursuer to go after the target at each instant of time so as to reduce or minimize the capture time.

Our strategy to solve this problem will be based on the dynamic assignment of the best pursuer to go after the target based on a Voronoi-like partition of the plane called the Zermelo-Voronoi partition, or the Zermelo-Voronoi Diagram (ZVD) [9]. Such Voronoi-like diagrams have been previously introduced in [9, 10, 14] and use time-to-intercept as the relevant distance metric. Essentially, a ZVD allows one to succinctly encode the "isocost" surfaces of the associated minimum-time to intercept problems emanating from the pursuer locations. The difficulty in our problem arises from the fact that, owing to the presence of a wind field, a point in the plane can be close to a pursuer in terms of Euclidean distance, but may not be close in terms of minimum-time to intercept. As a result, standard Voronoi partitions for this problem may lead to erroneous conclusions.

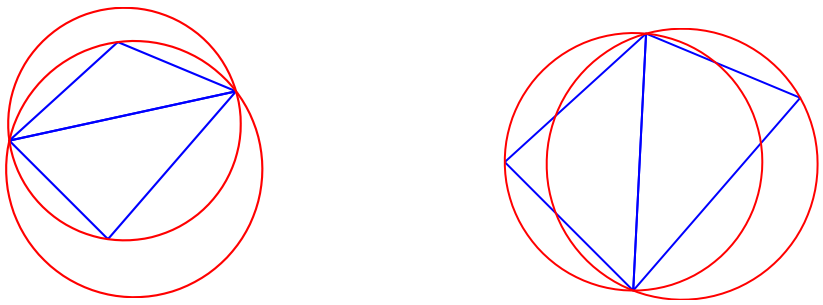
Our method in the multiple-pursuer-one-target problem can be extended naturally to solve our original problem with multiple targets. Owing to the sequential pursuit strategy, we only need to assign one pursuer to one of the targets at every instant of time and the rest of the pursuers are free to go after other targets. What is left for us is to propose the pursuer assignment laws in order to shorten the time duration until the last target is captured.

2.2 Preliminaries

Given a finite number of distinct points in the Euclidean plane, called the *generators*, we associate their locations with a set of points in the plane, such that each point is closer (with respect to a given distance metric) to its own generator than to any other generator. The result is a tessellation of the plane into a set of regions associated with the given generators. If we use a Euclidean distance metric, this tessellation results in the *ordinary Voronoi diagram* (VD) generated by the given point set. The corresponding regions are called the *Voronoi cells* of the tessellation [109].

Given an (ordinary) Voronoi diagram of a point set in a generic configuration (that is, no three points are on the same line and no four points on the same circle), we may join all pairs of generators whose Voronoi cells share a common edge. We thus obtain a second tessellation consisting of only triangles, called the *Delaunay triangulation* (DT) of VD. The Delaunay triangulation is the dual graph of the Voronoi diagram.

A circle circumscribing any Delaunay triangle contains no generator in its interior [109]. This is the *Delaunay property*. Consider a triangulation of four points, as shown in Fig. 1(a). If the two triangles in the triangulation do not meet this property, we can change the triangulation into one that does (and hence construct the DT) simply by flipping the common edge (see Fig. 1(b)). Thus, given any triangulation of a given



(a) A triangulation of 4 points (b) DT of the same points

Figure 1: Flip-edge method for generating the Delaunay triangulation.

point set, we can construct the DT by flipping the edges until no triangle violates the

Delaunay property. This method of generating the DT of a given point set is called the *flip-edge method* [129].

When we deal with pursuer-target problems, in many cases we want to know the proximity relation between a set of agents, acting as pursuers, and a target on the plane. The problem of obtaining this proximity relation can often be recast as a set membership problem. For instance, the question of determining which of the agents is closest (in terms of arrival time) to a static target at a particular instant of time, reduces to a set membership problem, namely, one of forming the so-called Zermelo-Voronoi Diagram (ZVD) [9], and then finding the cell in which the target resides at the given time instant.

It is reminded that, given a finite number of agents at some time t, the ZVD is a partition of the plane whose generalized distance is the minimum time for the corresponding Zermelo navigation problem [159] from each agent's current position (the generator) to the agent's terminal configuration. Thus, every cell is the collection of all the points in the plane that can be reached by the associated agent faster than any other agents in the agent set. We state the precise definition of ZVD below.

Definition 2.1 (Zermelo-Voronoi Diagram [9]) *Given a set of n agents starting from distinct initial positions, whose dynamics are given by*

$$\dot{X}^i = u^i + w(X^i, t), \qquad X^i(0) = X^i_{P_0}, \tag{1}$$

where $X^i = [x^i, y^i]^\intercal \in \mathbb{R}^2$ denotes the position of the i^{th} agent, $u^i \in \mathbb{R}^2$ is the control input of the i^{th} agent and $w(X^i, t) \in \mathbb{R}^2$ represents the environmental disturbance (winds/currents), the Zermelo-Voronoi diagram *(ZVD)*[1] *(or Zermelo-Voronoi partition) is a set partition of the plane $\mathcal{Z} = \{Z_1, Z_2, ..., Z_n\}$ such that*

[1]Note that in [9] this is referred to as the dual Zermelo-Voronoi diagram, not to be confused with the dual graph of the Zermelo-Voronoi diagram.

i) $\mathbb{R}^2 = \bigcup_{i=1}^{n} Z_i$,

ii) For any point in Z_i*, the* i^{th} *agent will reach this point faster than any other agent.*

The sets Z_i *are the* Zermelo-Voronoi cells *for the partition.*

As is shown in [9], given any $Z_i, Z_j \in \mathcal{Z},\ i \neq j, i, j \in I$, we have $\text{int}(Z_i) \cap \text{int}(Z_j) = \varnothing$.

The following proposition characterizes a useful property of the ZVD that will be used later on.

Proposition 2.1 ([9]) *Let* $\mathcal{V} = \{V_i, i \in I\}$*, where* $I = \{1, 2, ..., n\}$*, be the partition of the ordinary Voronoi diagram with generators* $P = \{P_i, i \in I\}$*. Assume that the dynamics of each agent initially placed at the generator positions are given by (1), and assume that* $w = w(t)$*. Let the one-to-one, continuous function* $F : \mathbb{R}^2 \to \mathbb{R}^2$ *be defined by*

$$F(X) = f_{P_i}(X), \quad X \in V_i,\ i \in I, \tag{2}$$

where

$$f_{P_i}(X) = X + \int_0^{|X-P_i|} w(\tau)\, \mathrm{d}\tau, \qquad i \in I. \tag{3}$$

Then $Z_i = F(V_i)$ *and thus ZVD is the image of VD under the mapping* F*.*

In other words, there exists a homeomorphism between the ordinary Voronoi diagram and the Zermelo-Voronoi diagram with the same generators.

2.3 Problem Setup

In this section, we formulate the dynamic pursuit problem with multiple pursuers and one target. Extension to problems with multiple targets is straightforward, as can be seen later on. To this end, consider a group of n pursuers in the plane, denoted

by the index set $I = \{1, 2, ..., n\}$, and assume that at time $t = 0$ the pursuers are located at n distinct positions in the plane, designated by $P_0 = \{X^i_{P_0} \in \mathbb{R}^2, i \in I\}$. The kinematics of the i^{th} pursuer, $i \in I$, are described by

$$\dot{X}^i_P = u^i_P + w(t), \qquad X^i_P(0) = X^i_{P_0}, \tag{4}$$

where $X^i_P := [x^i_P, y^i_P]^\intercal \in \mathbb{R}^2$ denotes the position of the i^{th} pursuer, $u^i_P \in \mathbb{R}^2$ is the control input of the i^{th} pursuer such that $u^i_P \in \mathcal{U}_P$, for all $i \in I$, and $w(t) \in \mathbb{R}^2$ represents the wind disturbance. The set $\mathcal{U}_P$ consists of all piecewise continuous functions whose range is included in the set $U_P = \{u \in \mathbb{R}^2, |u| \leqslant \bar{u}\}$. It is assumed, furthermore, that there exists $0 < \bar{w} < \bar{u}$ such that

$$|w(t)| \leqslant \bar{w}, \tag{5}$$

for all $t \geqslant 0$. The restriction on the magnitude of the wind disturbance is imposed in order to ensure complete pursuer controllability, namely, that the pursuers are able to reach any point on the plane in finite time. The absence of controllability leads to complicated behavior and requires a more detailed analysis [13].

The objective of the pursuers is to intercept a target, whose kinematics is given by

$$\dot{X}_T = u_T, \qquad X_T(0) = X_{T_0}, \tag{6}$$

where $X_T = [x_T, y_T]^\intercal \in \mathbb{R}^2$ is the position of the target, and u_T is its control input such that $u_T \in \mathcal{U}_T$, which consists of all piecewise continuous functions whose range is included in the set $U_T = \{u \in \mathbb{R}^2, |u| \leqslant \bar{q}\}$. Note that the target is not affected by the wind field.

We assume that the pursuers do not know how the target maneuvers a priori. Instead, they have accurate measurements of the current position of the target at

every instant of time. One reasonable strategy for every pursuer is therefore to use the Zermelo navigation law [14, 23] in order to intercept the target, that is, at every instant of time, the pursuer approaches the target with the control law obtained by the solution of the corresponding Zermelo navigation problem, assuming that the target is stationary. This control law is optimal at $t = 0$ if the target remains stationary for all $t \geqslant 0$ [23]. As discussed in [9], starting at time $t = 0$, the optimal time of arrival T^i_{ZN} of the i^{th} pursuer from $X^i_{P_0}$ to X_{T_0} is given by

$$T^i_{\text{ZN}} = \min\{T > 0 : \bar{u}T - |X_{T_0} - X^i_{P_0} - \int_0^T w(\tau)\,\mathrm{d}\tau| = 0\} \tag{7}$$

Then the Zermelo's navigation control can be obtained by

$$u^i_{\text{ZN}} = \bar{u}(\cos\theta^*_i, \sin\theta^*_i)^\intercal, \tag{8}$$

where

$$\theta^*_i = \text{Arg}\left(X_{T_0} - X^i_{P_0} - \int_0^{T^i_{\text{ZN}}} w(\tau)\,\mathrm{d}\tau\right), \tag{9}$$

for $i \in I$.

Assume that, at every instant of time, only one pursuer is chasing the target, i.e., at every time $t \geqslant 0$, there exists only one $i \in I$, such that $u^i_P(t) = u^i_{\text{ZN}}(t)$ whereas $u^j_P(t) = 0,$ for all $j \in I, j \neq i$. We call i the *active pursuer* at time t. Our goal is to find a sequence of active pursuers to capture the target in the shortest possible time, under the assumptions listed above.

To this end, define a mapping $\sigma : [0, \infty) \mapsto I$, where σ belongs to the set of all the right continuous, piecewise constant functions, denoted as Σ, such that $\sigma(t) = i$ implies that, at time t, the i^{th} pursuer is the active pursuer. We call σ the *assignment function* [14].

Assume now that the sequence of active pursuers is given by $i_1, i_2, \ldots \in I$, along

with a sequence of times $0 < \tau_1 < \tau_2 < \ldots$ such that the corresponding assignment function is $\sigma(t) = i_1$ for $t = [0, \tau_1)$, $\sigma(t) = i_2$ for $t = [\tau_1, \tau_2)$, etc. For simplicity, we will also write $\sigma \equiv i$ to denote the fact that pursuer i was the only active pursuer. In this case, no switching takes place. When a pursuer assignment changes, we say that a *switching* occurs and the corresponding time τ_j is the *switching time*. Consequently, we can describe the dynamics of the sequential pursuit problem as the switched system

$$\dot{X}_P^{\sigma(t)} = u_{\mathrm{ZN}}^{\sigma(t)} + w(t), \quad X_P^{\sigma(\tau_j)}(\tau_j) = X_P^{\sigma(\tau_j)}(\tau_j -), \tag{10}$$

$$\dot{X}_P^k = w(t), \quad X_P^k(\tau_j) = X_P^k(\tau_j -), \quad k \in I \backslash \{\sigma(t)\}, \tag{11}$$

$$\dot{X}_T = u_T. \tag{12}$$

The initial conditions at $t = 0$ are given by $X_P^i(0) = X_{P_0}^i$, for all $i \in I$, and $X_T(0) = X_{T_0}$.

Let $\varphi(t; \sigma)$ be the solution of (10) for the given assignment function $\sigma \in \Sigma$. This is a piecewise continuous trajectory such that $\varphi(t; \sigma) = X_P^{\sigma(t)}(t)$ for all $t \geq 0$. Note that $\varphi(t; \sigma)$ may be discontinuous at $\tau_1, \tau_2, \ldots$. For some small enough $\epsilon > 0$, we define the *capture time* as $T_c(\sigma) := \inf\{T \in [0, +\infty) : |\varphi(T; \sigma) - X_T(T)| < \epsilon\}$.

The pursuer-target assignment problem can then be restated as follows: Given a target and a set of pursuers in the plane, determine an assignment function $\sigma_{\min} \in \Sigma$, such that the capture time is as small as possible, assuming that only the current position of the target is available. In other words, we seek an assignment function that will minimize capture time under the assumption that each pursuer is using a stroboscopic strategy [11, 52, 112] based on the target's current location.

2.4 Analysis and implementation of the pursuer-target assignment problem

Before we proceed with the solution of the optimal pursuer assignment problem, we first need to determine the condition on the target's maneuverability such that there

exists an assignment function leading to finite capture time. Below we provide a sufficient condition for the existence of capture time.

The *robust optimal line-of-sight* navigation law (ROLS) steers a pursuer towards a target at every instant of time, while maximizing the speed along the ensuing path. This is the optimal strategy, among all control strategies that force the pursuer to move along the current line-of-sight [11]. We can use this strategy to ensure capture as follows. First, let $Y^i(t) = X_T(t) - X_P^i(t)$ be the vector from the pursuer to the target. The ROLS navigation law of the i^{th} pursuer can be expressed as [11]

$$u^i_{\text{ROLS}}(t, Y^i) = \sqrt{\bar{u}^2 - \langle w(t), e_2^i(t)\rangle^2}\, e_1^i(t) - \langle w(t), e_2^i(t)\rangle e_2^i(t), \tag{13}$$

where

$$e_1^i(t) = \frac{Y^i(t)}{|Y^i(t)|}, \qquad e_2^i(t) = S e_1^i(t), \quad i \in I, \tag{14}$$

where S is the rotation matrix $\left[\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right]$.

The following result is adapted from [11].

Proposition 2.2 *Let $\epsilon > 0$, and assume that the dynamics of each pursuer is given by (4) and the dynamics of the target is given by (6). Then, for each pursuer i, and for all initial conditions $X^i_{P_0}$ and X_{T_0}, there exists a finite time $T^i_{\text{ROLS}}(X^i_{P_0}, X_{T_0}) \geqslant 0$ such that the i-th pursuer driven by the ROLS navigation law (13) enters the set $\{X \in \mathbb{R} : |X - X_T(T^i_{\text{ROLS}})| \leqslant \epsilon\}$, provided that*

$$|\langle w(t) - u_T(t), e_1^i(t)\rangle| < \sqrt{\bar{u}^2 - \bar{w}^2}, \tag{15}$$

for all $t \geqslant 0$, where $e_1^i(t)$ as in (14).

Proof. From equation (5) we have $\langle w(t), e_2^i(t)\rangle \leqslant |w(t)| \leqslant \bar{w}$, which implies that

$$\langle u^i_{\text{ROLS}}(t, Y), e_1^i(t)\rangle = \sqrt{\bar{u}^2 - \langle w(t), e_2^i(t)\rangle^2} \geqslant \sqrt{\bar{u}^2 - \bar{w}^2}. \tag{16}$$

Using the control (13) in the i^{th} pursuer dynamics, and subtracting it from the target dynamics (6), we get

$$\dot{Y}^i(t) = -u^i_{\text{ROLS}}(t, Y^i) - w(t) + u_T(t). \tag{17}$$

Since the i^{th} pursuer moves along the line-of-sight, using (17) along with (15), it follows that

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}|Y^i| &= \langle \dot{Y}^i, e^i_1(t)\rangle \\ &= -\langle u^i_{\text{ROLS}}(t, Y), e^i_1(t)\rangle - \langle w(t) - u_T(t), e^i_1(t)\rangle \\ &< -\sqrt{\bar{u}^2 - \bar{w}^2} < 0.\end{aligned}$$

Thus, the ROLS navigation law will drive the i-th pursuer to within an ϵ-ball of the target in finite time. □

The following corollary is therefore immediate from Proposition 2.2.

Corollary 2.1 *Assume that (15) holds for all $i \in I$. Any sequential pursuit strategy in which each active pursuer employs the ROLS navigation law (13) leads to capture of the target by at least one pursuer.*

Proposition 2.2 implies that the minimum-time intercept problem using Zermelo's navigation law in (8)-(9) always has a solution, for all initial conditions for the pursuers and the target. Furthermore, applying Zermelo's navigation law instead of (13) results in a smaller intercept time, that is, $T^i_{\text{ZN}} \leq T^i_{\text{ROLS}}$ for all $i \in I$. This, in turn, implies that a sequential strategy that uses Zermelo's navigation law for each pursuer will eventually lead to capture. By imposing a somewhat stronger condition we can actually prove the following result.

Proposition 2.3 *Let $\epsilon > 0$, and assume that the dynamics of each pursuer is given by (4) and the dynamics of the target is given by (6). Then, for each pursuer i, and*

for all initial conditions $X^i_{P_0}$ and X_{T_0}, there exists a finite time $T^i_{\mathrm{ZN}}(X^i_{P_0}, X_{T_0}) \geqslant 0$ such that the i-th pursuer driven by the robust Zermelo navigation law (8)-(9) enters the set $\{X \in \mathbb{R} : |X - X_T(T^i_{\mathrm{ZN}})| \leqslant \epsilon\}$, regardless of the evader's strategy and the form of external wind field, if and only if

$$\bar{q} < \bar{u} - \bar{w}, \tag{18}$$

where $\bar{q}, \bar{u}, \bar{w} \in \mathbb{R}$ represents the upper bound of target, pursuers, wind speed respectively.

proof 2.4.1 *To show (18) is the necessary condition, consider an extreme case where one pursuer P is playing against the target T under a constant wind drift $w = \bar{w}(X_{P_0} - X_{T_0})/|X_{P_0} - X_{T_0}|$, which is parallel to the line-of-sight between the initial positions of the pursuer and the evader. Under this condition, the optimal strategy of the two players are both to move along the initial line-of-sight with their respective maximum speeds. Therefore, if (18) is not satisfied, then the pursuer will not be able to capture the evader.*

On the other hand, assume that (18) is satisfied. Let $Y^i(t) = X_T(t) - X^i_P(t)$ be the vector from the i-th pursuer to the target. Assume that at time $t = t_k$ the i-th pursuer and the target are located at positions $X^i_P(t_k)$ and $X_T(t_k)$ respectively. It follows from (7) that the time-to-intercept of a stationary target at $X_T(t_k)$ is given by $T^i_{\mathrm{ZN}_k} = \min\{T > 0 : \bar{u}T - |Y^i(t_k) - \int_{t_k}^{t_k+T} w(\tau)\,\mathrm{d}\tau| = 0\}$. In particular,

$$\left| Y^i(t_k) - \int_{t_k}^{t_k + T^i_{\mathrm{ZN}_k}} w(\tau)\,\mathrm{d}\tau \right| = \bar{u} T^i_{\mathrm{ZN}_k}, \tag{19}$$

and the corresponding optimal control law at t_k is given by

$$u^i_{\mathrm{ZN}}(t_k) = \frac{1}{T^i_{\mathrm{ZN}_k}} \left(Y^i(t_k) - \int_{t_k}^{t_k + T^i_{\mathrm{ZN}_k}} w(\tau)\,\mathrm{d}\tau \right). \tag{20}$$

At the next time step $t = t_k + \delta t$, we can easily compute that

$$X_P^i(t_k+\delta t) = X_P^i(t_k) + \int_{t_k}^{t_k+\delta t} w(\tau)\,\mathrm{d}\tau + \frac{\delta t}{T^i_{\mathrm{ZN}_k}}\left(Y^i(t_k) - \int_{t_k}^{t_k+T^i_{\mathrm{ZN}_k}} w(\tau)\,\mathrm{d}\tau\right),$$

and, similarly, $X_T(t_k+\delta t) = X_T(t_k) + u_T^k \delta t$, where $u_T^k = u_T(t_k)$. Thus,

$$\begin{aligned} Y^i(t_k+\delta t) &= X_T(t_k+\delta t) - X_P^i(t_k+\delta t) \\ &= Y^i(t_k) + u_T^k\delta t - \int_{t_k}^{t_k+\delta t} w(\tau)\,\mathrm{d}\tau - \frac{\delta t}{T^i_{\mathrm{ZN}_k}} Y^i(t_k) + \frac{\delta t}{T^i_{\mathrm{ZN}_k}} \int_{t_k}^{t_k+T^i_{\mathrm{ZN}_k}} w(\tau)\,\mathrm{d}\tau. \end{aligned} \tag{21}$$

The time-to-intercept at time step $t = t_k + \delta t$ is given by $T^i_{\mathrm{ZN}_{k+1}} = \min\{T > 0 : \bar{u}T - |Y^i(t_k+\delta t) - \int_{t_k+\delta t}^{t_k+\delta t+T} w(\tau)\,\mathrm{d}\tau| = 0\}$. In particular,

$$|Y^i(t_k+\delta t) - \int_{t_k+\delta t}^{t_k+\delta t+T^i_{\mathrm{ZN}_{k+1}}} w(\tau)\,\mathrm{d}\tau| = \bar{u}T^i_{\mathrm{ZN}_{k+1}}. \tag{22}$$

At this point, pick $\delta t = \epsilon/(\bar{u}+\bar{w})$, where $\epsilon > 0$ and assume that there exists $k > 0$ such that $T^i_{\mathrm{ZN}_k} \le \delta t$, then from (20) one obtains

$$\begin{aligned} |Y^i(t_k)| &= |u^i_{\mathrm{ZN}}(t_k)T^i_{\mathrm{ZN}_k} + \int_{t_k}^{t_k+T^i_{\mathrm{ZN}_k}} w(t)dt| \\ &\le |u^i_{\mathrm{ZN}}(t_k)|T^i_{\mathrm{ZN}_k} + |\bar{w}\int_0^{T^i_{\mathrm{ZN}_k}} 1dt| \\ &\le (\bar{u}+\bar{w})T^i_{\mathrm{ZN}_k} \le (\bar{u}+\bar{w})\delta t \\ &\le (\bar{u}+\bar{w})\frac{\epsilon}{\bar{u}+\bar{w}} = \epsilon. \end{aligned}$$

This shows that the pursuer is in the ϵ ball centered at the evader's position. In other words, the capture has occurred.

Suppose now that $T^i_{\mathrm{ZN}_k} > \delta t$ for all $k > 0$. In this case, using (21), and after some algebraic manipulations, one obtains the following expression for the term in

the left-hand-side of (22)

$$Y^i(t_k+\delta t) - \int_{t_k+\delta t}^{t_k+\delta t+T^i_{\mathrm{ZN}_{k+1}}} w(\tau)\,\mathrm{d}\tau = \big(1-\frac{\delta t}{T^i_{\mathrm{ZN}_k}}\big)\Big(Y^i(t_k) - \int_{t_k}^{t_k+T^i_{\mathrm{ZN}_k}} w(\tau)\,\mathrm{d}\tau\Big) - \int_{t_k+T^i_{\mathrm{ZN}_k}}^{t_k+\delta t+T^i_{\mathrm{ZN}_{k+1}}} w(\tau)\,\mathrm{d}\tau + u_T^k\delta t.$$

Let

$$\Theta_k = Y^i(t_k) - \int_{t_k}^{t_k+T^i_{\mathrm{ZN}_k}} w(\tau)\,\mathrm{d}\tau, \tag{23}$$

and

$$H_k = -\int_{t_k+T^i_{\mathrm{ZN}_k}}^{t_k+\delta t+T^i_{\mathrm{ZN}_{k+1}}} w(\tau)\,\mathrm{d}\tau + u_T^k\delta t. \tag{24}$$

Then (22) can be written as follows

$$|\alpha\Theta_k + H_k| = \bar{u}T^i_{\mathrm{ZN}_{k+1}}, \tag{25}$$

where $\alpha = 1-\delta t/T^i_{\mathrm{ZN}_k} > 0$*, whereas (19) can be written as follows*

$$|\Theta_k| = \bar{u}T^i_{\mathrm{ZN}_k}. \tag{26}$$

Subtracting (25) from (26) yields

$$-\bar{u}\Delta T_k = |\Theta_k| - |\alpha\Theta_k + H_k|, \tag{27}$$

where $\Delta T_k = T^i_{\mathrm{ZN}_{k+1}} - T^i_{\mathrm{ZN}_k}$.

We claim that $\Delta T_k < 0$ *for all* $k > 0$ *such that* $T^i_{\mathrm{ZN}_k} > \delta t$*. To this end, note that using the triangle inequality, (27) yields* $-\bar{u}\Delta T_k \geq |\Theta_k| - \alpha|\Theta_k| - |H_k| = \bar{u}\delta t - |H_k|$*, where we have made use of the fact that* $(1-\alpha)|\Theta_k| = \delta t/T^i_{\mathrm{ZN}_k}|\Theta_k| = \bar{u}\delta t$*. Hence,* $|H_k| \geq \bar{u}(\Delta T_k + \delta t)$.

From (24) we also have

$$\begin{aligned}|H_k| &= \left| u_T^k \delta t - \int_{t_k+T^i_{\mathrm{ZN}_k}}^{t_k+\delta t+T^i_{\mathrm{ZN}_{k+1}}} w(\tau)\,\mathrm{d}\tau \right| \\ &\leq |u_T^k|\delta t + \left| \int_{t_k+T^i_{\mathrm{ZN}_k}}^{t_k+\delta t+T^i_{\mathrm{ZN}_{k+1}}} w(\tau)\,\mathrm{d}\tau \right| \\ &\leq \bar{q}\delta t + \bar{w}|\Delta T_k + \delta t|.\end{aligned}$$

Thus, we get

$$\bar{u}(\Delta T_k + \delta t) \leq |H_k| \leq \bar{q}\delta t + \bar{w}|\Delta T_k + \delta t|. \tag{28}$$

If $\Delta T_k \geq 0$, for some $k > 0$ such that $T^i_{\mathrm{ZN}_k} > \delta t$, then it follows from the previous expression that $\bar{u}(\Delta T_k + \delta t) \leq \bar{q}\delta t + \bar{w}(\Delta T_k + \delta t)$ or that

$$(\bar{w} - \bar{u})\Delta T_k \geq (\bar{u} - \bar{q} - \bar{w})\delta t, \tag{29}$$

which leads to a contradiction since the left-hand side of inequality (29) is non-positive and the right-hand side is positive. It follows that $T^i_{\mathrm{ZN}_{k+1}} - T^i_{\mathrm{ZN}_k} = \Delta T < 0$. This implies that the sequence $\{T^i_{\mathrm{ZN}_k}\}_{k=1}^{\infty}$ is strictly decreasing, and since it is also bounded from below, it converges. Hence, $\lim_{k\to\infty} \Delta T_k = 0$. Taking the limit as $k \to \infty$ of (28) yields $\bar{u}\delta t \leq (\bar{q} + \bar{w})\delta t$ or that $\bar{u} \leq \bar{q} + \bar{w}$, contradicting (18).

The next corollary follows immediately from the previous proposition.

Corollary 2.2 *Assume that (18) holds for all $i \in I$. Any sequential pursuit strategy in which each active pursuer employs the robust Zermelo's navigation law (8)-(9) leads to capture of the target by at least one pursuer.*

Under the assumption of sequential (or relay) pursuit, and assuming that each active pursuer chooses the Zermelo navigation law, we may now propose the following

algorithm to assign the active pursuer:

Dynamic Assignment of Active Pursuer with Multiple Pursuers and One Target

a) Construct the ZVD and assign the i^{th} pursuer to be the active pursuer if the target resides in the corresponding Zermelo-Voronoi cell Z_i.

b) At every time step, update the ZVD and assign the j^{th} pursuer to be the active pursuer if the target resides in the corresponding Zermelo-Voronoi cell Z_j.

c) Check the distance between the target and the active pursuer and repeat step b) if the distance is bigger than ϵ. Otherwise, terminate the procedure and return the assignment function.

2.5 Update algorithm to dynamically generate the ZVD

At every instant of time, we need the knowledge of the ZVD in order to determine which cell the target resides in. In order to do this, we can either build a ZVD from scratch at each time, or update the ZVD from the previous time step. Since at every time interval, only one generator is moving relatively to the rest, it is reasonable to expect that it will be more efficient to update the ZVD from the previous time step. Hereby, we present an algorithm that updates the ZVD from one time step to the next time step when a single pursuer has moved[2].

From (2), we know that there exists an invertible, continuous transformation between the ordinary VD and the ZVD. Thus, our strategy for updating the ZVD is to update the ordinary VD corresponding to the same generators first and then form the ZVD through this transformation.

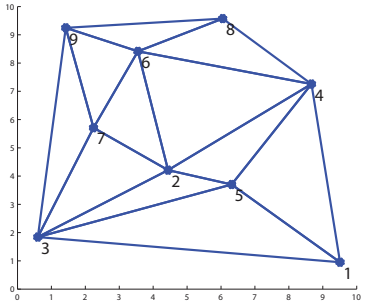
[2]This algorithm can also be applied to the case where more than one generator is moving.

In order to update the ordinary VD we will, instead, update its dual graph, namely, its Delaunay Triangulation. There exist several algorithms for updating DT in the literature of computational geometry [34, 51, 79, 92]. We will use a modification of the algorithm introduced in [51] since it is relatively efficient and it fits our problem.

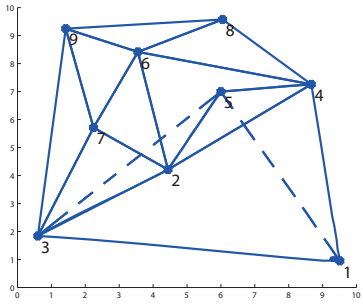
In order to update the DT from the previous time step to the current time step, a straightforward way would be to put all the points in a queue and every time we push a point out from the queue, we remove this point from the original triangulation and then insert it back at the new location at the present time [69]. Each deletion and insertion of the DT preserves the Delaunay property, so the procedure would yield a valid DT. However, the procedure is not very efficient since even if all the points remain static during the time interval, we still need to delete and insert all the points to complete the update. Moreover, removing a point from a DT is a fairly expensive process. As shown in [37], the algorithm to reconstruct the Delaunay Triangulation after a point is removed is quadratic in the number of its neighbors.

Given the previous considerations, we propose an alternative approach to deal with moving generators. We want the update algorithm to take advantage of the fact that part of the DT structure has not changed from the previous time step. To this end, denote by DT_k, and DT_{k+1} the Delaunay Triangulation at time steps t_k and t_{k+1} respectively. Assume that the corresponding generator sets are given by P_k and P_{k+1}. Our goal is to update DT_k into DT_{k+1} with as few deletions as possible. To this end, we want to check first if we can generate DT_{k+1} from DT_k using only the flip-edge method. The flip-edge method can be applied when DT_k is an embedding [51]. Recall that, given a point set, a triangulation is an *embedding* if the triangulation associated with this point set has no overlapping triangles. If a triangulation is not an embedding, we say that it is an *unembedding*. Fig. 2(a) shows a DT associated with a given point set, and Fig. 2(b) shows the DT associated with a new point set, where point 5 has changed its location. Some triangles overlap with each other in Fig. 2(b).

Thus, the DT in Fig. 2(b) is an unembedding. Also notice that if an unembedding occurs, there exists at least one triangle that has changed its orientation. For example, the triangle with vertices 3, 4, and 5 in Fig. 2(a) has a clockwise orientation. In Fig. 2(b), on the other hand, the orientation of the triangle with the same vertices has counter-clockwise orientation, i.e., its orientation has changed.



(a) Delaunay Triangulation of 9 generators.



(b) Generator 5 changed its location and caused unembedding.

Figure 2: Unembeding caused by relocation of a generator.

We introduce the *orientation certificate* to check whether DT_k is an embedding or not [51]. Given a triangle $[i, j, k]$ in the triangulation, where i, j, k denote the indices of the generators, define

$$\mathrm{CCW}(p, q, r) = \begin{vmatrix} x_p & y_p & 1 \\ x_q & y_q & 1 \\ x_r & y_r & 1 \end{vmatrix}. \tag{30}$$

Let $\{P_i, P_j, P_k\}$ be the location of the vertices of the triangle $[i, j, k]$ at the previous time step, and let $\{P'_i, P'_j, P'_k\}$ be the position of the vertices of the same triangle at the current time. Then the orientation certificate of this triangle is passed if $\mathrm{CCW}(P_i, P_j, P_k)$ and $\mathrm{CCW}(P'_i, P'_j, P'_k)$ have the same sign. The orientation certificate is passed for a triangulation if the certificate is passed for all its triangles.

If the orientation certificate is passed, we can simply use the flip-edge method to

update the DT. Otherwise, we need to remove the points that cause the unembedding and then check the orientation certificate until it is passed. After this iteration, we obtain a triangulation with no overlaps, and we can then use the flip-edge method to transform it into a DT. Finally, we insert the removed points to their current locations and form DT_{k+1}. The difference between this procedure and the method introduced in [51] is that in the latter reference the authors remove the points randomly to get the triangulation candidate, whereas in our case we only remove the moving points since they are the only possible generators that may cause unembedding.

The algorithm for updating the Zermelo-Voronoi diagram from the previous time step to current time step is given in Algorithm 1.

Algorithm 1 Update Zermelo Voronoi Diagram

Input: Coordinates P_{k-1} of the generators at the previous time step and the corresponding Delaunay triangulation DT, coordinates P_k of point set at current time step.

Output: Updated Zermelo Voronoi Diagram and Delaunay Triangulation at the current time step.

procedure UPDATE_DT(DT,P_{k-1},P_k)
  while the triangulation DT is not embedded under current coordinates P_k **do**
    Update DT by removing one of the points that cause the unembedding (in our case the active pursuer);
    store the current coordinates of removed points into set R;
  end while
  if R is not empty **then**
    flip the remaining triangulation into a Delaunay triangulation;
  end if
  for $i = 0$ to $length(R)$ **do**
    Update DT by inserting the i^{th} point in R into the triangulation;
  end for
  Transform DT into an ordinary Voronoi diagram VD;
  Transform VD into the ZVD at current time through the coordinate transformation (2).
  return ZVD and DT.
end procedure

To remove a point from the standard Delaunay Triangulation, we use the deletion

method introduced in [33] and to insert a point into the DT, we choose the algorithm given in [69]. Both these algorithms have complexity $O(n\log n)$. The flip-edge algorithm introduced in [31] has worst case complexity $O(n^2)$, but in practice, it is much faster.

2.6 Sequential Pursuit of Multiple Targets

In this section, we deal with the dynamic pursuit problem with multiple pursuers and targets. Consider a group of n pursuers in the plane, denoted by $\{P_1, P_2, ..., P_n\}$, and m targets, denoted by $\{E_1, E_2, ..., E_m\}$. The kinematics of the i^{th} pursuer, $i \in I$, are described by (4).

The objective of the pursuers is to intercept some targets. It is assumed in this chapter that after a pursuer intercepts a target, he is still capable of going after other targets. The cases where each pursuer can only capture one target (e.g. missiles) can be discussed in follow-up work. The kinematics of the j^{th} target is given by

$$\dot{X}_T^j = u_T^j, \qquad X_T^j(0) = X_{T_0}^j, \tag{31}$$

where $X_T^j = [x_T^j, y_T^j]^\intercal \in \mathbb{R}^2$ is the position of the j^{th} target, and u_T^j is its control input such that $u_T^j \in \mathcal{U}_T^j$, which consists of all piecewise continuous functions whose range is included in the set $U_T^j = \{v \in \mathbb{R}^2, |v| \leqslant \bar{q}_j\}$. Note that the targets are not affected by the wind field.

At this point, we are ready to extend our pursuer assignment algorithm in the multiple-pursuers-one-target case to the problem with multiple targets. In particular, we present two different methods to assign the pursuers to the targets. The first method is given as follows.

Dynamic Assignment of Active Pursuer with Multiple Pursuers and Targets - Method I

a) Construct the ZVD and for every target in the target set, pair the i^{th} pursuer to this target if the target resides in the corresponding Zermelo-Voronoi cell Z_i. When one pursuer is paired to multiple targets, it always goes after the target whose current location can be reached by the pursuer in the shortest time.

b) At every time step, update the ZVD and the locations of the pursuers and the targets. For each target, pair the j^{th} pursuer to it if the target resides in the corresponding Zermelo-Voronoi cell Z_j. Again, if one pursuer is paired to multiple targets at some time step, the pursuer will be assigned to the target whose current location can be reached by the pursuer in the shortest time under Zermelo's control law.

c) Check the distance between each target and its assigned pursuer and terminate the procedure if all the distances are smaller than ϵ. Otherwise, remove the targets whose distances are smaller than ϵ from the target set and return to step b).

In the previous algorithm, we simply pair each target to its closest pursuer, in the sense that this pursuer can reach the current location of the target faster than any other pursuer. This method may cause one pursuer to be assigned with multiple targets, and some targets may not be actively pursued until other targets have been captured. When the number of pursuers is larger than the number of targets, we can avoid the situation by applying the following algorithm.

Dynamic Assignment of Active Pursuer with Multiple Pursuers and Targets - Method II

a) Construct the ZVD from the pursuer set.

b) Pair each target with the i^{th} pursuer if the target resides in the corresponding Zermelo-Voronoi cell Z_i.

c) For each pursuer paired with only a single target, assign this pursuer to that target. If one or more pursuers are paired with multiple targets, for each of these pursuers, assign it to the corresponding paired target whose current location can be reached by the pursuer in the shortest time and discard the pairs between this pursuer and the rest of the targets, and go to d). Otherwise, assign every pursuer to its paired target and go to e).

d) Generate a new ZVD Z' form the pursuer set excluding those pursuers that are already assigned to targets and for every target that has not been paired, pair it with the j^{th} pursuer if the target resides in the corresponding Zermelo-Voronoi cell Z'_j. Go to c).

e) At every time step, update the ZVD with respect to the full pursuer set. Update the locations of the pursuers and the targets. Check the distance between each target and its assigned pursuer and if all the distances are smaller than ϵ, terminate the procedure. Otherwise, remove the targets whose distances with its assigned pursuers are smaller than ϵ from the target set and return to step b).

Both algorithms have their advantages and disadvantages. The first algorithm does not have a restriction on the number of pursuers, but when the number of pursuers is relatively large, multiple targets may still be assigned to one pursuer and the unassigned pursuers are not utilized to reduce the time-to-capture of all the targets. For the second algorithm, each evader is chased by a pursuer at every instant of time in order to reduce the time-to-capture of all the targets under our assumption of sequential pursuit.

The following corollary gives us a condition for our multiple-pursuer-multiple-target assignment schemes to terminate in finite time.

Corollary 2.3 *Assume that*

$$max_{j\in\{1,\ldots,m\}}(\bar{q}_j) < \bar{u} - \bar{w}, \tag{32}$$

then any sequential pursuit strategy in which each active pursuer employs the robust Zermelo's navigation law (8)-(9) leads to capture of all the targets by at least one pursuer.

Proof. Since $max_{j\in\{1,\ldots,m\}}(\bar{q}_j) < \bar{u} - \bar{w}$, it follows that $\bar{q}_j < \bar{u} - \bar{w}$, for all $j \in \{1, \ldots, m\}$. By Corollary 2.2, each evader can be captured by at least one pursuer in finite time. Therefore, for a finite number of evaders, our sequential pursuit schemes can be terminated in finite time. □

2.7 Simulation Results

In this section, we first present an example of multiple-pursuers-one-target problem to illustrate the update of the ZVD and the active pursuer along the time. We then deal with the problems with multiple pursuers and targets by applying the two methods we proposed earlier to show how our algorithms work and make a comparison between the two proposed alternatives.

We consider a scenario with multiple pursuers and one target. The target is moving in a straight line according to equation (6), where $u_T(t) = [-0.4, -0.5]^\intercal$. Assume that there exist 12 pursuers, which are initially located at distinct positions determined by P_0. The wind field that affects the pursuers is given by

$$w(t) = \begin{bmatrix} -0.2 - 0.2\cos(t) \\ 0.3 \end{bmatrix}.$$

Figure 3-5 illustrates the trajectories of the pursuers in the wind and the moving target. Specifically, Fig. 3 shows the ZVD formed by the pursuers at $t = 0$. As seen

in this figure, $i = 4$ is the active pursuer since the target falls in the Zermelo-Voronoi cell of X_P^4. Figure 4 illustrates the trajectories of the target and the pursuers in the time interval $[0, \tau_1]$, where $\tau_1 = 2.6$ is the switching time. The Zermelo-Voronoi Diagram at $t = \tau_1$ is also presented here to show that the target is about to leave the Zermelo-Voronoi cell of X_P^4 and enter another cell. Figure 5 illustrates the trajectories of the target and the pursuers from $t = \tau_1$ to capture time $T_c = 5.0$, as well as the Zermelo-Voronoi Diagram at $t = T_c$. In this last time interval the target is assigned to $i = 5$.

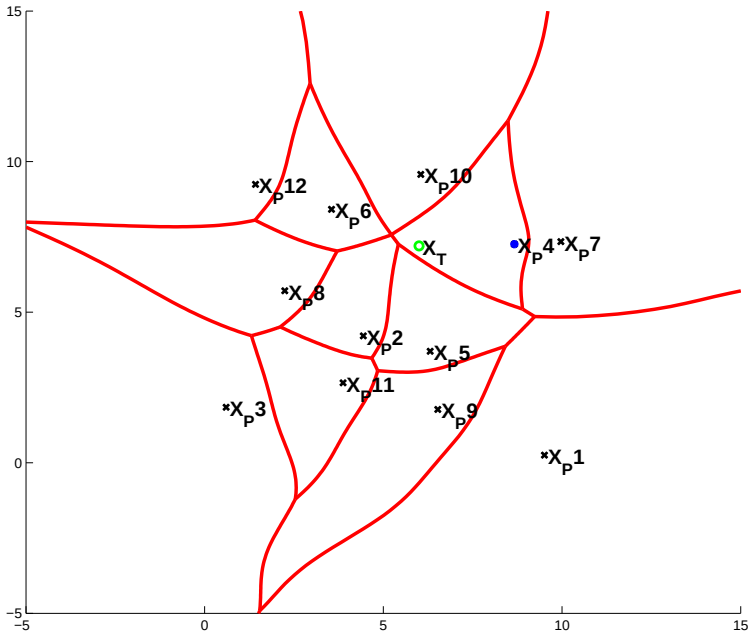


Figure 3: Zermelo-Voronoi Diagram formed by pursuers at $t = 0$, X_P^4 is the active pursuer

For comparison, note that when only one pursuer tries to capture the target, the shortest possible time is $T_c = 7.5$. In that case there is only a single active pursuer, namely, X_P^4.

Next, we deal with the problem with 12 pursuers and 3 evaders. The initial positions of the targets are given by $X_{T_0}^1 = [3; 5], X_{T_0}^2 = [6; 7], X_{T_0}^3 = [4.5; 6]$. Each

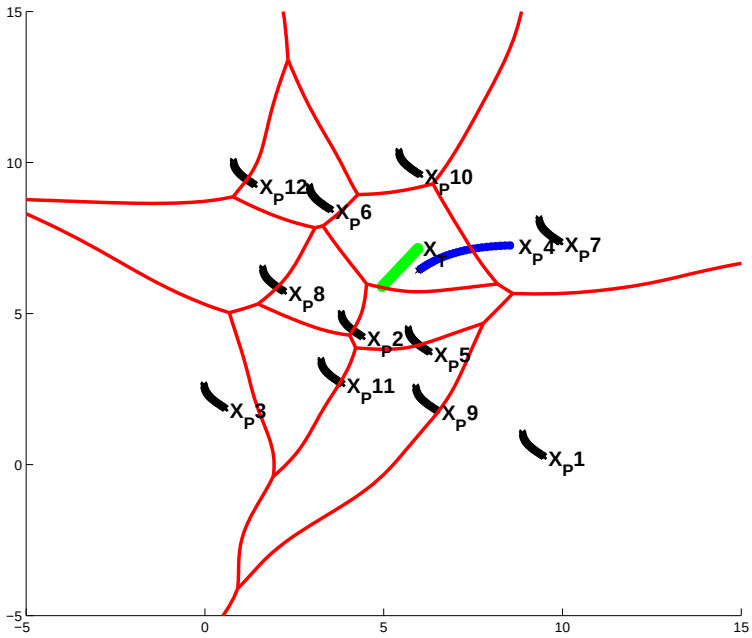


Figure 4: Zermelo-Voronoi Diagram formed by pursuers at the first switch time $t = 2.6$, and trajectories of pursuers and target for $t \in [0, 2.6)$.

target moves in a straight line with velocity $[0.4, 0.5]^\intercal$. The wind field that affects the pursuers is the same as in the previous example.

Figures 6 through 10 depict the trajectories of the pursuers and targets when the first algorithm is applied. Figure 6 illustrates the initial pairing between the pursuers and the evaders. In particular, Target 1 and 3 are paired with Pursuer 2 and Target 2 is paired with Pursuer 4. Pursuer 2 goes after Target 1 in the beginning since the time it takes for Pursuer 2 to arrive at the location where Target 1 resides in is smaller than the time to arrive at the initial location of Target 3. Pursuer 4 chases Target 2 since it is the only target paired with this pursuer. At time $s_1 = 0.70$, Target 3 switch its pair from Pursuer 2 to 5, because it enters the Zermelo-Voronoi cell of Pursuer 5 at this time step. The trajectories of the players in the time interval $[0, s_1]$ are shown in Fig. 7 . At time $s_2 = 1.35$, the original Target 1 is captured by Pursuer 2 and removed from the target set. The trajectories of the pursuers and targets in the time

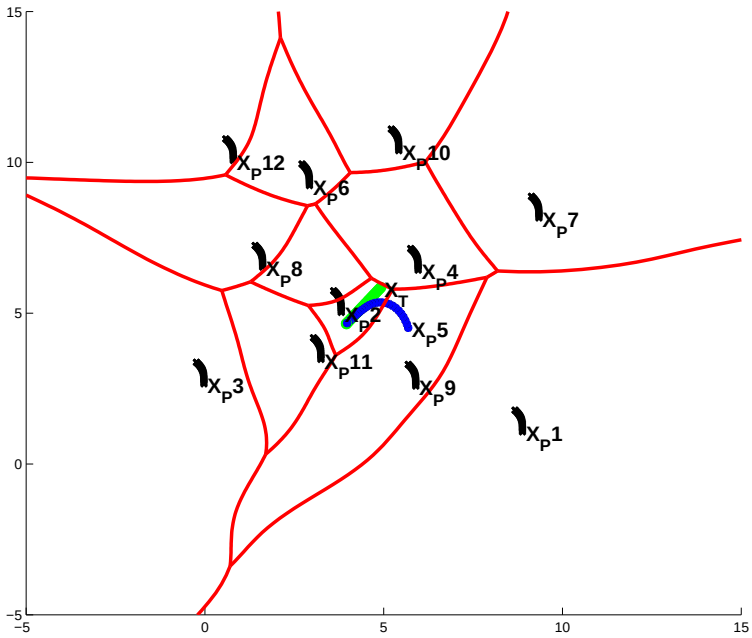


Figure 5: Zermelo-Voronoi Diagram formed by pursuers at the capture time $T_c = 5.0$, and trajectories of pursuers and target for $t \in [2.6, 5.0]$, X_P^5 is the active pursuer.

interval $[s_1, s_2]$ are shown in Fig. 8. Note that the targets are renumbered at this point since one of the targets is captured at s_2. One of the two remaining targets is captured at time $s_3 = 1.70$ and the corresponding trajectories in $[s_2, s_3]$ are shown in Fig. 9 . Figure 10 depicts the trajectories of the pursuers and the last evading target from s_3 to $s_4 = 3.90$, when the last target is captured by Pursuer 5.

The pursuit process under the second algorithm is presented in Fig. 11 to 14. As shown in Fig. 11, Pursuer $2, 4$ and 5 are assigned to Target $1, 2$ and 3, respectively, at $t = 0$. Both Target 1 and 3 are in the Zermelo-Voronoi cell of Pursuer 2 in the beginning and Pursuer 2 is assigned with Target 1 since it can reach the location of Target 1 faster than that of Target 3 through Zermelo's navigation law. Hence, Target 3 needs to be assigned to a pursuer in the pursuer set excluding Purser 2 and 4 by the rule in the algorithm, which leads to Pursuer 5. The original Target 1 is captured by Pursuer 2 at $t_1 = 1.35$. The trajectories of the players in $[0, t_1]$ are illustrated in

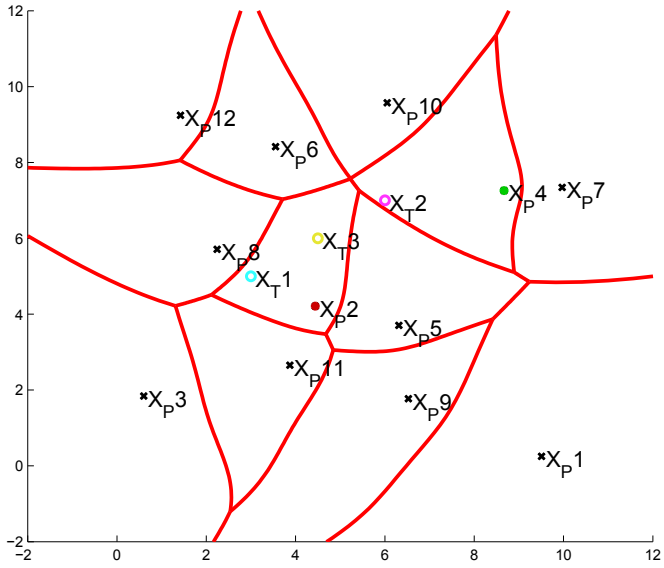


Figure 6: Pursuers and evaders at $t = 0$. Pursuer 2 is paired with Target 1 and 3. Purser 4 is paired with Target 2. Red curves represents the ZVD at $t = 0$.

Fig. 12. The second target is captured by Pursuer 4 at $t_2 = 1.70$ and the plot in the time interval $[t_1, t_2]$ is depicted in Fig. 13. Fig. 14 shows the plot of the time period between t_2 and $t_3 = 3.05$, when the last target is captured by Pursuer 5.

Comparing the results by applying the two algorithms, we can see that the second algorithm takes less time to capture all the targets. The reason is that since we assign a single active pursuer to each of the targets in the second algorithm, the 5th pursuer who captured the 3rd target in the end is already assigned to this target at the start of the pursuit process, whereas in the case of the first algorithm, the 3rd target is paired with the 2nd pursuer in the beginning and it was not actively pursued by any pursuer in a short time period until it enters a Zermelo-Voronoi cell of an inactive pursuer. Notice, however, that there is a possibility for more pursuer to be assigned in the second algorithm without reducing the time duration to capture all the targets. In this case, the fuels spent by the extra pursuers are essentially wasted.

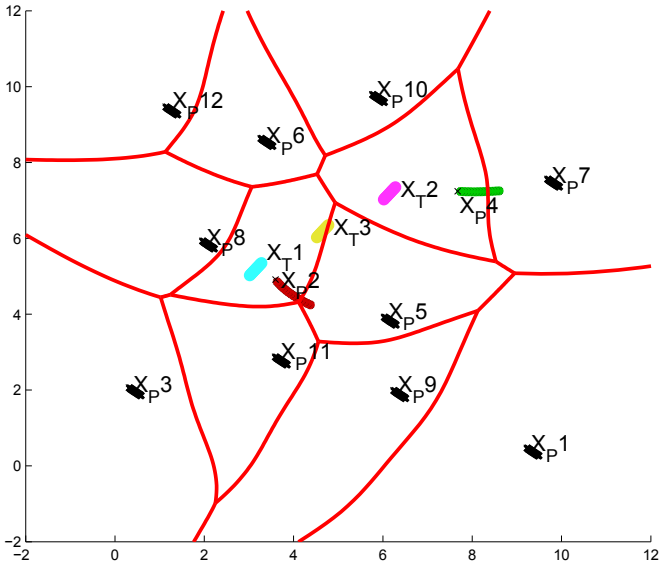


Figure 7: Trajectories of pursuers and evaders in $[0, s_1]$. Pursuer 2 goes after Target 1 during this period. Target 3 enters the Zermelo-Voronoi cell of Pursuer 5 at this time step and it will be paired with Pursuer 5 henceforth. ZVD at time s_1 is depicted in red curves.

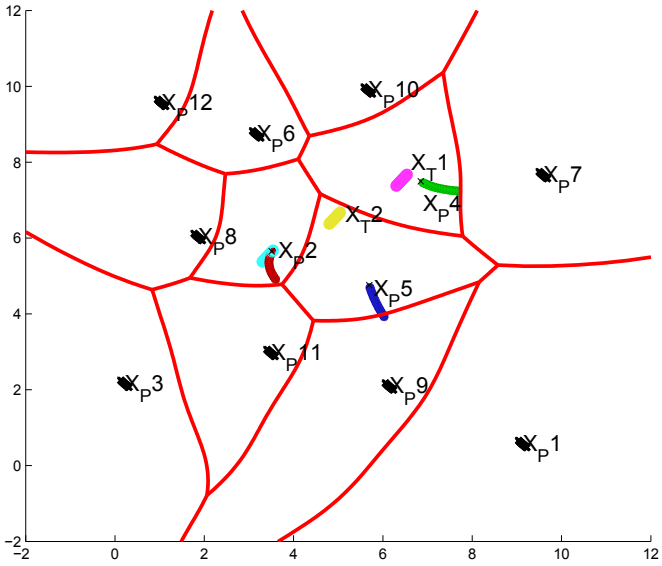


Figure 8: Trajectories of players in $[s_1, s_2]$. The target to the left in cyan is captured by Pursuer 2 at time s_2. Red curves represents the ZVD at s_2.

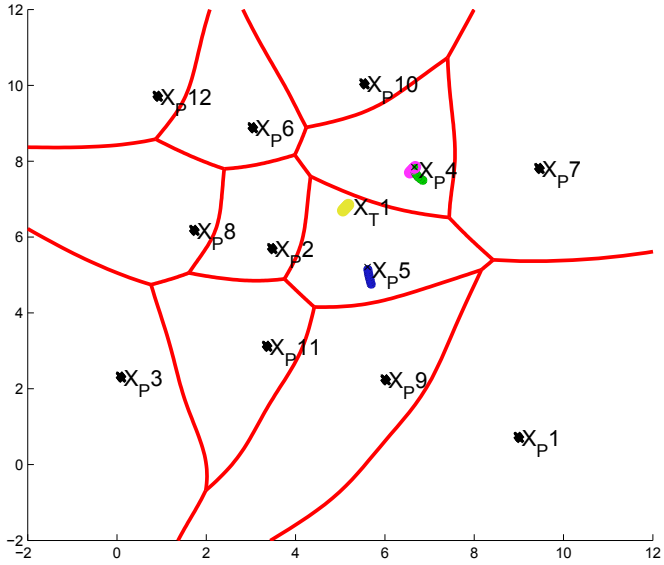


Figure 9: Trajectories of players in $[s_2, s_3]$. The target to the right in magenta is captured by Pursuer 4 at time s_3. Red curves represents the ZVD at s_3.

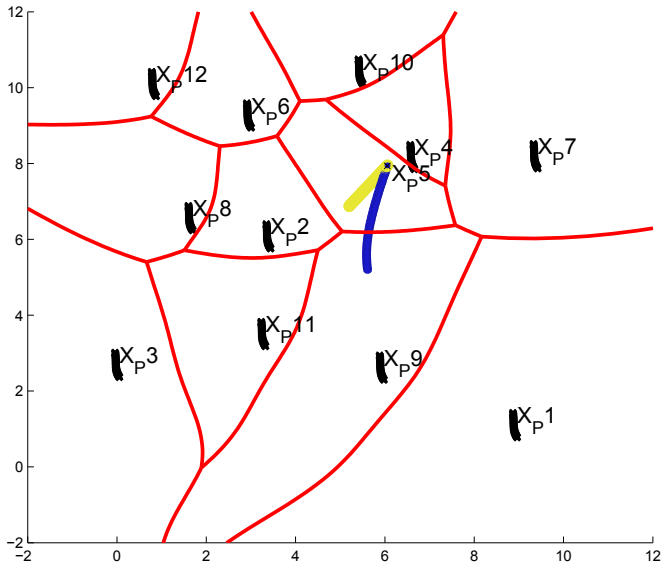


Figure 10: Trajectories of players in $[s_3, s_4]$. The only target left is captured by Pursuer 5 at time s_4. Red curves represents the ZVD at the terminal time s_4.

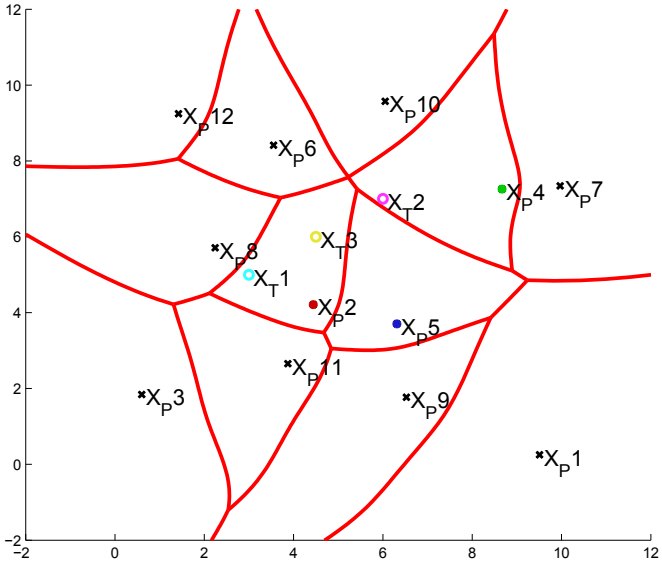


Figure 11: Pursuers and evaders at $t = 0$. Pursuer $2, 4$ and 5 are assigned to Target $1, 2$ and 3, respectively. Each target and its corresponding active pursuer are depicted by opposite colors for easier recognition, e.g., Target 3 is in yellow and Pursuer 5 is in blue. Red curves represents the ZVD at $t = 0$.

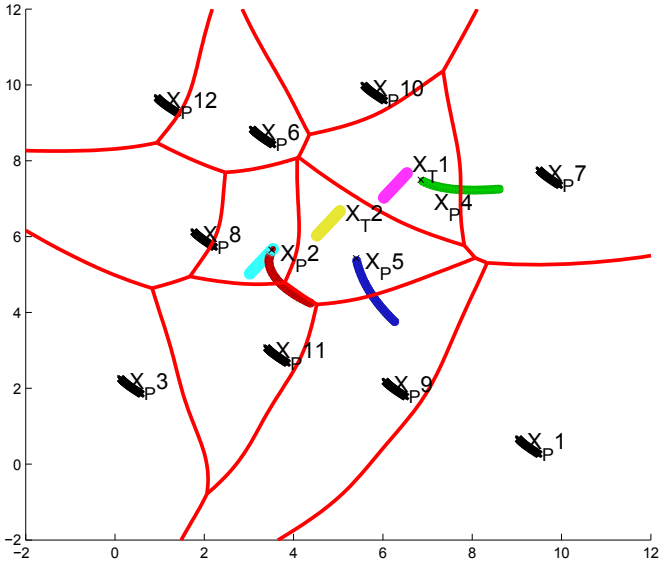


Figure 12: Trajectories of players in $[0, t_1]$. The target to the left in cyan is captured by Pursuer 2 at time t_1. Red curves represents the ZVD at t_1.

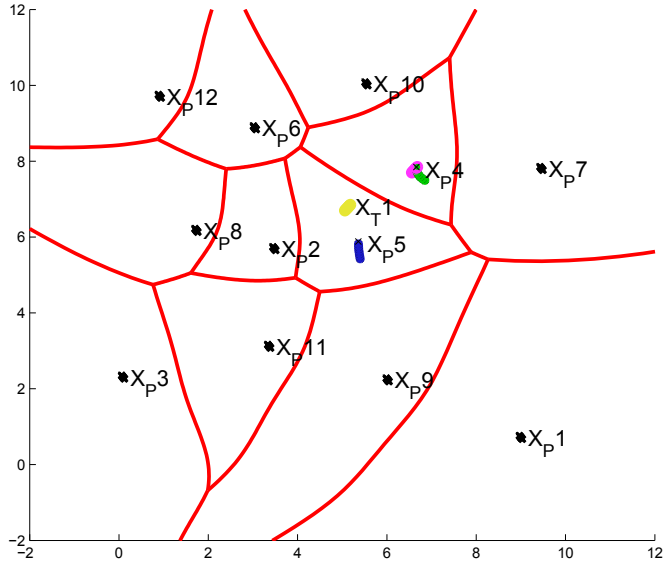


Figure 13: Trajectories of players in $[t_1, t_2]$. The target to the right in magenta is captured by Pursuer 4 at time t_2. Red curves represents the ZVD at t_2.

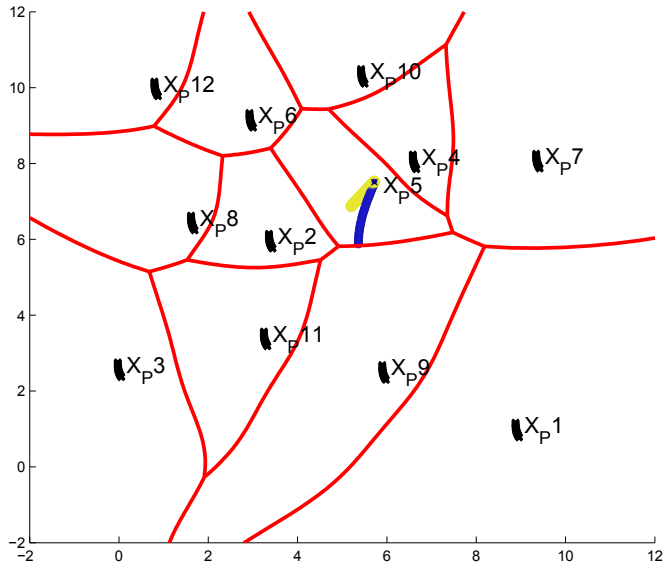


Figure 14: Trajectories of players in $[t_2, t_3]$. The only target left is captured by Pursuer 5 at time t_3. Red curves represents the ZVD at the terminal time t_3.

CHAPTER III

PURSUIT-EVASION GAMES IN LINEAR FLOW FIELDS

3.1 Introduction

In this Chapter, we consider the differential game of pursuit and evasion between two players on a plane under an external flow field. It is assumed that the pursuer and the evader move with constant but different speeds, and they are both agile, that is, they are allowed to change their bearings instantaneously. To simplify the analysis, it will be assumed that the flow field is approximated by a time-invariant, spatially-affine function. Our goal is to find the region of initial conditions of both players that leads to capture when both players act optimally, and derive the corresponding optimal strategies of the two players when capture is guaranteed.

3.2 Problem Formulation

3.2.1 Problem Setup

Consider a pursuer and an evader moving on a plane under the influence of an external flow field. Both players are agile, which means that they are allowed to alter the direction of their velocity vector instantaneously. The goal of the pursuer is to capture the evader in finite time, whereas the objective of the evader is to avoid interception indefinitely. Capture (termination of the game) occurs when the distance between the pursuer and the evader is less than a constant ℓ. The pursuer and the evader move with constant speeds, denoted by v_P and v_E, respectively. In this work, we will assume that $v_P > v_E$. The equations of motion for the pursuer and the evader in the

inertial reference frame are thus given by

$$\dot{x}_P = v_P \cos\phi + w_1(x_P, y_P), \tag{33}$$

$$\dot{y}_P = v_P \sin\phi + w_2(x_P, y_P), \tag{34}$$

$$\dot{x}_E = v_E \cos\psi + w_1(x_E, y_E), \tag{35}$$

$$\dot{y}_E = v_E \sin\psi + w_2(x_E, y_E), \tag{36}$$

where (x_P, y_P) and (x_E, y_E) denote the positions of the pursuer and the evader, respectively, $\phi, \psi \in [-\pi, \pi]$ are the control of the pursuer and the evader, respectively, v_P and v_E represent the maximum speed of the pursuer and the evader, and $w_1(\cdot,\cdot)$ and $w_2(\cdot,\cdot)$ are the components of an external spatially varying flow field along x-axis and y-axis, respectively.

3.2.2 Differential Game Formulation in the Reduced Space

In order to simplify the analysis, it will be assumed that the external flow field is approximated by time-invariant affine functions. Specifically, let

$$w_1(x, y) = \alpha_1 x + \beta_1 y + \gamma_1, \tag{37}$$

$$w_2(x, y) = \alpha_2 x + \beta_2 y + \gamma_2, \tag{38}$$

where α_i, β_i, $\gamma_i \in \mathbb{R}$, $i = 1, 2$ are prescribed constants. By choosing a new reference frame whose origin is at the pursuer, the kinematic equations can be represented in a two-dimensional space space. In particular, let $x = x_E - x_P$ and $y = y_E - y_P$ be the relative distance between the evader and the pursuer along the x-axis and y-axis, respectively. The kinematics equations in terms of x and y are given by

$$\dot{x} = v_E \cos\psi - v_P \cos\phi + \alpha_1 x + \beta_1 y, \tag{39}$$

$$\dot{y} = v_E \sin\psi - v_P \sin\phi + \alpha_2 x + \beta_2 y. \tag{40}$$

Define the reduced state as $\mathbf{x} = [x, y]^\mathsf{T}$. The equations can then be written compactly as

$$\dot{\mathbf{x}} = v_E \mathbf{v} - v_P \mathbf{u} + w(\mathbf{x}), \tag{41}$$

where $\mathbf{v} = [\cos\psi, \sin\psi]^\mathsf{T}$ and $\mathbf{u} = [\cos\phi, \sin\phi]^\mathsf{T}$ are the controls, and where the wind field is given by

$$w(\mathbf{x}) = A\mathbf{x} + \boldsymbol{\gamma}, \tag{42}$$

where

$$A = \begin{bmatrix} \alpha_1 & \beta_1 \\ \alpha_2 & \beta_2 \end{bmatrix}, \qquad \boldsymbol{\gamma} = \begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix}. \tag{43}$$

The game terminates when capture occurs, that is, when the evader is in the *interior* of a ball $\mathcal{B}$ of radius ℓ centered at the pursuer's current location, given by $\mathcal{B} = \{\mathbf{x} \in \mathbb{R}^2 : |\mathbf{x}| \leq \ell\}$. The *terminal surface* is the manifold in the state space which, once penetrated, determines termination of the game. The terminal surface $\mathcal{C}$ is thus the circle centered at the origin of radius ℓ, i.e., $\mathcal{C} = \{\mathbf{x} \in \mathbb{R}^2 : |\mathbf{x}| = \ell\}$. Accordingly, the state space $\mathcal{E}$ is the portion of the x, y-plane exterior to $\mathcal{C}$, that is, $\mathcal{E} = \{\mathbf{x} \in \mathbb{R}^2 : |\mathbf{x}| \geq \ell\}$.

Under this setup, the first question we want to answer is whether the game can terminate given any initial condition of $\mathbf{x}(0) \in \mathcal{E}$, that is, we want to find the region in the state space such that the evader can be captured by the pursuer if their initial relative coordinates fall inside this region. This region is denoted as the *capture zone*. The region of the initial coordinates of the state which leads to escape of the evader is the *escape zone*. To this end, we formulate the problem as a *game of kind* [62], that is, the game has finitely many outcomes. In particular, when capture occurs, i.e., the terminal surface is penetrated by the evader's relative trajectory, we assign the value $+1$ to the payoff, whereas when the trajectory never reaches the terminal surface, we say that escape is achieved and assign -1 to the payoff. The outcome is

neutral [62] if the trajectory of the evader intersects the terminal surface, but it does not penetrate it. In this case, we assign 0 to the payoff. Therefore, the payoff is given by

$$J(\mathbf{x}(t_0), \phi(\cdot)\psi(\cdot)) = \begin{cases} +1, & \text{for escape,} \\ 0, & \text{for neutral outcome,} \\ -1, & \text{for capture.} \end{cases} \tag{44}$$

Our goal is to find the region of the initial conditions that leads to capture or escape with conflicting actions of the pursuer's control ϕ and the evader's control ψ that minimize/maximize the payoff (44) under the dynamic equations (39) and (40).

After identifying the capture and escape zones, our next objective will be to solve a *game of degree* [62] within the capture zone. Within the capture zone the problem has a continuous outcome; we wish to obtain the optimal relative trajectory of the evader, along with the corresponding optimal control of the pursuer and the evader. The cost for this problem is the time-to-capture t_c.

3.3 Problem Analysis

3.3.1 Effect of External Field

Before we proceed with the formulation of the differential game of kind, some discussion that can help the reader intuitively understand the types of solutions one may expect to obtain is in order. From (41) and (42) it is clear that in the reduced state space the system is described by a linear differential equation, controlled by $\mathbf{v}$ and $\mathbf{u}$. Broadly speaking, the objective of the pursuer is to make $|\mathbf{x}| \to 0$ (i.e., stabilization to the origin), whereas the objective of the evader is to ensure that this does not occur (and, ideally, make perhaps $|\mathbf{x}| \to \infty$ as time increases). The problem of controllability/stabilizability of a linear system with bounded controls has been extensively studied in the literature [22, 81, 132, 143]. The main result in this context states that

global stabilizability with bounded controls is possible only if all the eigenvalues of the matrix A have non-positive real part. However, this is a global result that holds for all initial conditions. In our problem we restrict our analysis to a region locally around the origin where capture can occur even if the matrix has eigenvalues with positive real part. An alternative point of view to clarify this difference is to consider the following: Since the flow field is approximated by an affine function, the solution of the problem will depend on the relative contributions of the flow field term $A\mathbf{x}+\boldsymbol{\gamma}$ and the contribution by both players, namely, $v_E\mathbf{v} - v_P\mathbf{u}$. Since the latter term is uniformly bounded whereas the former term increases without bound, it is clear that, if the relative distance between the players is very large, the external flow field will be too strong to be overcome by the (constrained) control actions of either player. In that case, the trajectories of both players will tend to follow the vectorfield directions of the external flow.

3.3.2 The Game of Kind

In this subsection, we follow the standard approach of the game of kind introduced in [62]. First, we focus on identifying the *usable part* of the terminal surface. The UP is the subset of $\mathcal{C}$ in which the pursuer can cause termination immediately when both players act optimally. The remaining points on $\mathcal{C}$ form the nonuseable part, that is, termination will not occur even if the trajectory reaches this part of $\mathcal{C}$ under optimal play (i.e., when both players act optimally). The part of $\mathcal{C}$ that separates the usable part and the nonuseable part of $\mathcal{C}$ is called the *boundary of the usable part* (BUP).

In order to find the usable part, we parameterize $\mathcal{C}$ with the variable s according to

$$x = \ell \cos s, \qquad y = \ell \sin s. \tag{45}$$

Let r be the Euclidean norm of the state $\mathbf{x}$, that is, let $r^2 = x^2 + y^2 = |\mathbf{x}|^2$. Taking

the time derivative on both sides, we have

$$\begin{aligned} r\dot{r} &= x(v_E \cos\psi - v_P \cos\phi + +\alpha_1 x + \beta_1 y) \\ &\quad + y(v_E \sin\psi - v_P \sin\phi + \alpha_2 x + \beta_2 y). \end{aligned} \tag{46}$$

For points on $\mathcal{C}$, equation (46) can be rewritten as

$$\begin{aligned} \ell\dot{r} &= \ell\cos s(v_E \cos\psi - v_P \cos\phi + \alpha_1 \ell\cos s + \beta_1 \ell \sin s) \\ &\quad + \ell \sin s(v_E \sin\psi - v_P \sin\phi + \alpha_2 \ell\cos s + \beta_2 \ell \sin s). \end{aligned} \tag{47}$$

The usable part of $\mathcal{C}$ is specified by the condition

$$\min_{\phi} \max_{\psi} \dot{r}(\mathbf{x}) < 0, \quad \mathbf{x} \in \mathcal{C}, \tag{48}$$

which implies that the relative trajectory is able to penetrate the terminal surface $\mathcal{C}$.

From (47) and (48) we have that, for $\mathbf{x} \in \mathcal{C}$,

$$\begin{aligned} &\min_{\phi} \max_{\psi} \dot{r}(\mathbf{x}) \\ &= \min_{\phi} \max_{\psi} \{\cos s(v_E \cos\psi - v_P \cos\phi + \alpha_1 \ell\cos s + \beta_1 \ell \sin s) \\ &\quad + \sin s(v_E \sin\psi - v_P \sin\phi + \alpha_2 \ell\cos s + \beta_2 \ell \sin s)\} \\ &= \max_{\psi} \{v_E \cos(s-\psi)\} + \min_{\phi} \{-v_P \cos(s-\phi)\} \\ &\quad + (\alpha_1 \ell\cos s + \beta_1 \ell \sin s)\cos s + (\alpha_2 \ell\cos s + \beta_2 \ell \sin s)\sin s \\ &= v_E - v_P + \ell[\alpha_1 \cos^2 s + (\beta_1 + \alpha_2)\sin s \cos s + \beta_2 \sin^2 s]. \end{aligned} \tag{49}$$

Collecting terms and using standard trigonometric identities yields

$$\begin{aligned} &= v_E - v_P + \ell[\alpha_1 \frac{1+\cos 2s}{2} + (\beta_1+\alpha_2)\frac{\sin 2s}{2} + \beta_2 \frac{1-\cos 2s}{2}] \\ &= v_E - v_P + \frac{\ell}{2}(\alpha_1+\beta_2) + \frac{\ell}{2}[(\alpha_1-\beta_2)\cos 2s + (\beta_1+\alpha_2)\sin 2s]. \end{aligned} \tag{50}$$

Let $\sigma = \sqrt{(\alpha_1-\beta_2)^2+(\beta_1+\alpha_2)^2}$. When $\sigma = 0$, we have $\alpha_1-\beta_2 = 0$ and $\beta_1+\alpha_2 = 0$. Hence, whether the game can terminate depends on the sign of $v_E - v_P + \ell(\alpha_1+\beta_2)/2$. Specifically, when $v_E - v_P + \ell(\alpha_1+\beta_2)/2 < 0$, the usable part of the terminal surface is $\mathcal{C}$ itself, whereas when $v_E - v_P + \ell(\alpha_1+\beta_2)/2 > 0$, the game will not terminate under any initial conditions of the pursuer and the evader, which means that the evader always escapes. In the latter case the whole state space $\mathcal{E}$ is the escape zone.

Henceforth, we assume that $\sigma \neq 0$. Then (48) and (50) imply that

$$\min_{\phi}\max_{\psi}\dot{r} = v_E - v_P + \frac{\ell}{2}(\alpha_1+\beta_2) + \frac{\ell\sigma}{2}\sin(\theta+2s) < 0, \tag{51}$$

where θ is given by

$$\sin\theta = \frac{\alpha_1-\beta_2}{\sigma}, \qquad \cos\theta = \frac{\beta_1+\alpha_2}{\sigma}. \tag{52}$$

From (51) we reach the following conclusion:

Proposition 3.1 *In the reduced space the game will not terminate if*

$$\frac{2(v_P - v_E) - \ell(\alpha_1+\beta_2)}{\ell\sigma} < -1, \tag{53}$$

where $\sigma = \sqrt{(\alpha_1-\beta_2)^2+(\beta_1+\alpha_2)^2}$.

Proof. From (51), we have

$$\sin(\theta+2s) < \frac{2(v_P - v_E) - \ell(\alpha_1+\beta_2)}{\ell\sigma}. \tag{54}$$

Let

$$\zeta = \frac{2(v_P - v_E) - \ell(\alpha_1 + \beta_2)}{\ell\sigma}. \tag{55}$$

Clearly, when $\zeta < -1$, the inequality (54) has no solution for s. That is, when (53) is satisfied, the game will not be able to terminate since no usable part exists under this condition. □

Corollary 3.1 *When $\zeta \geq 1$ the usable part is the whole terminal surface $\mathcal{C}$.*

Remark 3.1 *Note that (53) is a "controllability"-like condition that relates the elements of the matrix A and the bounds of the velocities of both the players so that capture is possible.*

Henceforth, we assume that $-1 \leq \zeta < 1$. Under this assumption, the BUP is determined from $\sin(\theta + 2s) = \zeta$. This yields four solutions in $[0, 2\pi)$, denoted by s_1, s_2, $s_3 = s_1 + \pi$, $s_4 = s_2 + \pi$. Hence, the BUP contains four points on $\mathcal{C}$, represented by $P_i = (\cos s_i, \sin s_i)$, $i = 1, \ldots, 4$.

A typical illustration of the terminal surface, which is divided into the usable and nonusable parts by the BUP, which consists of four points on the terminal surface, is shown in Figure 15.

Now we turn to the construction of the *barrier* [62]. The barrier is a surface in the state space that consists of initial conditions for which the outcome is neutral. One property of the barrier is that it is never crossed by either the pursuer or the evader during optimal play. In particular, the barrier emanates from the BUP and is tangent to $\mathcal{C}$ at the BUP. We denote the barrier by $\mathcal{S}$. At each point on $\mathcal{S}$ we define the normal vector $\nu = [\nu_1, \nu_2]^\mathsf{T} \in \mathbb{R}^2$ extending into the escape zone.

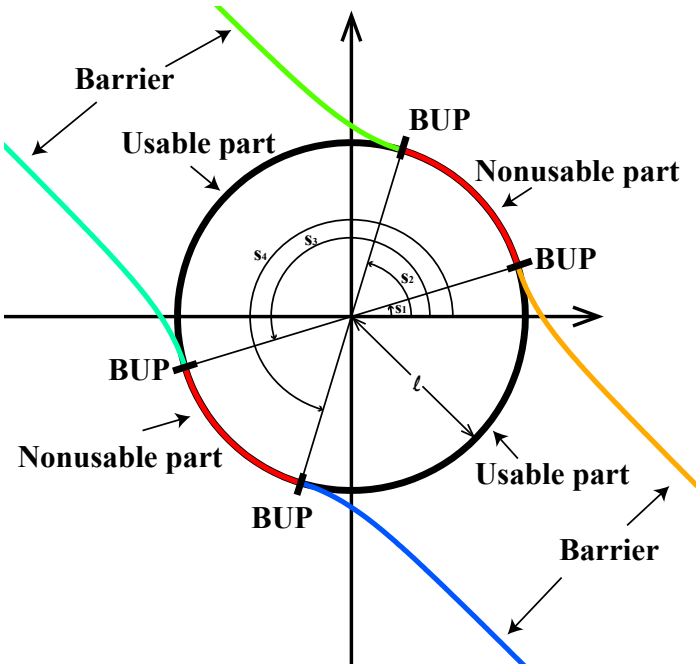


Figure 15: The terminal surface $\mathcal{C}$ of the game is given by a circle of radius ℓ. The circle is separated by the BUP (4 points on the circle parameterized by s_1 through s_4) into the usable part (black lines) and the nonusable part (red lines). Every barrier meets the terminal surface at the BUP tangentially.

The *Isaacs equation* for the game of kind for this problem becomes

$$\begin{aligned}
0 &= \min_{\phi}\max_{\psi}\{\nu_1(v_E\cos\psi - v_P\cos\phi + \alpha_1 x + \beta_1 y) \\
&\quad + \nu_2(v_E\sin\psi - v_P\sin\phi + \alpha_2 x + \beta_2 y)\} \\
&= \max_{\psi}\{v_E(\nu_1\cos\psi + \nu_2\sin\psi)\} + \min_{\phi}\{-v_P(\nu_1\cos\phi + \nu_2\sin\phi)\} \\
&\quad + \nu_1(\alpha_1 x + \beta_1 y) + \nu_2(\alpha_2 x + \beta_2 y) \\
&= v_E\rho - v_P\rho + \nu_1(\alpha_1 x + \beta_1 y) + \nu_2(\alpha_2 x + \beta_2 y), \qquad (56)
\end{aligned}$$

where $\rho = \sqrt{\nu_1^2 + \nu_2^2}$ and the corresponding optimal control of the pursuer and the

evader on the barrier are specified by

$$\cos\phi^* = \frac{\nu_1}{\rho}, \qquad \sin\phi^* = \frac{\nu_2}{\rho}, \tag{57}$$

$$\cos\psi^* = \frac{\nu_1}{\rho}, \qquad \sin\psi^* = \frac{\nu_2}{\rho}. \tag{58}$$

From [62], it follows that $\nu_i, i = 1, 2$, satisfy the differential equations

$$\dot{\nu}_i = -\sum_j \nu_j \frac{\partial f_j(\mathbf{x}, \phi^*, \psi^*)}{\partial \mathbf{x}_i}, \quad i = 1, 2, \tag{59}$$

where $f_j, j = 1, 2$, stands for the right-hand side of (39) and (40), respectively. We take these equations and the original equations under the optimal control ϕ^* and ψ^*, and reverse the time direction by replacing t with $\tau = t_c - t$ to obtain the Retrogressive Path Equations (RPE). These are the differential equations with respect to the retrograde time τ and indicate the fact that the game will be solved backwards in time starting from the terminal surface $\mathcal{C}$. Denoting with $(\,\mathring{}\,)$ the derivative with respect to τ, the retrograde evolution of the states and the vector ν can be established as:

$$\begin{aligned}
\mathring{x} &= (v_P - v_E)\frac{\nu_1}{\rho} - \alpha_1 x - \beta_1 y, \\
\mathring{y} &= (v_P - v_E)\frac{\nu_2}{\rho} - \alpha_2 x - \beta_2 y, \\
\mathring{\nu}_1 &= \alpha_1 \nu_1 + \alpha_2 \nu_2, \\
\mathring{\nu}_2 &= \beta_1 \nu_1 + \beta_2 \nu_2.
\end{aligned} \tag{60}$$

By the definition of the BUP and the barrier, it is clear that the barrier starts at the BUP towards the state space $\mathcal{E}$ in a retrogressive sense. Moreover, the two surfaces meet tangentially, since no penetration occurs at the BUP and the vectorfields of both players are tangential to the barrier. Pick any $\bar{s} \in \{s_1, s_2, s_3, s_4\}$. The initial

conditions for the RPEs are thus given by

$$\begin{aligned} x(\tau = 0) &= \ell \cos \bar{s}, \\ y(\tau = 0) &= \ell \sin \bar{s}, \\ \nu_1(\tau = 0) &= \cos \bar{s}, \\ \nu_2(\tau = 0) &= \sin \bar{s}. \end{aligned} \tag{61}$$

By integrating (60) subject to (61) we obtain

$$\nu(\tau) = e^{A^\top \tau} \nu(\tau = 0), \tag{62}$$

$$\mathbf{x}(\tau) = e^{-A\tau} \mathbf{x}(\tau = 0) + \int_0^\tau e^{-A(\tau - \xi)} b(\xi) \, \mathrm{d}\xi, \tag{63}$$

where

$$b(\tau) = (v_P - v_E) \frac{\nu(\tau)}{|\nu(\tau)|}. \tag{64}$$

After we substitute the solution of $\nu(\tau)$ and the initial conditions to the solution of $\mathbf{x}$, we obtain

$$\mathbf{x}(\tau) = \ell e^{-A\tau} \begin{bmatrix} \cos \bar{s} \\ \sin \bar{s} \end{bmatrix} + (v_P - v_E) e^{-A\tau} \int_0^\tau \frac{e^{(A + A^\top)\xi}}{|\nu(\xi)|} \begin{bmatrix} \cos \bar{s} \\ \sin \bar{s} \end{bmatrix} \mathrm{d}\xi. \tag{65}$$

By plotting the trajectories of (65) given the four initial conditions of the BUP, we can determine whether these are valid barriers and whether the state space $\mathcal{E}$ is separated by the barriers.

If $\mathcal{E}$ is indeed separated by the barriers, then the regions of $\mathcal{E}$ that contains the usable part of the terminal surface will form the capture zone, and the rest of the regions associated with the nonusable part of the terminal surface will form the escape zone. Otherwise, the whole state space is either the capture zone or the escape zone.

3.3.3 Game of Degree in the Capture Region

Now that we have identified the capture region, we aim at determining the optimal trajectory of $\mathbf{x}$ inside this region by solving a game of degree. We assume that within the capture region the pursuer tries to minimize capture time, whereas the evader tries to maximize capture time. Hence, within the capture region the performance index is

$$\mathcal{J} = \int_0^{t_c} \mathrm{d}t, \tag{66}$$

To this end, we define the value function $\mathcal{V}(\mathbf{x})$, which satisfies the HJI equation

$$0 = \min_{\phi} \max_{\psi} \mathcal{H}(\mathbf{x}, \mathcal{V}_{\mathbf{x}}), \tag{67}$$

where the Hamiltonian $\mathcal{H}$ is given by

$$\begin{aligned} \mathcal{H} = 1 &+ \frac{\partial \mathcal{V}}{\partial x}\Big(v_E \cos\psi - v_P \cos\phi + \alpha_1 x + \beta_1 y\Big) \\ &+ \frac{\partial \mathcal{V}}{\partial y}\Big(v_E \sin\psi - v_P \sin\phi + \alpha_2 x + \beta_2 y\Big). \end{aligned} \tag{68}$$

Let $\mathcal{V}_x = \frac{\partial \mathcal{V}}{\partial x}$, $\mathcal{V}_y = \frac{\partial \mathcal{V}}{\partial y}$, then (67) can be rewritten as

$$\begin{aligned} 0 = 1 &+ \mathcal{V}_x(\alpha_1 x + \beta_1 y) + \mathcal{V}_y(\alpha_2 x + \beta_2 y) \\ &+ \min_{\phi}\{-v_P(\mathcal{V}_x \cos\phi + \mathcal{V}_y \sin\phi)\} + \max_{\psi}\{v_E(\mathcal{V}_x \cos\psi + \mathcal{V}_y \sin\psi)\}. \end{aligned} \tag{69}$$

Hence, the optimal controls ϕ^* and ψ^* are given by

$$\begin{aligned} \cos\phi^* &= \frac{\mathcal{V}_x}{\mu}, \qquad \sin\phi^* = \frac{\mathcal{V}_y}{\mu}, \\ \cos\psi^* &= \frac{\mathcal{V}_x}{\mu}, \qquad \sin\psi^* = \frac{\mathcal{V}_y}{\mu}, \end{aligned} \tag{70}$$

where $\mu = \sqrt{\mathcal{V}_x^2 + \mathcal{V}_y^2}$. Plugging (70) back into the Hamiltonian, we get the optimal Hamiltonian

$$\mathcal{H}^* = 1 + (v_E - v_P)\mu + \mathcal{V}_x(\alpha_1 x + \beta_1 y) + \mathcal{V}_y(\alpha_2 x + \beta_2 y). \tag{71}$$

The RPEs can then be expressed as

$$\mathring{x} = (v_P - v_E)\frac{\mathcal{V}_x}{\mu} - \alpha_1 x - \beta_1 y, \tag{72}$$

$$\mathring{y} = (v_P - v_E)\frac{\mathcal{V}_y}{\mu} - \alpha_2 x - \beta_2 y, \tag{73}$$

$$\mathring{\mathcal{V}}_x = \alpha_1 \mathcal{V}_x + \alpha_2 \mathcal{V}_y, \tag{74}$$

$$\mathring{\mathcal{V}}_y = \beta_1 \mathcal{V}_x + \beta_2 \mathcal{V}_y. \tag{75}$$

On the terminal surface $\mathcal{C}$, we have $\mathcal{V} = 0$. Along with the parameterization of $\mathcal{C}$ by $x = \ell \cos s$, $y = \ell \sin s$, we get

$$0 = \frac{\partial \mathcal{V}}{\partial s} = \ell(-\mathcal{V}_x \sin s + \mathcal{V}_y \cos s).$$

Upon solving these equations, we further get, for some $\delta > 0$,

$$\mathcal{V}_x(\tau = 0) = \delta \cos s, \qquad \mathcal{V}_y(\tau = 0) = \delta \sin s. \tag{76}$$

By substituting (76) into the expression for $\mathcal{H}^*$, we can solve for δ to obtain

$$\delta = \frac{1}{v_P - v_E - \ell(\alpha_1 \cos^2 s + (\beta_1 + \alpha_2) \sin s \cos s + \beta_2 \sin^2 s)}. \tag{77}$$

Integrating the RPE's (72) through (75) subject to the initial conditions (76) yields

$$\begin{bmatrix} \mathcal{V}_x(\tau) \\ \mathcal{V}_y(\tau) \end{bmatrix} = e^{A^\top \tau} \begin{bmatrix} \delta \cos s \\ \delta \sin s \end{bmatrix}, \tag{78}$$

and hence

$$\begin{bmatrix} x(\tau) \\ y(\tau) \end{bmatrix} = e^{-A\tau} \begin{bmatrix} \ell \cos s \\ \ell \sin s \end{bmatrix} + (v_P - v_E) e^{-A\tau} \int_0^\tau \frac{e^{(A+A^\top)\xi}}{\mu(\xi)} \begin{bmatrix} \delta \cos s \\ \delta \sin s \end{bmatrix} \mathrm{d}\xi. \tag{79}$$

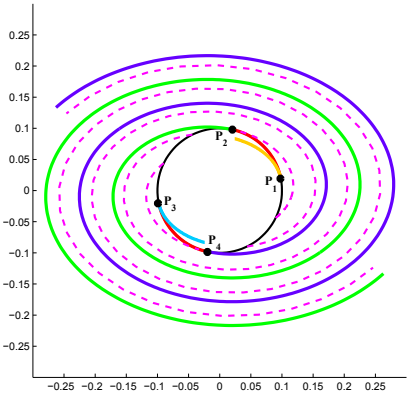
3.4 Simulation Results

In this section, we present numerical simulations to illustrate the previous analysis. In the following cases, we vary the matrix in (42) while we keep ℓ, v_P and v_E fixed to compute different types of barriers under different flow fields. Henceforth, we let $\ell = 0.1$, $v_P = 1.0$, $v_E = 0.9$. For the parameters of the flow field in the inertial frame, we set $\gamma_1 = \gamma_2 = 0$.

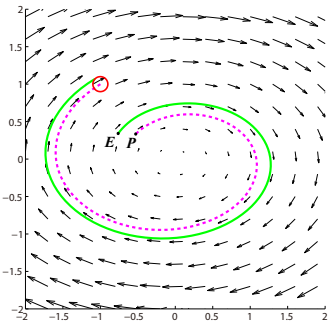
Case 1: $A = \begin{bmatrix} 0 & 10 \\ -5 & 0 \end{bmatrix}$. This matrix has two pure imaginary eigenvalues (center). In this case, $\sigma = 5$, $\zeta = 0.4$ and the corresponding values for the BUP are $s_1 = 0.2058, s_2 = 1.3650, s_3 = 3.3474$ and $s_4 = 4.5066$. As shown in Figure 16(a), the trajectories of the RPEs emanating from P_1 and P_3 are inside $\mathcal{B}$; these two trajectories are outside the state space $\mathcal{E}$. Hence, they are not valid barriers and are discarded. On the other hand, the trajectories emanating from P_2 and P_4 are valid barriers. They have spiral-like shapes but they fail to separate the state space into two parts. The whole state space is a capture zone; regardless of the initial conditions of the two players, capture is guaranteed. The dashed magenta lines in Figure 16(a) show the optimal trajectories in relative coordinates with respect to different initial positions on the usable part of the terminal surface. Although the barrier does not separate the state space into capture and escape zones, it is still not crossed during optimal

play, which gives us some information as to how the optimal trajectories look like. The barrier also marks a discontinuity in the value function.

Given the initial positions for the evader and the pursuer as $\mathbf{x}_E(0) = [-0.764, 0.337]$ and $\mathbf{x}_P(0) = [-0.524, 0.336]$, respectively, the optimal trajectories of the evader and the pursuer in the inertial frame are depicted in Figure 16(b). These trajectories are consistent with the external flow field represented by the black arrows. The results suggest that both players are trying to take advantage of the flow field, in this case. Intuitively, this makes sense. Since the matrix A has purely imaginary eigenvalues, the uncontrolled system trajectories are circles around the origin. The flow field does not give an advantage to either the pursuer or the evader. It is then reasonable that under optimal controls of both players, the trajectories in the reduced state move in spiral-like patterns, as confirmed in Figure 16(a).



(a) Barriers in frame $\mathcal{M}$. The dashed magenta lines are the optimal trajectories of the relative coordinates emanating from the usable part of the terminal surface.



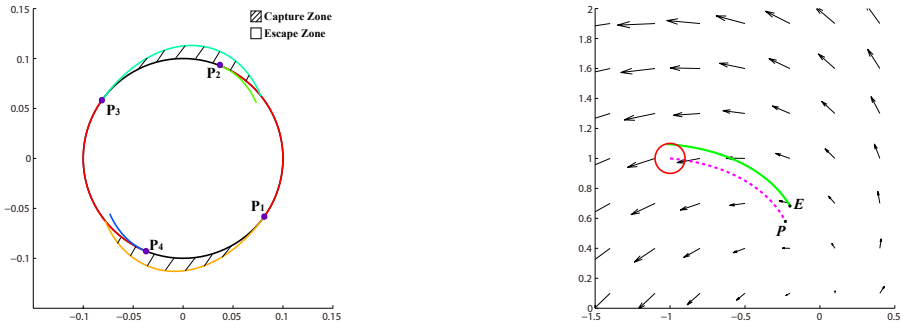
(b) Optimal trajectories of the evader in green and the pursuer in dashed magenta, respectively. Red circle around the final position of the pursuer represents the terminal surface. The flow field is depicted by the black arrows in the background.

Figure 16: Barriers and optmal trajectories of pursuit-evasion problem when $A = [0, 10; -5, 0]$.

Case 2: $A = \begin{bmatrix} 1.4020 & -1.0772 \\ 1.4770 & 0.7756 \end{bmatrix}$. In this case, the eigenvalues are a complex conjugate pair with positive real part (unstable spiral). These values correspond to $\sigma = 0.7431$, $\zeta = -0.2390$ and the corresponding parameters for the BUP are

$s_1 = 5.6612, s_2 = 1.1901, s_3 = 2.5196, s_4 = 4.3317$. As depicted in Figure 17(a), and similarly to the first case, the trajectories of the RPEs emanating from P_2 and P_4 are inside $\mathcal{B}$ and are thus discarded. The trajectories starting from P_1 and P_3 intersect $\mathcal{C}$ after some time, and thus the trajectories after the intersection are discarded. In this case, the barrier separates the capture zone from the escape zone. The capture zone is represented by the shaded region in Figure 17(a). All the remaining space outside the circle is the escape zone.

The optimal trajectories of the evader and the pursuer in the inertial frame are depicted in Figure 17(b) with initial positions $\mathbf{x}_E(0) = [-0.20, 0.683]$ and $\mathbf{x}_P(0) = [-0.230, 0.579]$, respectively. Notice that in this case, there is a small region of relative initial positions for the pursuer and the evader such that capture occurs.



(a) Barriers in frame $\mathcal{M}$. The shaded region is the escape zone and the white region outside the circle is the escape zone.

(b) Optimal trajectories of the evader in green and the pursuer in dashed magenta.

Figure 17: Barriers and optmal trajectories of pursuit-evasion problem when $A = [1.4020, -1.0772; 1.4770, 0.7756]$.

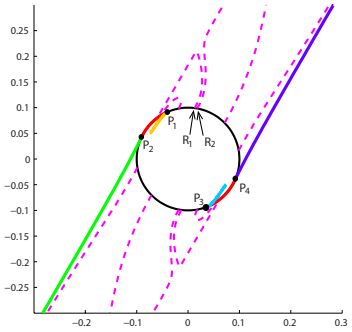
In this case the matrix A has two complex eigenvalues with positive real parts, which implies that the origin is an unstable spiral. The trajectories of the uncontrolled system would result in $|x| \to \infty$ as time goes on. In this case, the flow field gives an advantage to the evader. Indeed, as shown in Figure 17(a), the capture zone is very small compared to the escape zone.

Case 3: $A = \begin{bmatrix} 0.3188 & -0.4336 \\ -1.3077 & 0.3426 \end{bmatrix}$. In this case and the next one, both matrices have two real eigenvalues, one positive and the other negative. However, these two cases generate totally different shapes of capture and escape regions. In this case, $\sigma = 1.7415$, $\zeta = 0.7687$ and the corresponding parameters for the BUP are $s_1 = 2.0023, s_2 = 2.6964, s_3 = 5.1439, s_4 = 5.8380$. As can be seen in Figure 18(a), the barriers emanating from P_1 and P_3 are inside the ball $\mathcal{B}$ and are thus discarded. The remaining valid barriers are illustrated by green and blue lines emanating from P_2 and P_4, respectively. These two barriers fail to divide the state space into capture and escape zones. However, this does not imply that the whole state space is a capture zone. The dashed magenta lines in Figure 18(a) are the optimal trajectories of the relative coordinates emanating from different initial positions on the usable part of the terminal surface. These magenta lines fail to cover the whole state space, suggesting that there exist some initial conditions of the pursuer and the evader such that the game will not terminate, and the evader escapes.

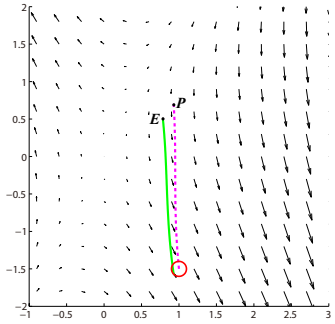
Notice in Figure 18(a) that the optimal trajectories emanating from two very close positions, namely, R_1 and R_2, end up separating from each other after some time. The reason that such phenomenon occurs is that around the point of separation, there exists a critical point of the system described by (72) and (73). To verify the existence of this critical point, we integrate (72) through (75) backwards in time starting from R_1 and R_2 for 10 seconds and plot the values of the right-hand side of (72) and (73) (or equivalently the velocity components) with respect to time. The results are shown in Figure 19, where the blue and dashed red lines correspond to the trajectories emanating from R_1 and R_2, respectively. Around $t = 6$, the blue lines gets really close to zero, which indicates the existence of a critical point in this neighborhood. After $t = 6$, both blue lines take negative values, whereas both red lines take positive values. In other words, the velocity vectors of the trajectory emanating from R_1 and R_2 take

almost opposite direction from one another, which leads to the situation shown in Figure 18(a).

Typical optimal trajectories of the evader and the pursuer in the inertial frame with initial positions $\mathbf{x}_E(0) = [0.785, 0.499]$ and $\mathbf{x}_P(0) = [0.926, 0.747]$ are depicted in Figure 18(b).



(a) Barriers in frame $\mathcal{M}$. The dashed magenta lines represent the optimal trajectories of the relative coordinates emanating from the usable part of the terminal surface.



(b) Optimal trajectories of the evader in green and the pursuer in dashed magenta.

Figure 18: Barriers and optmal trajectories of pursuit-evasion problem when $A = [0.3188, -0.4336; -1.3077, 0.3426]$.

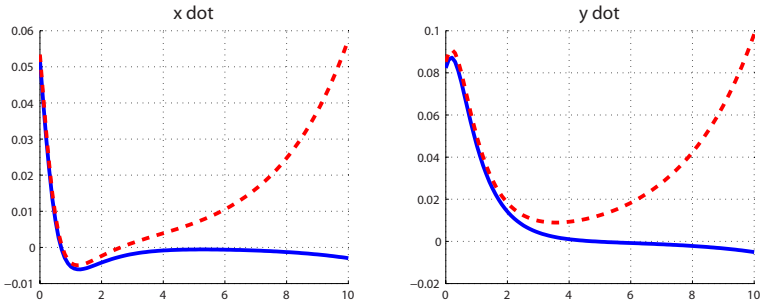
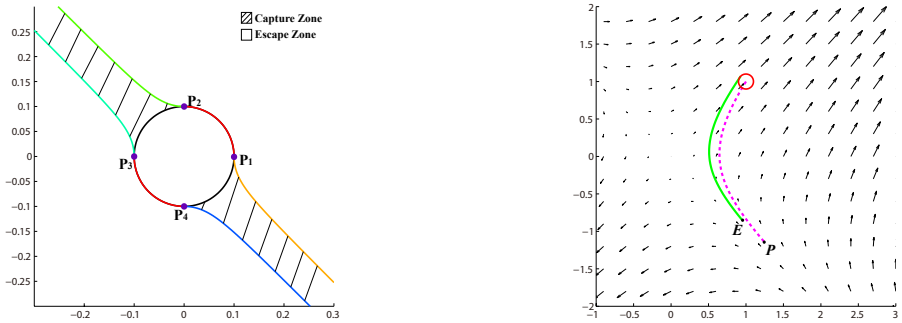


Figure 19: Velocity components of the system (72) and (73), where the blue and dashed red lines are with respect to the trajectories emanating from R_1 and R_2, respectively.

Case 4: $A = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}$. In this case, $\sigma = 4$, $\zeta = 0$ and the corresponding parameters for the BUP are $s_1 = 0, s_2 = \pi/2, s_3 = \pi$ and $s_4 = 3\pi/2$. As illustrated in

Figure 20(a), all four trajectories emanating from P_1, P_2, P_3 and P_4 are valid barriers. They separate the state space into two capture zones and two escape zones, depicted in the figure by the two shaded regions and the two white regions, respectively. Typical optimal trajectories of the evader and the pursuer in the inertial frame with initial positions $\mathbf{x}_E(0) = [0.951, -0.852]$ and $\mathbf{x}_P(0) = [1.265, -1.165]$ are shown in Figure 20(b).



(a) Four valid barriers in frame $\mathcal{M}$ emanating from P_1 through P_4. The state space is divided by two shaded capture zones and two white escape zones.

(b) Optimal trajectories of the evader in green and the pursuer in dashed magenta.

Figure 20: Barriers and optmal trajectories of pursuit-evasion problem when $A = [1, 2; 2, 1]$.

In both Cases 3 and 4, the matrix A has one positive eigenvalue and one negative eigenvalue. Hence, the origin is a saddle point, and in some part of the plane the flow field points towards the origin (helping the pursuer), whereas in other parts it points away from the origin (thus giving an advantage to the evader), as indicated by the black vector fields in Figures 18(b) and 20(b). This suggests that the pursuer tries to steer the game in the part of the space that the flow field is beneficial to him and the evader does the same, i.e., tries to steer the state to the parts of the state space that are more helpful to him. In this case, the game will terminate (or not) depending on whether the pursuer can capture the evader before the latter moves in the part of the space that the former has an advantage.

These strategies of the pursuer and the evader are supported by the trajectories depicted in Figures 18(a) and 20(a). Specifically, we observe in Figure 20(a) that although the barriers fail to specify all the boundaries of the capture zone, the state space is indeed divided into different parts, with the capture zone covered by the optimal trajectories emanating from the usable part of the terminal surface, and the escape zone indicated by the blank space on the left and right side of the figure.

CHAPTER IV

PURSUIT-EVASION GAMES IN GENERAL FLOW FIELDS

4.1 Introduction

In this Chapter we consider a multi-pursuer/one-evader pursuit-evasion game in an external dynamic flow field that is assumed to be known. Due to the generality of the external flow, Isaacs's approach is not readily applicable. Instead, we follow a different approach and we find the optimal trajectories of the players through a reachable set method. Specifically, we utilize the level set method [111, 133] to generate the reachable sets of both the evader and the pursuers and to retrieve the corresponding optimal control actions at the current location of the agents by backward propagation of their respective reachable sets.

Level set methods have been previously applied by Tomlin et al. to solve pursuit-evasion games [66,99]. Ref. [66] aims at solving a non-zero sum pursuit-evasion game where the evader and each individual pursuer are assigned their own value functions. The authors of Ref. [99] first decrease the degrees of freedom of the problem by reformulating it in terms of the relative distance between the pursuer and the evader. The level set method is then applied to solve the corresponding Hamilton-Jacobi-Isaacs (HJI) equation that governs the *backward reachable set* from the target set in order to solve the differential game. Our approach differs from that in Refs. [99] since we do not attempt to solve the pursuit-evasion game directly by solving the corresponding HJI equation. Instead, we generate the *forward* reachable sets of the players, and find the optimal time-to-capture as the first instance when the reachable set of the evader is fully covered by the reachable set of the pursuer [102]. We then

identify the first rendezvous point of the players and retrieve the optimal trajectories and controls of both players through backtracking of their respective trajectories [86, 88]. The reason we follow this approach instead of the more direct approach in Ref. [99] is due to the dimensionality the problem. When introducing complex dynamic environmental effects into the system, the pursuit-evasion problem cannot be reduced to a problem described solely in terms of the relative distance between the pursuer and the evader, unless some very restrictive assumptions are imposed on the structure of the external flow field [142]. In other words, in order to deal with a pursuit-evasion problem between one pursuer and one evader taking place in the presence of a general flow field, the level set method needs to be implemented on a fourth-dimensional state space. The computational cost of level set methods is very high when the dimension exceeds three or four [98]. On the other hand, the forward reachable set approach is quite efficient, since the propagated level sets all remain two-dimensional. The approach works even for realistic flows with dynamic currents whose speed can be much larger than the vehicle speeds [83], and can also treat dynamic obstacles [84]. Finally, since the generation of the reachable sets of each player can be done independently of the other players, the solution can be implemented in a decentralized manner using parallel computing.

4.2 Problem Formulation

We consider a pursuit-evasion game in the presence of an external flow field with n pursuers and a single evader. Henceforth, we will refer to the pursuers and the evader collectively as "agents." The dynamics of the pursuers P_i $(i = 1, \ldots, n)$, are given by

$$\dot{X}_P^i = u_P^i + w(X_P^i, t), \qquad X_P^i(t_0) = X_{P_0}^i, \tag{80}$$

where $X_P^i := [x_P^i, y_P^i]^\intercal \in \mathcal{D} \subset \mathbb{R}^2$ denotes the position of the i-th pursuer. Here $\mathcal{D}$ denotes a compact subset of $\mathbb{R}^2$ and u_P^i is the control input (i.e., velocity) of the i-th

pursuer such that $u_{P}^{i} \in \mathcal{U}_{P}^{i}$, for all $i \in \mathcal{I}$, and $\mathcal{I} = \{1, 2, \ldots, n\}$ stands for the index set of the pursuers. The set $\mathcal{U}_{P}^{i}$ consists of all piecewise continuous functions whose range is included in the set $U_{P}^{i} = \{u^{i} \in \mathbb{R}^{2}, |u^{i}| \leqslant \bar{u}^{i}\}$, where $|\cdot|$ represents the Euclidean norm and $\bar{u}^{i}$, $i \in \mathcal{I}$, are constants. If $u_{P}^{i} \in \mathcal{U}_{P}^{i}$, we say that u_{P}^{i} is an *admissible* control for the i-th pursuer. In (80), $w(X,t) \in \mathbb{R}^{2}$ represents an exogenous dynamic flow, but it could also represent an endogenous drift owing to the nonlinear dynamics of the agent. It is reasonable to assume that the magnitude of this flow (e.g., winds or currents) is bounded from above by some constant, that is, there exists a constant $\bar{w}$ such that $|w(X,t)| \leq \bar{w}$, for all $(X,t) \in \mathcal{D} \times [t_0, \infty)$.

The objective of the pursuers is to intercept the evader, whose kinematics is given by

$$\dot{X}_{E} = u_{E} + w(X_{E}, t), \qquad X_{E}(t_0) = X_{E_0}, \tag{81}$$

where $X_{E} = [x_{E}, y_{E}]^{\intercal} \in \mathcal{D} \subset \mathbb{R}^{2}$ is the position of the evader, and u_{E} is its control input (i.e., velocity) such that $u_{E} \in \mathcal{U}_{E}$, where $\mathcal{U}_{E}$ consists of all piecewise continuous functions whose range is included in the set $U_{E} = \{v \in \mathbb{R}^{2}, |v| \leqslant \bar{v}\}$. When $u_{E} \in \mathcal{U}_{E}$, we say that u_{E} is an *admissible* control of the evader.

The game begins at time $t = t_0$ with initial positions X_{E_0} and $X_{P_0}^{i}, (i \in \mathcal{I})$, for the evader and the pursuers, respectively, and terminates when the evader coincides with at least one of the pursuers, in which case capture occurs. That is, capture implies that there exists $i \in \mathcal{I}$ and a terminal time $T \geq t_0$ such that $X_{P}^{i}(T) = X_{E}(T)$. Equivalently, the game terminates if, for any admissible control of the evader $u_{E} \in \mathcal{U}_{E}$, there exists a set of admissible controls $(u_{P}^{1}, \ldots, u_{P}^{n}) \in \mathcal{U}_{P}^{1} \times \cdots \times \mathcal{U}_{P}^{n}$ of the pursuers such that $X_{P}^{i}(T) = X_{E}(T)$ for some $i \in \mathcal{I}$ and some time $T \geq t_0$. The pursuers aim to minimize the time-to-capture if possible, whereas the evader prefers to avoid capture for as long as possible.

Let $\bar{X} = [X_{E}^{\intercal}, X_{P}^{1\intercal}, X_{P}^{2\intercal}, \ldots, X_{P}^{n\intercal}]^{\intercal} \in \mathbb{R}^{2(n+1)}$ denote the state of the game. The game begins at initial time $t_0 = 0$ with initial positions $\bar{X}_0 = [X_{E_0}^{\intercal}, X_{P_0}^{1\intercal}, X_{P_0}^{2\intercal}, \ldots, X_{P_0}^{n\intercal}]^{\intercal}$,

and terminates when at least one of the pursuers reaches the location of the evader. The terminal time T of the game is defined by

$$T = \inf\{t \in \mathbb{R}_+ : X_P^i(t) = X_E(t), \quad i \in \mathcal{I}\}. \tag{82}$$

Let $J(\gamma_P^1, \gamma_P^2, \ldots, \gamma_P^n, \gamma_E) = T$ be the cost function of the game, where $\gamma_P^i, \gamma_E : \mathbb{R}_+ \times \mathbb{R}^4 \mapsto \mathbb{R}^2$ denote the feedback strategies of the pursuers and the evader, respectively, namely $\gamma_P^i(t, \bar{X}(t))$ and $\gamma_E(t, \bar{X}(t))$, where $\bar{X}(t)$ is the solution of the system of equations

$$\dot{X}_P^i = \gamma_P^i(t, \bar{X}(t)) + w(X_P^i, t), \quad i \in \mathcal{I} \tag{83}$$

$$\dot{X}_E = \gamma_E(t, \bar{X}(t)) + w(X_E, t), \tag{84}$$

subject to $\bar{X}(0) = \bar{X}_0$. It is assumed that each player has perfect knowledge of the dynamics of the system represented by (80) and (81), the constraint sets U_P^i and U_E, the cost function J, as well as the initial state $\bar{X}_0$. It is also assumed that the value V of the game [62] exists, that is,

$$V = \min_{\gamma_P^1, \ldots, \gamma_P^n} \max_{\gamma_E} J = \max_{\gamma_E} \min_{\gamma_P^1, \ldots, \gamma_P^n} J. \tag{85}$$

Note that pursuit-evasion games of the form addressed here are a specific class of differential games for which the Isaacs condition [62] holds owing to the separability of the dynamics and the cost, and hence the value of the game exists [42]. The objective is to find the open-loop representation of the optimal strategies of the pursuer and the evader. In particular, we utilize a reachability-based method to obtain optimal controls $u_P^\star(t) = \gamma_P^\star(t, \bar{X}^\star(t))$ and $u_E^\star(t) = \gamma_E^\star(t, \bar{X}^\star(t))$, with $\bar{X}^\star$ denoting the corresponding optimal state trajectory of the system (83)-(84) with strategies $\gamma_P^{i\star}(t, \bar{X}^\star(t))$ and $\gamma_E^\star(t, \bar{X}^\star(t))$.

4.3 Problem Analysis

4.3.1 Reachable Sets

In order to solve the differential game introduced in the previous section, that is, in order to find the conditions for capture and to derive the corresponding optimal controls and trajectories of both players, we make use of reachable set analysis. Reachable sets provide a quick snapshot of all possible future trajectories of the agent and thus succinctly encode all possible future positions of the agent under any possible control action. Knowledge of the reachable sets of the pursuer and the evader can then be used to draw conclusions about the potential meeting of the two at some future time (or not). In this chapter we use this intuition behind the information conveyed by the reachable set of each player to solve the pursuit-evasion problem under minimal assumptions about the maximum number of players and the environment they operate in. Since the computation of the reachable sets for each player can be done independently from the other players, the proposed method is decentralized and scales well with the number of players, something that is not the case with more traditional approaches that require directly the solution of a HJI partial differential equation (see also discussion at the end of Section 4.1).

We start this section with some basic definitions and facts about reachable sets that will be useful throughout the chapter.

Definition 4.1 *(Ref. [136]) The* reachable set $\mathcal{R}(X_0, t)$ *at time* $t \geq t_0$ *of a system of the form (80) or (81) starting at initial condition* $X(t_0) = X_0$ *is the set of all the points that can be reached by the agent at time* t*.*

In particular, the reachable set of the i-th pursuer at time $\tau \geq t_0$, denoted by $\mathcal{R}^i_P(X^i_{P_0}, \tau)$, is the set of all points $X \in \mathbb{R}^2$, such that there exists a trajectory satisfying (80) for all $t \in [t_0, \tau]$ with $X_P(t_0) = X^i_{P_0}$ and $X_P(\tau) = X$. Similarly, the reachable set $\mathcal{R}_E(X_{E_0}, \tau)$ of the evader at time $\tau \geq t_0$ is the set of all points

$X \in \mathbb{R}^2$ such that there exists a trajectory satisfying (81) for all $t \in [t_0, \tau]$ with initial condition $X_{_E}(t_0) = X_{_{E_0}}$ and terminal condition $X_{_E}(\tau) = X$.

Definition 4.2 *(Ref. [86]) The boundary of the reachable set is the* reachability front*.*

The reachability fronts of the i-th pursuer and the evader at time $t \geq t_0$ will be denoted by $\partial\mathcal{R}^i_{_P}(X^i_{_{P_0}}, t)$ and $\partial\mathcal{R}_{_E}(X_{_{E_0}}, t)$, respectively.

Definition 4.3 *Given the reachable sets of the pursuers, we define the* usable reachable set *of the evader at time $t \geq t_0$ as*

$$\mathcal{R}^\star_{_E}(X_{_{E_0}}, t) = \left\{ X \in \mathcal{D} : X = X_{_E}(t) \text{ and } X_{_E}(\tau) \notin \bigcup_{i=1}^{n} \mathcal{R}^i_{_P}(X^i_{_{P_0}}, \tau), \forall \tau \in [t_0, t] \right\}. \quad (86)$$

From this definition, it is clear that $\mathcal{R}^\star_{_E}(X_{_{E_0}}, t) \subseteq \mathcal{R}_{_E}(X_{_{E_0}}, t)$. The definition implies that $\mathcal{R}^\star_{_E}(X_{_{E_0}}, t)$ is the set of all terminal points of the evader at time t, whose trajectories do not pass through any reachable sets of the pursuers at *any* time in the interval $[t_0, t]$. In other words, $\mathcal{R}^\star_{_E}(X_{_{E_0}}, t)$ is the set of terminal points of all "safe" evader trajectories.

Suppose now that at some time $t_c > t_0$, and for some $i \in \mathcal{I}$, we have that $\mathcal{R}_{_E}(X_{_{E_0}}, t_c) \subseteq \mathcal{R}^i_{_P}(X^i_{_{P_0}}, t_c)$. It follows that, for each $u_{_E} \in \mathcal{U}_{_E}$, there exists $u^i_{_P} \in \mathcal{U}^i_{_P}$ such that $X^i_{_P}(t_c) = X_{_E}(t_c)$. In other words, capture of the evader is guaranteed at time t_c by the i-th pursuer. Note that $\mathcal{R}^\star_{_E}(X_{_{E_0}}, t_c) = \emptyset$ in this case.

If, on the other hand, for some $t_e > t_0$, we have that $\mathcal{R}^\star_{_E}(X_{_{E_0}}, t_e) \neq \emptyset$, then it follows that there exists $u_{_E} \in \mathcal{U}_{_E}$ such that capture can be avoided in the time interval $[t_0, t_e]$, no matter how the pursuers choose their (admissible) controls. In other words, if $\mathcal{R}^\star_{_E}(X_{_{E_0}}, t_e) \neq \emptyset$, the game will not terminate in the time interval $[t_0, t_e]$.

The previous observations lead to the following theorem, which is the main theoretical result of this chapter. It is used later on in order to develop an efficient numerical algorithm for solving the pursuit-evasion game with multiple pursuers in the presence of a known dynamic flow field.

Theorem 4.1 *Let* $T = \inf\{t \in [t_0, +\infty) : \mathcal{R}_E^\star(X_{E_0}, t) = \emptyset\}$. *If* $T < \infty$, *then capture is guaranteed for any time greater than* T, *while the evader can always escape within a time smaller than* T. *Hence,* T *is the time-to-capture if both players play optimally. Furthemore, let* X_f *denote the location where the evader is captured by at least one of the pursuers. Then we have that*

$$X_f \in \mathcal{X} = \left\{ X \in \mathcal{D} : X = X_E(T) \text{ and } X_E(\tau) \notin \bigcup_{i=1}^{n} \mathcal{R}_P^i(X_{P_0}^i, \tau), \forall \tau \in [t_0, T) \right\}. \quad (87)$$

Proof. Since U_P^i is compact and convex for all $i \in \mathcal{I}$, it follows that, for each $(t, X) \in [t_0, \infty) \times \mathcal{D}$, the sets $\{u + w(X, t) : u \in U_P^i\}$ are compact and convex for all $i \in \mathcal{I}$. Also, since by assumption $u_P^i(t)$ and $w(X, t)$ are bounded for all $X \in \mathcal{D}$ and $t < \infty$, the solution of (80) exists on $[t_0, t_f]$, for all $t_f < \infty$. Therefore, by Filippov's Theorem [48], the reachable sets $\mathcal{R}_P^i(X_{P_0}^i, t)$ are compact, for all $t \in [t_0, t_f]$ and $i \in \mathcal{I}$. Similarly, $\mathcal{R}_E(X_{E_0}, t)$ is compact, for all $t \in [t_0, t_f]$.

Since $\mathcal{R}_E^\star(X_{E_0}, T) = \emptyset$, it follows from Definition 4.3 that for any trajectory $X_E(\cdot)$ of the evader that satisfies (81) subject to an admissible evading control $u_E \in \mathcal{U}_E$, there exists $\tau \in [t_0, T]$ such that $X_E(\tau) \in \bigcup_{i=1}^{n} \mathcal{R}_P^i(X_{P_0}^i, \tau)$. Therefore, $X_E(\tau) \in \mathcal{R}_P^k(X_{P_0}^k, \tau)$, for some $k \in \mathcal{I}$. In other words, there exists at least one admissible control $u_P^k \in \mathcal{U}_P^k$ for the k-th pursuer, such that $X_P^k(\tau) = X_E(\tau)$. Therefore, regardless of the strategy of the evader, it can be captured by the k-th pursuer at some time $\tau \in [t_0, T]$. This implies that capture is guaranteed for any time greater than or equal to T.

On the other hand, since T is the first time such that $\mathcal{R}_E^\star(X_{E_0}, T) = \emptyset$, it follows that $\mathcal{R}_E^\star(X_{E_0}, t) \neq \emptyset$ for all $t_0 \leq t < T$. Hence, for any $t \in [t_0, T)$, there exists $X_t \in \mathcal{R}_E^\star(X_{E_0}, t)$. That is, $X_t = X_E(t)$ and $X_E(\tau) \notin \bigcup_{i=1}^{n} \mathcal{R}_P^i(X_{P_0}^i, \tau)$ for all $\tau \in [t_0, t]$, for some trajectory $X_E(\cdot)$ of the evader, defined over the interval $[t_0, t]$. This means that $X_E(\cdot)$ does not pass through the reachable set of any pursuer. Hence, for any $t \in [t_0, T)$, there exist $u_E \in \mathcal{U}_E$ such that $X_E(t) = X_t$, and for all $\tau \in [t_0, t]$ and

$i \in \mathcal{I}$, there exists no $u_{_P}^i \in \mathcal{U}_{_P}$. It follows that the evader can always avoid capture before time T. From the two previous statements, we conclude that T is the optimal time-to-capture.

To complete the proof we just need to show that $X_f \in \mathcal{X}$. For any point $X \in \mathcal{X}$, it is clear that $X \in \mathcal{R}_{_E}(X_{_{E_0}}, T)$ and no pursuer can capture the evader at X prior to time T. This implies that X should be the destination of the evader if the latter aims to maximize its time-to-capture. Furthermore, at least one of the pursuers needs to reach X in order to capture the evader. Hence, the point $X = X_f$, where X_f is defined as the location where the evader is captured by at least one of the pursuers. This completes the proof.

□

The previous theorem gives us a criterion for capture of the evader, that is, capture is guaranteed when $\mathcal{R}_{_E}^{\star}(X_{_{E_0}}, t) = \emptyset$ for some $t \in [t_0, \infty)$. Also notice from (86) that, in general, $\mathcal{R}_{_E}^{\star}(X_{_{E_0}}, t)$ can be generated by keeping track of the reachable sets of the pursuers and the evader at all time prior to the capture time. However, when we add some constraints with respect to the speeds of the players, we can replace this criterion with an instantaneous condition which is easier to check and implement. Before we state and prove this result, the following lemma is needed.

Lemma 4.1 *Let $\bar{u} = \bar{u}_i$ for all $i = 1, \ldots, n$ denote the maximum speed of a pursuer, and let $\bar{v}$ denote the maximum speed of the evader, respectively, and assume that $\bar{v} < \bar{u}$. If there exists some time $t_s \geq t_0$ such that $X_{_E}(t_s) \in \mathcal{R}_{_P}(X_{_{P_0}}, t_s)$, then $X_{_E}(t) \in \mathcal{R}_{_P}(X_{_{P_0}}, t)$ for all $t \geq t_s$.*

Proof. Since $X_{_E}(t_s) \in \mathcal{R}_{_P}(X_{_{P_0}}, t_s)$ for some time $t_s \geq t_0$, it follows that there exists $u_{_P} \in \mathcal{U}_{_P}$ such that $X_{_P}(t_s) = X_{_E}(t_s)$. By assumption, we have $\bar{v} < \bar{u}$. Therefore, for any $u_{_E} \in \mathcal{U}_{_E}$ that starts from $X_{_E}(t_s)$ at time t_s, the pursuer by choosing $u_{_P} = u_{_E}$, which is admissible since $\bar{v} < \bar{u}$, can ensure that $X_{_P}(t) = X_{_E}(t)$ for all $t \geq t_s$. Hence, $X_{_E}(t) \in \mathcal{R}_{_P}(X_{_{P_0}}, t)$ for all $t \geq t_s$. □

This lemma essentially states that when the maximum speed of the evader is smaller than the maximum speed of all the pursuers, then once the evader enters the reachable set of a pursuer, it can never leave this reachable set. We are now ready to present the simplified condition on the capture of the evader as follows.

Proposition 4.1 *When* $\bar{v} \leq \min_{i \in \mathcal{I}} \bar{u}_i$, *the set* $\mathcal{R}^{\star}_{E}(X_{E_0}, t)$ *satisfies*

$$\mathcal{R}^{\star}_{E}(X_{E_0}, t) = \mathcal{R}_{E}(X_{E_0}, t) \backslash \bigcup_{i=1}^{n} \mathcal{R}^{i}_{P}(X^{i}_{P_0}, t), \tag{88}$$

for all $t \geq t_0$. *In such cases, the condition* $\mathcal{R}^{\star}_{E}(X_{E_0}, t) = \emptyset$ *is equivalent to the condition*

$$\mathcal{R}_{E}(X_{E_0}, t) \subseteq \bigcup_{i=1}^{n} \mathcal{R}^{i}_{P}(X^{i}_{P_0}, t). \tag{89}$$

Proof. By Definition 4.3, for any $X \in \mathcal{R}^{\star}_{E}(X_{E_0}, t)$, we have that $X \in \mathcal{R}_{E}(X_{E_0}, t)$ and $X \notin \bigcup_{i=1}^{n} \mathcal{R}^{i}_{P}(X^{i}_{P_0}, t)$. Therefore,

$$\mathcal{R}^{\star}_{E}(X_{E_0}, t) \subseteq \mathcal{R}_{E}(X_{E_0}, t) \backslash \bigcup_{i=1}^{n} \mathcal{R}^{i}_{P}(X^{i}_{P_0}, t). \tag{90}$$

Let now

$$X \in \mathcal{R}_{E}(X_{E_0}, t) \backslash \bigcup_{i=1}^{n} \mathcal{R}^{i}_{P}(X^{i}_{P_0}, t). \tag{91}$$

It follows that there exists a trajectory $X_{E}(\cdot)$ of the evader, defined over the interval $[t_0, t]$ such that $X = X_{E}(t)$. Furthermore, $X_{E}(t) \notin \bigcup_{i=1}^{n} \mathcal{R}^{i}_{P}(X^{i}_{P_0}, t)$. We claim that $X_{E}(\tau) \in \mathcal{R}_{E}(X_{E_0}, \tau) \backslash \bigcup_{i=1}^{n} \mathcal{R}^{i}_{P}(X^{i}_{P_0}, \tau)$, for all $\tau \in [t_0, t]$. Since, trivially, $X_{E}(\tau) \in \mathcal{R}_{E}(X_{E_0}, \tau)$ it only suffices to show that $X_{E}(\tau) \notin \bigcup_{i=1}^{n} \mathcal{R}^{i}_{P}(X^{i}_{P_0}, \tau)$, for all $\tau \in [t_0, t]$. Suppose, on the contrary, that there exist $\tau \in [t_0, t]$ and $i \in \mathcal{I}$ such that $X_{E}(\tau) \in \mathcal{R}^{i}_{P}(X^{i}_{P_0}, \tau)$. Since $\bar{v} \leq \min_{i \in \mathcal{I}} \bar{u}_i$, it follows that once the evader enters the reachable set of a pursuer, it can never leave the reachable set of this pursuer. Hence, $X_{E}(\sigma) \in$

$\mathcal{R}_P^i(X_{P_0}^i, \sigma)$ for all $\sigma \geq \tau$, contradicting (91). It follows that $X \in \mathcal{R}_E^\star(X_{E_0}, t)$ and thus

$$\mathcal{R}_E(X_{E_0}, t) \backslash \bigcup_{i=1}^{n} \mathcal{R}_P^i(X_{P_0}^i, t) \subseteq \mathcal{R}_E^\star(X_{E_0}, t). \tag{92}$$

From (90) and (92) it follows that $\mathcal{R}_E^\star(X_{E_0}, t) = \mathcal{R}_E(X_{E_0}, t) \backslash \bigcup_{i=1}^{n} \mathcal{R}_P^i(X_{P_0}^i, t)$. The equivalence of condition $\mathcal{R}_E^\star(X_{E_0}, t) = \emptyset$ with (89) follows immediately. □

In the case where $\bar{v} \leq \min_{i \in \mathcal{I}} \bar{u}_i$ the optimal time-to-capture is the first time instant when the union of the reachable sets of the pursuers $\bigcup_{i=1}^{n} \mathcal{R}_P^i(X_{P_0}^i, \tau)$ completely covers the reachable set of the evader $\mathcal{R}_E(X_{E_0}, t)$. If $\bar{v} > \bar{u}_i$, for some $i \in \mathcal{I}$ (the relative maximum speed of the evader is larger than that of the i-th pursuer), the relation (88) may not always hold. Some admissible evader trajectories may temporarily enter the reachable set of the i-th pursuer and exit later on. This is not allowable. To eliminate this possibility, in such cases $\mathcal{R}_E^\star(X_{E_0}, t)$ can be determined by treating the reachable set of the i-th pursuer as a dynamic "forbidden" region for the evader [84, 88]. That is, whenever the reachable set of the evader intersects the reachable set of any of the pursuers, we can either stop the evolution of the intersected part of the evader's reachable set or let it evolve at the same speed as the reachable set of the pursuer. This way, we can ensure that the terminal points of the admissible trajectories of the evader that temporarily enter the reachable set of the pursuer and exit later on are not included in the usable reachable set of the evader.

4.4 Numerical Construction

4.4.1 Level Set Method

In order to construct the reachable sets of the pursuers and the evader, we apply the level set method [111, 133]. The level set method is a convenient mathematical tool to track the evolution of the reachability front. It evolves the reachability front by embedding it as a hyper-surface in a higher dimension, where time is the additional

dimension. Automatic handling of merging and splitting of fronts and other topological changes are made possible by this higher dimensional embedding. The level set formulation provides an implicit representation of the front, which offers several advantages over an explicit representation [111, 133]. For example, implicit function representations are widely used for describing closed and multi-valued curves, for point classification (such as determining whether a point is inside or outside an interface) and for finding intersection points and offsets.

The choice of implicit function is not unique in order to represent a curve. The signed distance function is one of the most common choices and will be utilized in this chapter. Its definition is given as follows.

Definition 4.4 *The* signed distance function $\varphi(X)$ *with respect to a set* $\mathcal{R}$ *is defined as*

$$\varphi(X) = \begin{cases} \min\limits_{Y \in \partial\mathcal{R}} |X - Y|, & \text{if} \quad X \notin \mathcal{R}, \\ -\min\limits_{Y \in \partial\mathcal{R}} |X - Y|, & \text{if} \quad X \in \mathcal{R}. \end{cases} \tag{93}$$

Recall that, for any $c \in \mathbb{R}$, the *c-level set* of a φ is the set $\{X : \varphi(X) = c\}$. We hereby utilize the signed distance function from the reachable set to track the evolution of the fronts of the reachable sets of all agents. This is achieved by expressing the reachable front at time t as the zero-level set of the corresponding signed distance function. Assuming that the signed distance function with respect to the i-th pursuer reachable set $\mathcal{R}_P^i(X_{P_0}^i, t)$ at time t is $\phi_P^i(X, t)$, then the evolution of the reachable front $\partial\mathcal{R}_P^i(X_{P_0}^i, t)$ is governed by the viscosity solution of the Hamilton-Jacobi equation [82, 86]

$$\frac{\partial \phi_P^i(X,t)}{\partial t} + \bar{u}|\nabla \phi_P^i(X,t)| + \nabla \phi_P^i(X,t) w(X,t) = 0, \quad X \in \mathcal{D} \subset \mathbb{R}^2, \tag{94}$$

with initial condition $\phi_P^i(X, t_0) = |X - X_{P_0}^i|$. Note that $\mathcal{R}_P^i(X_{P_0}^i, t) = \{X \in \mathcal{D} :$

$\phi_P^i(X,t) \leq 0\}$ and $\partial\mathcal{R}_P^i(X_{P_0}^i,t) = \{X \in \mathcal{D} : \phi_P^i(X,t) = 0\}$.

Similarly, the reachable front $\partial\mathcal{R}_E(X_{E_0},t)$ of the evader is computed by solving the Hamilton-Jacobi equation

$$\frac{\partial\phi_E(X,t)}{\partial t} + \bar{v}|\nabla\phi_E(X,t)| + \nabla\phi_E(X,t)w(X,t) = 0, \quad X \in \mathcal{D} \subset \mathbb{R}^2, \tag{95}$$

with initial condition $\phi_E(X,t_0) = |X - X_{E_0}|$, where $\phi_E(X,t)$ is the signed distance function with respect to the reachable set $\mathcal{R}_E(X_{E_0},t)$ of the evader at time t.

In the case where the condition in Proposition 4.1 is not satisfied, we need to track $\partial\mathcal{R}_E^\star(X_{E_0},t)$ in order to determine the optimal time-to-capture. Instead of propagating $\partial\mathcal{R}_E^\star(X_{E_0},t)$ directly, we propagate an intermediate reachable front $\partial\tilde{\mathcal{R}}_E(X_{E_0},t)$ which can be computed by solving the following modified version of the Hamilton-Jacobi equation

$$\frac{\partial\tilde{\phi}_E(X,t)}{\partial t} + \tilde{v}(t)\,|\nabla\tilde{\phi}_E(X,t)| + \nabla\tilde{\phi}_E(X,t)w(X,t) = 0, \quad X \in \mathcal{D} \subset \mathbb{R}^2, \tag{96}$$

where

$$\tilde{v}(t) = \begin{cases} \min\limits_{i\in\mathcal{I}} \bar{u}_i, & \text{if } \bigcup\limits_{i=1}^{n} \phi_P^i(X,t) < 0, \\ \bar{v}, & \text{otherwise,} \end{cases} \tag{97}$$

and initial condition $\tilde{\phi}_E(X,t_0) = |X - X_{E_0}|$.

The main idea here is to treat the reachable sets of the pursuers as moving obstacles and propagate $\tilde{\mathcal{R}}_E(X_{E_0},t)$ with the maximum speed of the evader $\bar{v}$ for the parts that fall outside the union of the reachable sets of the pursuers, and to keep pace with the propagation of the reachable set of the slowest pursuer when the front of the evader enters any reachable set of the pursuers. By doing this, we can make sure that the front of the evader does not grow out of the union of the reachable sets

of the pursuers. The parts of the reachable front of the evader that do not encounter the reachable sets of the pursuers remain unaffected by the change of speed from $\bar{v}$ to $\tilde{v}$, since these changes are only performed for points inside the reachable sets of the pursuers.

Let $\tilde{\mathcal{R}}_E(X_{E_0}, t) = \{X \in \mathcal{D} : \tilde{\phi}_E(X, t) \leq 0\}$. At every time instant t, by construction, $\tilde{\mathcal{R}}_E(X_{E_0}, t)$ excludes all the points X such that $X = X_E(t)$ and $X \notin \bigcup_{i=1}^{n} \mathcal{R}_P^i(X_{P_0}^i, t)$, while $X \in \bigcup_{i=1}^{n} \mathcal{R}_P^i(X_{P_0}^i, \tau)$, for some $\tau \in [t_0, t)$. It follows that

$$\mathcal{R}_E^\star(X_{E_0}, t) = \tilde{\mathcal{R}}_E(X_{E_0}, t) \backslash \bigcup_{i=1}^{n} \mathcal{R}_P^i(X_{P_0}^i, t).$$

Moreover, since $\mathcal{R}_P^i(X_{P_0}^i, t) = \{X \in \mathcal{D} : \phi_P^i(X, t) \leq 0\}$, the usable reachable set of the evader can also be represented in a form that is more suitable for numerical calculations, that is,

$$\mathcal{R}_E^\star(X_{E_0}, t) = \{X \in \mathcal{D} : \tilde{\phi}_E(X, t) \leq 0 \text{ and } \bigcup_{i=1}^{n} \phi_P^i(X, t) \geq 0\}.$$

4.4.2 Classification of Pursuers

For problems with a large number of pursuers it is quite possible that optimal capture may not involve all pursuers. That is, not all pursuers need to go after the target at the same time. In Ref. [14], for instance, a sequential pursuit strategy was introduced, according to which only a single pursuer participates in the pursuit of the target/evader, although the specific pursuer may change dynamically as the game evolves. In certain applications, such as when the pursuers are subject to energy or fuel limitations, or when they play a dual role as pursuers and guards of a certain region of responsibility, it may be beneficial that some of the pursuers remain inactive. In group pursuit problems involving several pursuers, we may therefore classify the pursuers according to their level of involvement as either redundant, active, or

guards. Below we elaborate on the motivation of this classification.

Redundant Pursuers. When we formulate a multiple-pursuers/one-evader pursuit-evasion game, depending on the initial positions of the pursuers and the evader, there may be some pursuers that do not affect the outcome of the pursuit.

Definition 4.5 *A pursuer P_k is* redundant *if, given the pursuer set $\{P_1, \ldots, P_n\}$, the optimal time-to-capture T is the same as the optimal time-to-capture $\tilde{T}$ given the pursuer set $\{P_1, \ldots, P_n\} \backslash P_k$.*

From the point of view of the pursuers, it is important to identify any redundant pursuers, since fuel or energy saving may result by placing these redundant pursuers on standby, and deploy them only if the evader shows up in their vicinity, or when it is absolutely necessary to ensure capture.

Through the reachable set approach, we can find the minimum number of pursuers needed to capture an evader under optimal time-to-capture pursuit. One way to identify any redundant pursuers is, for each pursuer, to compare the two optimal values of time-to-capture with and without this pursuer. If these two values turn out to be equal to each other, then this pursuer is redundant.

When the condition $\bar{v} \leq \min_{i \in \mathcal{I}} \bar{u}_i$ is satisfied, the following method to determine the redundant pursuers is more practical. Specifically, the j-th pursuer is redundant if $\mathcal{R}_E(X_{E_0}, T) \subseteq \bigcup_{i=1, i \neq j}^{n} \mathcal{R}_P^i(X_{P_0}^i, T)$, where T is the optimal time-to-capture given the pursuer set $\{P_1, \ldots, P_n\}$. For instance, Figures 21 and 22 show two pursuit-evasion problems restricted in the domain $\mathcal{D} = [0, 25] \times [0, 25]$. The initial positions of the two pursuers and the evader are depicted by the green, red and blue dots, respectively. The maximum speeds of the pursuers and the evader are given by $\bar{u}^1 = \bar{u}^2 = 2, \bar{v} = 1$. The vector field of the flow field is shown in the background in black. In the first problem there are two pursuers P_1 and P_2 against one evader, while in the second problem an additional pursuer, P_3, is added to the pursuer team.

In this example, and all subsequent ones (unless stated otherwise), it will be assumed that the pursuers have larger maximum speed than the evader. In the absence of an external field, and under simple motion by all players, it is known that the evader can always avoid capture if it has a speed advantage over the pursuers [124]. In the presence of an external field, however, this may not always be the case. Later on, we provide an example, where the evader is captured even when all pursuers have maximum speed that is smaller than the speed of the evader. Please also note that, similarly, an evader may be able to avoid capture from a team of pursuers that have a speed advantage in the presence of an external flow field.

All examples in this section are subject to a linear flow field approximated by an affine function $w(X) = A(X - X_s)$, where

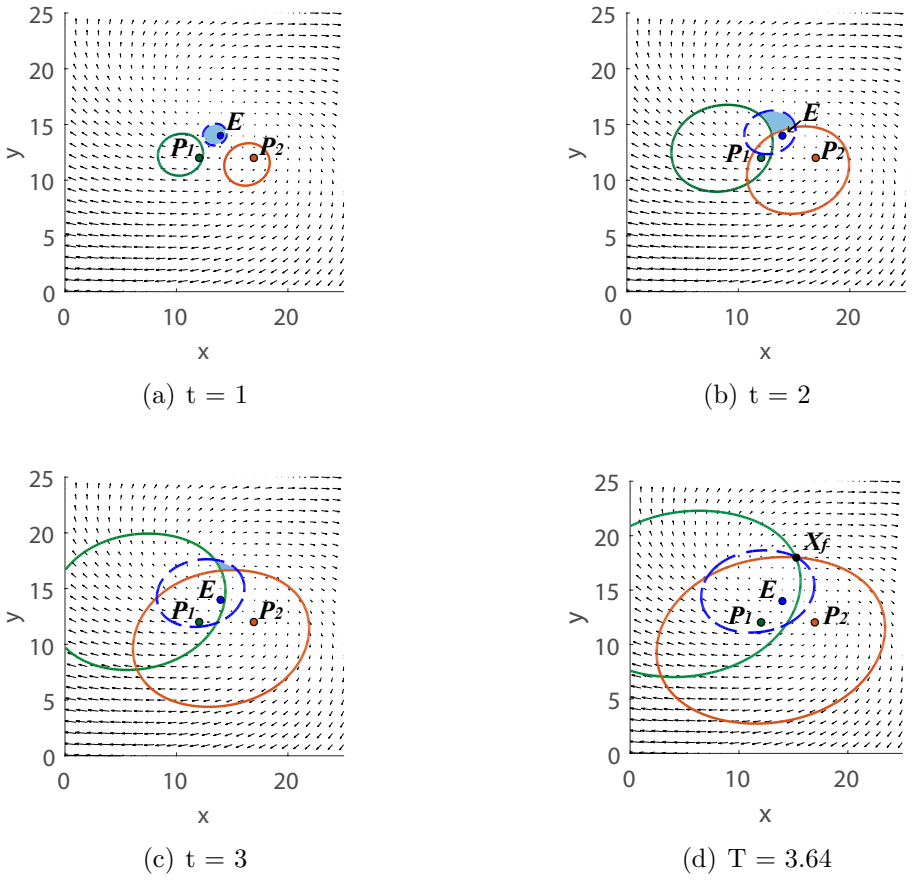
$$A = \begin{bmatrix} 0.2 & 0.3 \\ -0.15 & 0.1 \end{bmatrix}, \quad X_s = \begin{bmatrix} 15 \\ 15 \end{bmatrix}. \tag{98}$$

This wind field can be seen as a flow generated from a single singularity point located at X_s, with its characteristics captured by A. Also, the front of the reachable set of the evader is depicted in dashed blue color in each of the following examples so that it can be easily distinguished from the front of the reachable sets of the pursuers.

The evolution of the reachability fronts between two pursuers and one evader are depicted in Figure 21. The usable reachable set of the evader is illustrated by the light blue area in each of the subfigures. Notice that it is known from Theorem 4.1 and shown in this example that the capture point X_f is the point in the reachability set of the evader that is not covered by the union of the reachability sets of the pursuers until the capture time T.

For the example shown in Figure 22, pursuer P_3 turns out to be a redundant pursuer since the optimal time-to-capture of the evader is the same regardless of whether P_3 exists or not. If we remove P_3 and its corresponding reachable set at time

T, we can recover the case presented in Figure 21, that is, the outcome of the game is not changed by the presence of pursuer P_3.



(a) t = 1 (b) t = 2

(c) t = 3 (d) T = 3.64

Figure 21: Evolution of the reachability fronts between two pursuers and one evader.

Once a redundant pursuer is identified, it can be removed from the set of the active pursuers. It is important to note, however, that we cannot remove two or more redundant pursuers at the same time. Instead, we have to re-identify the redundant pursuers after one redundant pursuer is removed. The reason is that a redundant pursuer may not remain redundant after another redundant pursuer is removed. One such example is shown in Figure 23, where pursuers P_4 and P_5 guard the same shaded subset of the evader's reachable set, which is the subset of the reachable set of the evader not covered by the union of the reachable sets of pursuers P_1, P_2 and P_3. In this scenario pursuers P_4 and P_5 are both redundant. However, if we remove both at

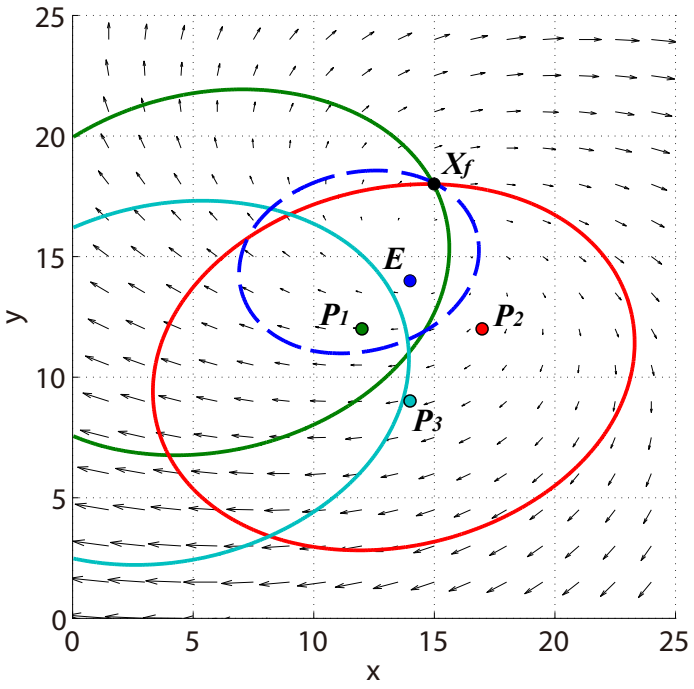


Figure 22: Level sets of three pursuers and one evader at time T. Here X_f denotes the capture point.

the same time, then the reachable set of the evader cannot be fully covered by the reachable sets of the remaining pursuers, resulting in the extension of the time-to-capture. On the other hand, if we re-identify the redundant pursuers after we have removed one of these two pursuers, the other pursuer will not be a redundant pursuer in the updated, reduced set of pursuers.

Active Pursuers and Guards. Henceforth, we assume that all pursuers are not redundant, otherwise, we can identify and remove any redundant pursuers one by one until no redundant pursuers are left as explained previously. Under this assumption, we can further divide the pursuer set into two distinct subsets. One subset consists of all the *active* pursuers, while the second subset contains pursuers that do not chase the evader, but without their presence there would be no guarantee of capture. The pursuers in the latter subset are called the *guards*. Once the capture point X_f is found, the active pursuers can be identified as the pursuers whose boundary of the

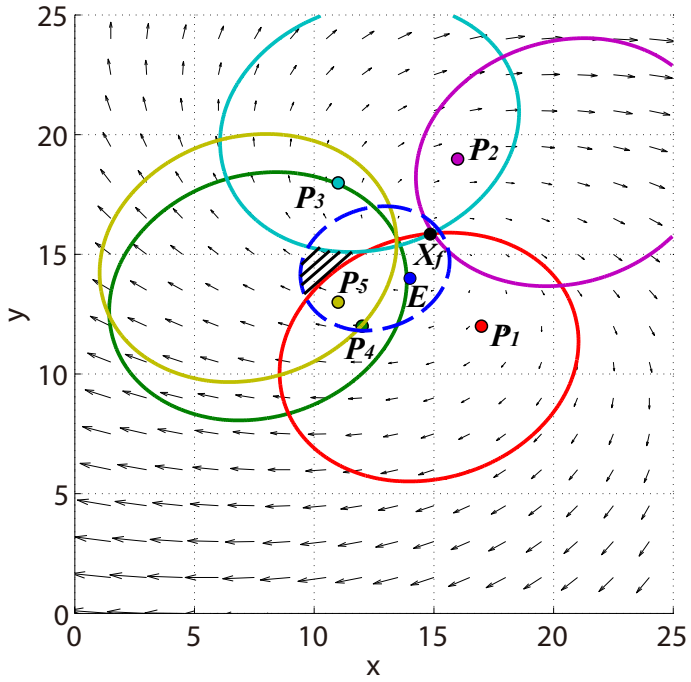


Figure 23: Level sets of five pursuers and one evader at time T. Pursuers P_4 and P_5 are each redundant pursuer by definition, but they cannot be removed together.

reachable sets at time T intersects X_f, while the rest of the pursuers are guards.

The classification of the pursuer set into active pursuers and guards can be demonstrated by the situation depicted in Figure 24. For this problem $\bar{u}^1 = \bar{u}^2 = \bar{u}^3 = \bar{u}^4 = \bar{u}^5 = 2, \bar{v} = 1$. As is shown in this figure, the reachable sets of pursuers P_1, P_2 and P_3 at time T coincide at X_f. These three pursuers need to reach X_f at time T to ensure capture of the evader. Hence, these are the active pursuers. On the other hand, pursuer P_4 cannot reach X_f within time T, but its reachable set covers a portion of the reachable set of the evader. Therefore, P_4 acts as a guard of the shaded region depicted in Figure 24 so that the evader cannot use a control to escape from that area.

It is also worth noting that if one would like to account for the possibility that the evader may not maneuver optimally, then the process of classifying active pursuers and guards should be repeated at each time step. Otherwise, a pursuer that has been

initially classified as a guard might remain inactive even if the evader moves towards it and away from the active pursuers (e.g., the shaded area in Figure 24).

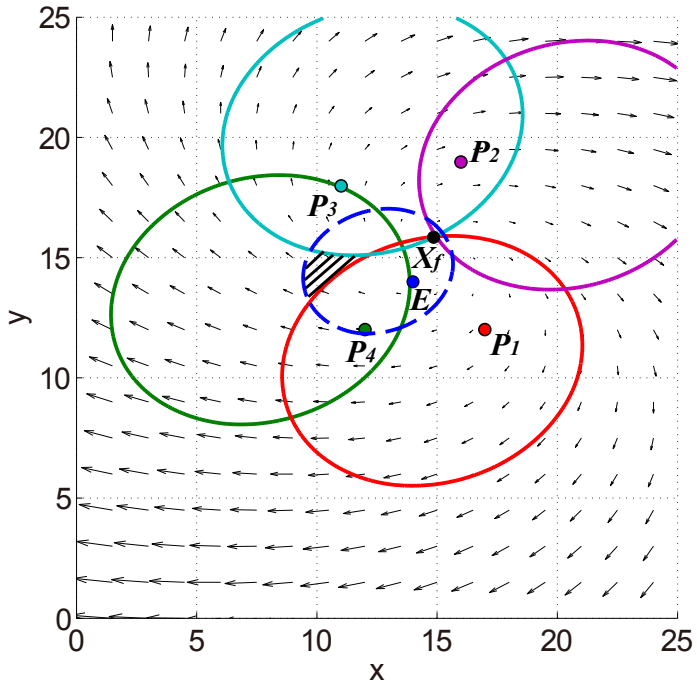


Figure 24: Level sets of four pursuers and one evader at time T. Pursuers P_1, P_2 and P_3 are active pursuers, and P_4 is a guard. Capture occurs at point X_f.

4.4.3 Time-Optimal Paths

In this section, we present a method to retrieve the optimal controls of the evaders and the active pursuers, as well as their corresponding optimal trajectories.

When we deal with multiple pursuers ($n > 1$), the first goal is to find the optimal trajectories of the active pursuers along with their corresponding optimal controls. Since the active pursuers can reach X_f at time T, it is clear that X_f resides on the boundary of their reachable sets, otherwise capture would have occurred earlier. Therefore, when the ϕ_P^i's are differentiable, the optimal trajectory for each active

pursuer satisfies

$$\frac{\mathrm{d}X_{P}^{i^*}}{\mathrm{d}t} = \bar{u}^i \frac{\nabla \phi_{P}^{i}}{|\nabla \phi_{P}^{i}|} + w(X_{P}^{i^*}, t), \quad i \in \mathcal{I}_A, \tag{99}$$

where $\mathcal{I}_A \subseteq \mathcal{I}$ denotes the index set of the active pursuers. The corresponding optimal controls of the active pursuers are thus

$$u_{P}^{i*} = \bar{u}^i \frac{\nabla \phi_{P}^{i}}{|\nabla \phi_{P}^{i}|}, \quad i \in \mathcal{I}_A. \tag{100}$$

As for the evader, there are two possible outcomes after the termination of the evolution of the reachable sets of the pursuers and the evader. One possibility is that at the terminal time T, X_f resides on $\partial\tilde{\mathcal{R}}_{E}(X_{E_0}, T)$ (or $\partial\mathcal{R}_{E}(X_{E_0}, T)$ when $\bar{v} \leq \min_{i\in\mathcal{I}} \bar{u}^i$). In this case it follows that the boundary of the reachable set of the evader is not fully covered for all $t < T$. When differentiable, the optimal trajectory of the evader is then unique and it satisfies the differential equation

$$\frac{\mathrm{d}X_{E}^{*}}{\mathrm{d}t} = \bar{v} \frac{\nabla \phi_{E}}{|\nabla \phi_{E}|} + w(X_{E}^{*}, t). \tag{101}$$

The corresponding optimal control for the evader is given by

$$u_{E}^{*} = \bar{v} \frac{\nabla \phi_{E}}{|\nabla \phi_{E}|}. \tag{102}$$

It may also happen that X_f lies in the interior of $\tilde{\mathcal{R}}_{E}(X_{E_0}, T)$ (or the interior of $\mathcal{R}_{E}(X_{E_0}, T)$ when $\bar{v} \leq \min_{i\in\mathcal{I}} \bar{u}_i$). This situation occurs when there exists $t_c \in (t_0, T)$ such that $\partial\mathcal{R}_{E}(X_{E_0}, t) \subset \bigcup_{i=1}^{n} \mathcal{R}_{P}^{i}(X_{P_0}^{i}, t)$, for all $t \in [t_c, T]$. However, some part of the interior of $\mathcal{R}_{E}(X_{E_0}, t)$ may not be covered until time T. In this case, the optimal control of the evader is not necessarily unique. In particular, the control of the evader

can be chosen from the set

$$\mathcal{U}_E^* = \left\{ u_E \in \mathcal{U}_E : X \text{ satisfies (81) and } X(\tau) \notin \bigcup_{i=1}^{n} \mathcal{R}_P^i(X_{P_0}^i, \tau), \forall \tau \in [t_0, T] \right\}. \quad (103)$$

It follows that an optimal control for the evader is valid, as long as it can bring the evader to X_f at time T, without getting captured by any of the pursuers prior to time T.

4.5 Simulation Results

We present simulation results for the multiple-pursuer/one-evader pursuit-evasion problem under a realistic flow field, and for different initial conditions for the pursuers and the evader.

We first consider a state-dependent wind field approximation generalized from the Rankine model of a vortex [29]:

$$w(X) = w_0 + \sum_{i=1}^{n_s} \omega_i A_i (X - X_{s_i}), \quad (104)$$

where

$$\omega_i = \frac{1}{\max\{r_{s_i}^2, |X - X_{s_i}|^2\}}. \quad (105)$$

In (104), n_s is the number of flow singularities, X_{s_i} is the location of the i-th flow singularity and r_{s_i} denotes the singularity radius, and A_i is a 2×2 matrix, whose structure captures the local characteristics of the i-th flow singularity. The model approximates the velocity field of a vortex with a linear vector field inside a disk and the velocity outside of the disk decreases as the inverse squared distance to the center of the disk.

We set the number of flow singularities to $n_s = 3$. The locations of the flow singularities are $X_{s_1} = [18, 18]^\intercal$, $X_{s_2} = [12, 19]^\intercal$, $X_{s_3} = [14, 12]^\intercal$, and the corresponding radii are $r_{s_1} = 3$, $r_{s_2} = 2$, $r_{s_3} = 3$, respectively. We also let $w_0 = [0.2, -0.3]^\intercal$. The local wind field matrices are given by

$$A_1 = \begin{bmatrix} 0 & 3 \\ -1.5 & 0 \end{bmatrix}, \qquad A_2 = \begin{bmatrix} 4 & 2 \\ 0 & -2 \end{bmatrix} \qquad A_3 = \begin{bmatrix} 2 & 1 \\ -2 & 2 \end{bmatrix}.$$

In the first example, we formulate a three-pursuers/one-evader problem. The three pursuers are initially located at $X^1_{P_0} = [13, 13]^\intercal$, $X^2_{P_0} = [16, 14]^\intercal$ and $X^3_{P_0} = [14, 17]^\intercal$, respectively. Their corresponding maximum speeds are given by $\bar{u}_1 = 1.5$, $\bar{u}_2 = 1.2$ and $\bar{u}_3 = 0.5$. The initial location of the evader is given by $X_{E_0} = [14, 15]^\intercal$ and its maximum speed is $\bar{v} = 1$. Note that, in this example, the maximum speed of the evader is larger than the speed of one of the pursuers. Therefore, we need to propagate the intermediate reachability front of the evader in order to recover $\mathcal{R}^\star_E(X_{E_0}, t)$.

The optimal time-to-capture is $T = 4.25$. P_1 is the only active pursuer in this case, whereas P_2 and P_3 are guards. Notice that the optimal time-to-capture in the case of only P_1 and P_2 against E is $T_{12} = 5.33$. Similarly, the optimal time-to-capture in the case of P_1 and P_3 against E is $T_{13} = 5.04$. Therefore, P_2 and P_3 are not redundant. Also, the optimal time-to-capture between P_1 and E is $T_1 = 5.38$. It can be observed from this example, and in accordance with the Definition 4.5 that the optimal time-to-capture is reduced as more (non-redundant) pursuers join the pursuit. The reachable fronts of the pursuers and the evader at time T, as well as the corresponding optimal trajectories of the active pursuer and the evader are shown in Figure 25. The red, cyan and green color curves represent the reachable fronts of the pursuers at the terminal time. As before, black arrows on the background represent the external flow field. In the figure X_f denotes the capture point.

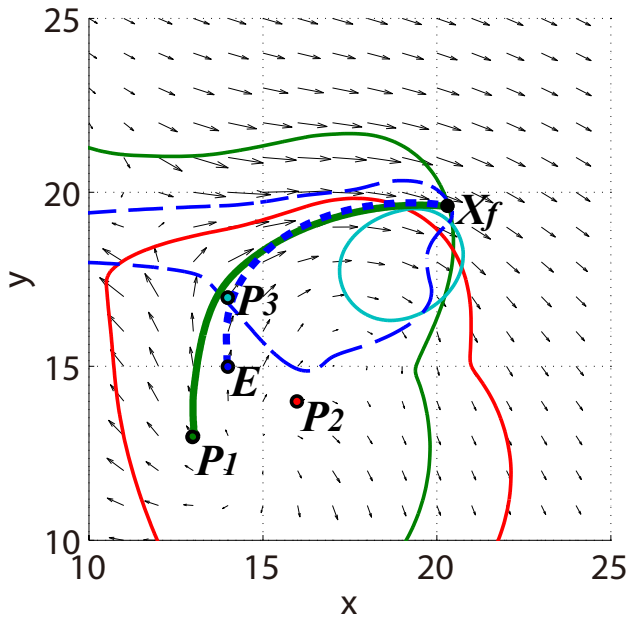


Figure 25: The optimal trajectory and the intermediate reachability front of the evader are shown in dashed blue. The optimal trajectory of the active pursuer is shown in green.

For the next example we keep all the initial conditions unchanged, but we modify the maximum speed of the third pursuer from $\bar{u}_3 = 0.5$ to $\bar{u}_3 = 2$. After this change, the condition $\bar{v} \leq \min_{i \in \mathcal{I}} \bar{u}_i$ is satisfied. By Proposition 4.1, it suffices to update $\mathcal{R}_E(X_{E_0}, t)$ instead of $\mathcal{R}^\star_E(X_{E_0}, t)$ and show condition (89). As illustrated in Figure 26, simultaneous capture of the three pursuers is required at the optimal time-to-capture $T = 1.31$. Optimal trajectories of the three pursuers are depicted to demonstrate the capture of the evader. The terminal position of the evader X_f is denoted by the black dot. The optimal trajectory of the evader is not shown since it is not unique and can be picked from (103).

Next, we consider the case where $\max_{i \in \mathcal{I}} \bar{u}_i < \bar{v}$. In particular, we consider four pursuers with maximum speed $\bar{u}^1 = \bar{u}^2 = \bar{u}^3 = \bar{u}^4 = 0.9$ and an evader with maximum speed $\bar{v} = 1$. The initial positions of the pursuers are given as $X^1_{P_0} = [13, 13]^\intercal$, $X^2_{P_0} = [15, 13]^\intercal$, $X^3_{P_0} = [15, 15]^\intercal$ and $X^4_{P_0} = [13, 15]^\intercal$, whereas the evader is initially

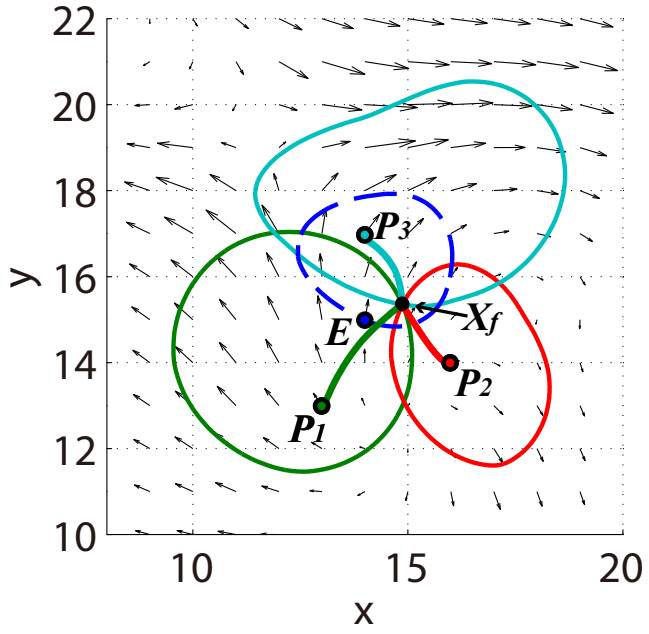


Figure 26: Optimal trajectories of the three active pursuers in green, red, cyan. Red, cyan and green closed curves represent the reachable fronts of the pursuers at the terminal time.

located at $X_{E_0} = [14, 14]^\intercal$. The flow field is the same one as in the previous example. Capture in this case occurs at $T = 1.99$ and the corresponding optimal trajectories of the active pursuers are shown in Figure 27.

The next example intents to demonstrate the effect of the flow field in the game outcome. To this end, consider a pursuit-evasion game between four pursuers and one faster evader. The initial positions of the pursuers are given as $X^1_{P_0} = [13, 13]^\intercal$, $X^2_{P_0} = [15, 13]^\intercal$, $X^3_{P_0} = [15, 15]^\intercal$ and $X^4_{P_0} = [13, 15]^\intercal$, whereas the evader is initially located at $X_{E_0} = [14, 14]^\intercal$. The maximum speed of the pursuers are set as $\bar{u}^1 = \bar{u}^2 = 0.65$, $\bar{u}^3 = \bar{u}^4 = 0.95$, and the evader's maximum speed is set to $\bar{v} = 1$. The flow field is given by $w(X) = A(X - x_s)$, where

$$A = \begin{bmatrix} -0.2 & 0.3 \\ -0.15 & -0.1 \end{bmatrix}, \quad X_s = \begin{bmatrix} 17 \\ 17 \end{bmatrix}. \tag{106}$$

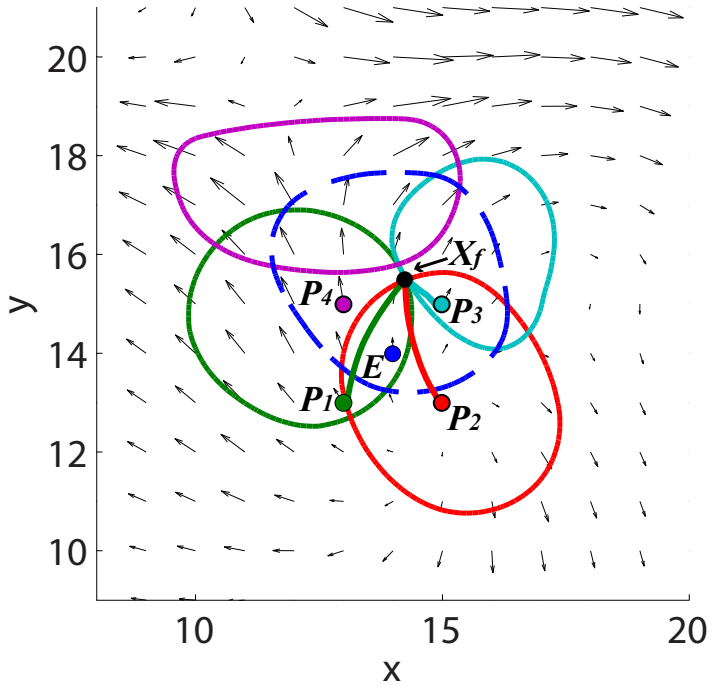


Figure 27: Optimal trajectories of the three active pursuers and the reachable fronts of the pursuers at the terminal time.

For this case, capture occurs at $T = 1.71$, as illustrated in Figure 28. In contrast, the evader escapes in the absence of an external flow field. This is demonstrated in Figure 29 where a snapshot of the level sets at $t = 4.11$ is shown. Notice that at that time instant, the evader reaches the point Y without being captured. The evader can keep avoiding capture after that time since it is faster and Y is outside the convex hull of the pursuers. This example shows that the presence of the flow field can change the outcome of the game and hence it is imperative that its effect is quantified and be included, if needed, in the game formulation.

We finally apply our algorithm to a pursuit-evasion problem taking place inside a smooth water channel with a circular island obstacle. The external flow enters the rectangle region shown in Figure 30(a) from the left edge and drift past a circular island, which induces vortices downstream. The island is centered at $[4.5, 1.5]$ and has radius 0.5. More details regarding the simulation of this flow field can be found

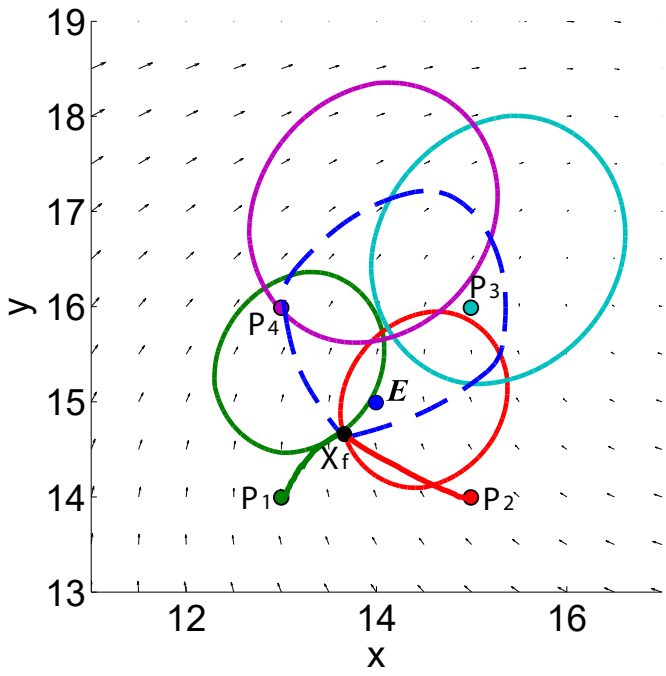


Figure 28: Reachable fronts of the pursuers and usable reachable front of the evader at the terminal time, and optimal trajectories of the two active pursuers in green and red.

in Ref. [[86]]. Through this example, we demonstrate that the proposed algorithm can handle scenarios with arbitrary spatio-temporal flow fields and that the algorithm can be naturally extended to deal with obstacles in a flow field.

The initial positions of the pursuers and the evader are set as $X^1_{P_0} = [1, 2]^\intercal$, $X^2_{P_0} = [3, 1]^\intercal$ and $X_{E_0} = [6, 1]^\intercal$, respectively. Their corresponding maximum speeds are given by $\bar{u}_1 = 4.5$, $\bar{u}_2 = 3$ and $\bar{v} = 1$. Since $\min\{\bar{u}_1, \bar{u}_2\} > \bar{v}$, we simply evolve the reachability sets of both players. When the reachability front of any one of the players encounters the obstacle, we stop the evolution of the parts on the reachability front that would otherwise go through the obstacle to ensure that the optimal path we find later on is guaranteed to be a collision-free path. Capture occurs at time $T = 1.44$ and the optimal paths of the pursuers in magenta and the evader in cyan are depicted in Figure 30(c). Their corresponding reachability fronts are also illustrated in green, red and blue colors. Snapshots of the reachability fronts of the pursuer and the evader

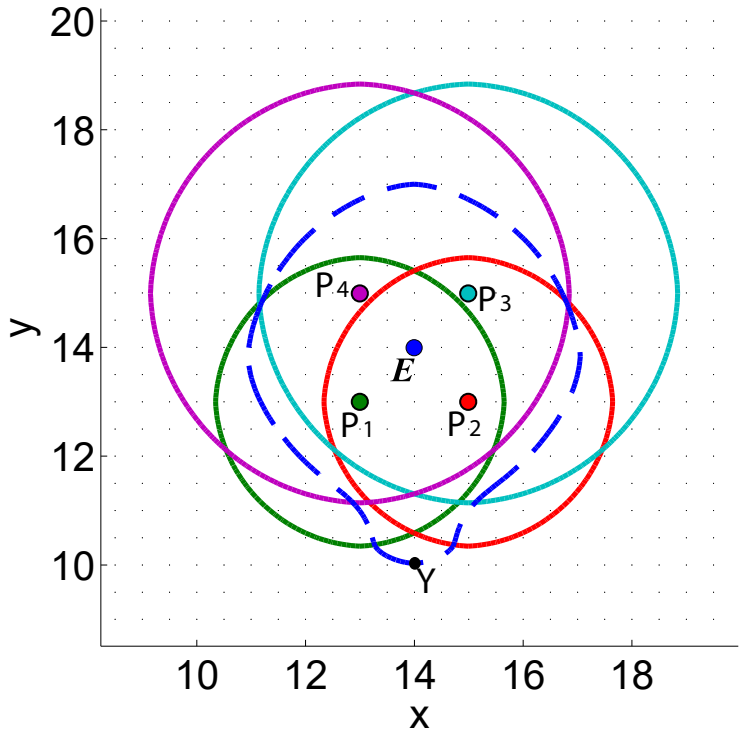
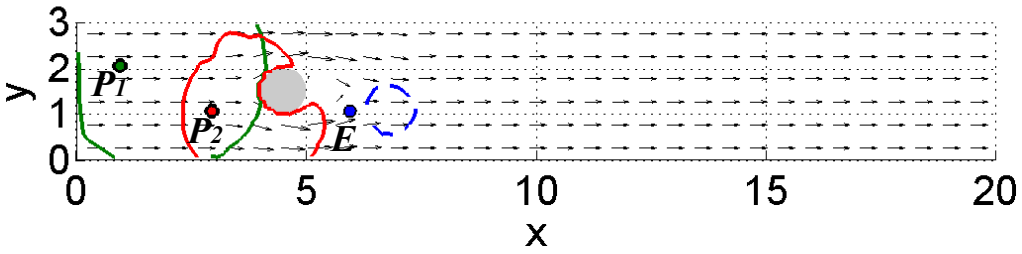


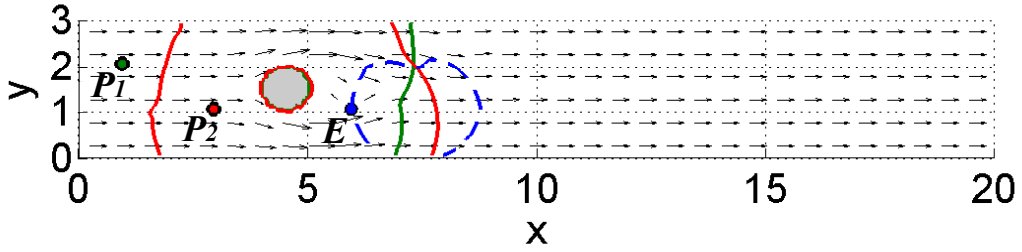
Figure 29: Reachble fronts of the pursuers and usable reachable front of the evader at $t = 4.41$. Case of faster evader without flow field.

at time $t = 0.5$ and $t = 1$ are included in Figures 30(a) and 30(b) to demonstrate the evolution of the reachability fronts.

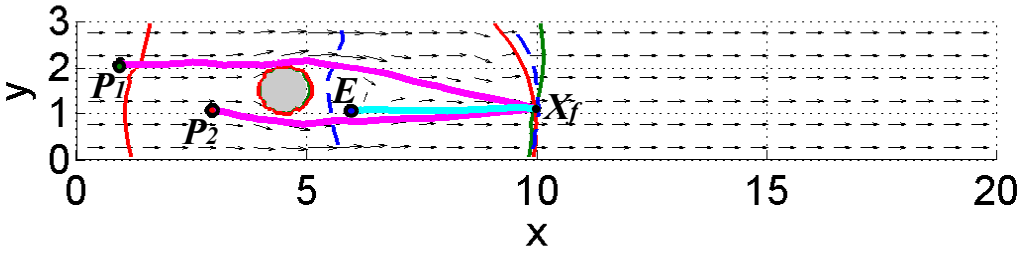
In order to demonstrate the feedback nature of the proposed strategies, in Figures 31 we show the result of a game with just two players, one pursuer and one evader, in which the evader employs a suboptimal strategy. Specifically, while the pursuer determines its control action at each instant of time using the reachability set analysis outlined in Section 4.3, in Figure 31 (left) the evader implements a constant bearing strategy given by $u_E = \bar{v}[\cos(\pi/2), \sin(\pi/2)]^\intercal$. Capture occurs at $T = 0.93$, whereas if the evader had acted optimally, capture would have occurred at $T = 1.08$, which is the value of this game. Figure 31 (right) shows another similar scenario where the evader uses the (also suboptimal) strategy $u_E = \bar{v}[\cos(\pi/4), \sin(\pi/4)]^\intercal$. In this case capture occurs at $T = 1.04$, somewhat better than before, but still less than the



(a) t = 0.5



(b) t = 1.0



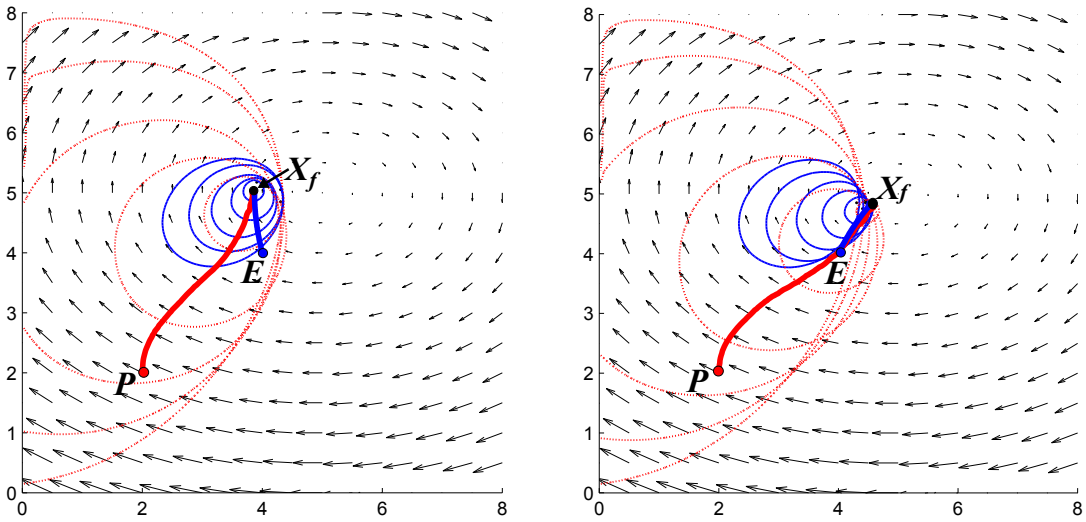
(c) T = 1.44

Figure 30: Evolution of the reachability fronts and optimal trajectories at the optimal time-to-capture. Black arrows on the background represent the time-varying external flow field.

optimal value of $T = 1.08$. For both of these examples, the flow field is affine, given by $w(X) = A(X - X_s)$, where

$$A = \begin{bmatrix} 0 & 0.3 \\ -0.15 & 0 \end{bmatrix}, \quad X_s = \begin{bmatrix} 5 \\ 5 \end{bmatrix}. \tag{107}$$

For this example the maximum speeds are $\bar{u} = 4$ and $\bar{v} = 1$ and the initial conditions are $X_{P_0} = [2, 2]^\intercal$ and $X_{E_0} = [4, 4]^\intercal$ for the pursuer and the evader, respectively.



(a) (b)

Figure 31: Evolution of the reachability fronts and optimal trajectories at the optimal time-to-capture. In this case evader plays suboptimally.

CHAPTER V

PURSUIT-EVASION GAMES UNDER 3-DIMENSIONAL FLOW FIELDS

5.1 Introduction

There is a plethora of work in the literature that deal with multiplayer pursuit-evasion problems. In general, these are difficult problems to solve analytically, owing to the delicate nature of the problem itself [62]. Different techniques have been utilized to simplify and subsequently deal with this problem. Single integrator kinematics are commonly assumed in most multiplayer pursuit problems. Under such assumption, the pursuit problem can be further divided into group-pursuit and relay-pursuit problems. A group-pursuit problem refers to the case where several (or all) pursuers act simultaneously to capture the targets, whereas a relay-pursuit problem considers the case where for each target, only one pursuer actively chases it. In the category of group-pursuit, conditions for target interception between multiple pursuers and one evader in the simplest form is studied by Pshenichnyi in [123]. The result is extended to simultaneous k-capture in [19] and ϵ-capture in [70] where capture occurs when some pursuer is within an ϵ distance from the evader. Conditions for guaranteed evasion and the corresponding evading strategies are studied in [25] and [61]. Pursuit-evasion between one evader and countably many pursuers is investigated in [60]. In terms of relay-pursuit problems, generalized Voronoi diagrams have been utilized to assign active pursuers dynamically to capture the evader, capture therefore is achieved through a relay of the pursuers in a multi-pursuer/one-evader problem [9, 14]. The idea has also been applied in [141] to deal with known environmental disturbance and in [39] for cooperative relay tracking of targets. Some results exist for cases with

more general dynamics for the agents (pursuers/evaders), but extra assumptions are made for the problem to be tractable. Pursuit-evasion between a group of pursuers and one evader with linear time-varying dynamics is studied in [122] and later on was generalized to the case of multiple evaders [15]. Group pursuit of a target under the so-called 'soft' capture where capture occurs only when at least one pursuer and the target have identical orientation, velocity and acceleration is investigated in [121] under linear time invariant dynamics. A multiplayer extension of the classical Homicidal Chauffeur game [62] is discussed in [21], where a chain formation of faster, yet less maneuverable, pursuers is utilized to ensure capture of a single slower, but agile evader. However, no optimality analysis of such strategy is given.

In this Chapter, we consider a multi-pursuer/multi-evader pursuit-evasion problem of agents having the single integrator kinematics under known external dynamic flow field in the 3D Euclidean space. The reason we coinsider dynamic environmental conditions is that the presence of time-varying or spatially-varying flows may significantly affect the vehicle's motion and corresponding strategy. This is the case, for example, when either the pursuers or the evaders (or both) are small autonomous underwater vehicles (AUV) or small unmanned aerial vehicles (UAV). We solve this problem through a two-step approach. We first recast the problem as a partition of the pursuer set and assign each pursuer to an evader. Subsequently, the original problem is separated into sub-games between multiple pursuers and one evader. We find the optimal trajectories of the agents in each sub-game through a reachable set method. Specifically, we utilize the level set method [111, 133] to generate the reachable sets of both the evader and the pursuers and to retrieve the corresponding optimal control actions at the current location of the agents by backward propagation of their respective reachable sets.

5.2 Problem Formulation

We consider a pursuit-evasion game under an external flow field between n pursuers and m evaders. Henceforth, we will refer to the pursuers and the evader collectively as "agents". The dynamics of the pursuers P_i, $i \in \mathcal{I}$, and the evaders E_j, $j \in \mathcal{J}$, where $\mathcal{I} = \{1, 2, \ldots, n\}$ and $\mathcal{J} = \{1, 2, \ldots, m\}$, are given by

$$\dot{X}_P^i = u_P^i + w(X_P^i, t), \qquad X_P^i(t_0) = X_{P_0}^i, \tag{108}$$

$$\dot{X}_E^j = u_E^j + w(X_E^j, t), \qquad X_E^j(t_0) = X_{E_0}^j, \tag{109}$$

where $X_P^i \in \mathcal{D} \subset \mathbb{R}^3$ denotes the position of the i-th pursuer and $X_E^j \in \mathcal{D} \subset \mathbb{R}^3$ denotes the position of the j-th evader. Here $\mathcal{D}$ denotes a compact subset of $\mathbb{R}^3$ and u_P^i is the control input (i.e., velocity) of the i-th pursuer such that $u_P^i \in \mathcal{U}_P^i$, for all $i \in \mathcal{I}$. The set $\mathcal{U}_P^i$ consists of all piecewise continuous functions whose range is included in the set $U_P^i = \{u^i \in \mathbb{R}^3, |u^i| \leqslant \bar{u}^i\}$, where $|\cdot|$ represents the Euclidean norm and $\bar{u}^i$, $i \in \mathcal{I}$, are constants. If $u_P^i \in \mathcal{U}_P^i$, we say that u_P^i is an *admissible* control for the i-th pursuer. Similarly, u_E^j is the control input of the j-th evader such that $u_E^j \in \mathcal{U}_E^j$, where $\mathcal{U}_E^j$ consists of all piecewise continuous functions whose range is included in the set $U_E^j = \{v^j \in \mathbb{R}^3, |v^j| \leqslant \bar{v}^j\}$. We say that u_E^j is an *admissible* control of the j-th evader if $u_E^j \in \mathcal{U}_E^j$. In (80), $w(X, t) \in \mathbb{R}^3$ represents an exogenous dynamic flow, but it could also represent an endogenous drift owing to the nonlinear dynamics of the agent. It is reasonable to assume that the magnitude of this flow is bounded from above by some constant, that is, there exists a constant $\bar{w}$ such that $|w(X, t)| \leq \bar{w}$, for all $(X, t) \in \mathcal{D} \times [t_0, \infty)$.

The pursuers aim to intercept the evaders and minimize the overall time-to-capture if possible, whereas the evaders prefer to avoid capture for as long as possible. The game begins at time $t = t_0$ with initial positions $X_{P_0}^i, (i \in \mathcal{I})$, and $X_{E_0}^j, (j \in \mathcal{J})$, for the pursuers and the evaders, respectively, and terminates when each evader coincides

with at least one of the pursuers, in which case capture occurs. That is, capture implies that for each $j \in \mathcal{J}$, there exists $i^j \in \mathcal{I}$ and a terminal time $T^j \geq t_0$ such that $X_P^{i^j}(T^j) = X_E^j(T^j)$. The overall time-to-capture is then denoted as $T^o = \max_{j \in \mathcal{J}} T^j$. Equivalently, the game terminates if, for each $j \in \mathcal{J}$ and any admissible control of the j-th evader $u_E^j \in \mathcal{U}_E^j$, there exists a set of admissible controls $(u_P^{k_1}, \ldots, u_P^{k_{n_j}}) \in \mathcal{U}_P^{k_1} \times \cdots \times \mathcal{U}_P^{k_{n_j}}$ such that $X_P^i(T^j) = X_E^j(T^j)$ for some $i \in \{k_1, \ldots, k_{n_j}\}$ and some time $T^j \geq t_0$.

We assume in this paper that each pursuer can capture at most one evader, that is, once a pursuer captures an evader, it does not re-enter the game to chase another evader. Such assumption is valid in cases where the pursuers are disposable (such as missiles) or the amount of pursuers is abundant. Subject to this assumption, and in accordance with the previous notion of capture, it can be seen that a necessary condition for the game to terminate is $n > m$. In other words, there should be more pursuers than evaders for all the evaders to be captured. Henceforth, we assume that this condition holds, that is, there are more pursuers than evaders.

The original problem is intractable due to the existence of the external flow field and the large number of possible pairings between the evaders and the pursuers. Therefore, instead of attempting to solve this problem head on, we propose to decompose the problem into two steps. In the first step, we apply a partition to the set of pursuers by assigning each pursuer to an evader at the beginning of the game. In the second step, the pursuit-evasion game between each evader and its assigned (paired) pursuers is solved to obtain the optimal trajectories and controls of the pursuers and the evader. The problems in these two steps are formally stated as follows.

Problem 5.1 *Given the system described by (108) and (109), find a partition* $\{\mathcal{I}_j, j \in \mathcal{J}\}$ *of the pursuer set* $\mathcal{I}$ *such that* $i \in \mathcal{I}_j$ *if* $T^*(X_{P_0}^i, X_{E_0}^j) \leq T^*(X_{P_0}^i, X_{E_0}^k)$, $\forall k \in \mathcal{J}$, *where* $T^*(X_{P_0}^i, Y)$ *denotes the minimum time-to-reach from* $X_{P_0}^i$ *to* Y *for some* $Y \in \mathbb{R}^3$ *under the dynamics of the* i*-th pursuer.*

Problem 5.2 *For some $j \in \mathcal{J}$, consider a pursuit-evasion game in the presence of an external flow field between multiple pursuers $P_i, i \in \mathcal{I}_j$ and one evader E_j. The dynamics of the pursuers are given by (108), where $i \in \mathcal{I}_j$. The pursuers aim to intercept the evader whose dynamics is given by (109). Without loss of generality, let $\mathcal{I}_j = \{1, \ldots, k\}$ and let E denote E_j for abbreviation. The game begins at initial time $t_0 = 0$ with initial positions $\bar{X}_0 = [X_{E_0}^\intercal, X_{P_0}^{1\intercal}, X_{P_0}^{2\intercal}, \ldots, X_{P_0}^{k\intercal}]^\intercal$, and terminates when at least one of the pursuers reaches the location of the evader. The objective is to find the open-loop representation of the optimal strategies of the pursuers and the evader.*

5.3 Problem Analysis

5.3.1 Pursuer Assignment

As mentioned in Section 5.2, we decompose the problem into two steps, where the first step assigns each pursuer to an evader, and the second step solves the pursuit-evasion game between each evader and its assigned pursuers in the precense of the external flow field.

In this subsection, we focus on the first step, that is, obtaining a pursuer assignment such that each pursuer is paired with an evader. To this end, we introduce the following Proposition.

Proposition 5.1 *Given the system described by (108) and (109), and the partition $\{\mathcal{I}_j, j \in \mathcal{J}\}$ of the pursuer set $\mathcal{I}$ introduced in Problem 5.1, then given $i \in \mathcal{I}$, i is within the partition $\mathcal{I}_j$ if $t_j^i = \min_{k \in \mathcal{J}} t_k^i$, where $t_k^i := \inf\{t \in [t_0, \infty) : X_{E_0}^k \in \mathcal{R}_P^i(X_{P_0}^i, t)\}$, that is, t_k^i is the first time such that $X_{E_0}^k$ is inside $\mathcal{R}_P^i(X_{P_0}^i, t)$.*

proof 5.3.1 *It can be easily seen that the proposition is true as long as $t_j^i = T^*(X_{P_0}^i, X_{E_0}^j)$. In other words, let $\{\mathcal{I}_j', j \in \mathcal{J}\}$ be a partition of the pursuer set $\mathcal{I}$ such that $i \in \mathcal{I}_j'$ if $t_j^i = \min_{k \in \mathcal{J}} t_k^i$. Then $\{\mathcal{I}_j'\}$ coincides with $\{\mathcal{I}_j\}$ when $t_j^i = T^*(X_{P_0}^i, X_{E_0}^j)$. Therefore, it suffices to show that $t_j^i = T^*(X_{P_0}^i, X_{E_0}^j)$.*

To this end, consider the definition of $\mathcal{R}_P^i(X_{P_0}^i, t)$. *Since* $\mathcal{R}_P^i(X_{P_0}^i, t)$ *contains all the points the* i*-th pursuer can reach at time* t *starting from* $X_{P_0}^i$, *and since* t_j^i *is defined as the first time such that* $X_{E_0}^j \in \mathcal{R}_P^i(X_{P_0}^i, t_j^i)$, *it can be seen that* t_j^i *is the minimum time-to-reach from* $X_{P_0}^i$ *to* $X_{E_0}^j$. *In other words,* $t_j^i = T^*(X_{P_0}^i, X_{E_0}^j)$. *If ad absurdum assume that* $t_j^i \neq T^*(X_{P_0}^i, X_{E_0}^j)$, *then we have two cases to consider. One is* $t_j^i < T^*(X_{P_0}^i, X_{E_0}^j)$ *and the other is* $t_j^i > T^*(X_{P_0}^i, X_{E_0}^j)$. *Suppose that* $t_j^i < T^*(X_{P_0}^i, X_{E_0}^j)$. *It clearly contradicts with the fact that* $T^*(X_{P_0}^i, X_{E_0}^j)$ *is the minimum time-to-reach from* $X_{P_0}^i$ *to* $X_{E_0}^j$. *On the other hand, if* $t_j^i > T^*(X_{P_0}^i, X_{E_0}^j)$, *it follows that* $X_{E_0}^j \in \mathcal{R}_P^i(X_{P_0}^i, T^*(X_{P_0}^i, X_{E_0}^j))$ *which happens at a time prior to* t_j^i. *This contradicts the definition* $t_j^i = \inf\{t \in [t_0, \infty) : X_{E_0}^j \in \mathcal{R}_P^i(X_{P_0}^i, t)\}$. *Finally, the assumption* $t_j^i \neq T^*(X_{P_0}^i, X_{E_0}^j)$ *leads to a contradition and thus* $t_j^i = T^*(X_{P_0}^i, X_{E_0}^j)$.

The previous proposition provides us with a guideline to obtain the partition of the pursuer set through propagation of the reachable sets of the pursuers. Specifically, we propose the algorithm to find the partition $\{\mathcal{I}_j, j \in \mathcal{J}\}$ of the pursuer set $\mathcal{I}$ as follows.

Pursuer Assignment

For each pursuer $i \in \mathcal{I}$:

a) Propagate its reachable set from the initial position and time.

b) Stop the propagation when an evader (say E_k) enters the reachable set of this pursuer.

c) Place i in the cell $\mathcal{I}_k$.

We hereby assume that there is always a sufficient number of pursuers such that $\mathcal{I}_j \neq \emptyset$, for all $j \in \mathcal{J}$. This assumption ensures that each evader can be actively chased by at least one of the pursuers.

5.3.2 Multiple-Pursuers/One-Evader Game

In this subsection, we utilize the reachable set analysis to find the conditions for capture and to derive the corresponding optimal controls and trajectories of the agents in the sub-game between multiple pursuers and one evader. Reachable sets provide a quick snapshot of all possible future trajectories of the agent and thus succinctly encode all possible future positions of the agent under any possible control action. Knowledge of the reachable sets of the pursuer and the evader can then be used to draw conclusions about the potential meeting of the two at some future time (or not). In this paper we use this intuition behind the information conveyed by the reachable set of each player to solve the pursuit-evasion problem under minimal assumptions about the maximum number of players and the environment they operate in. Since the computation of the reachable sets for each pursuer can be done independently from the other pursuers, the proposed method is decentralized and scales well with the number of pursuers, something that is not the case with more traditional approaches that require directly the solution of a HJI partial differential equation [17].

The evader plays a game with the set of pursuers and we would like to find the set of terminal points of all "safe" evader trajectories, that is, the set of all terminal points of the evader at time t, whose trajectories do not pass through any reachable sets of the pursuers at *any* time in the interval $[t_0, t]$. The formal definition of such set is given as follows.

Definition 5.1 *Given the reachable sets of the pursuers, we define the* usable reachable set *of the evader at time $t \geq t_0$ as*

$$\mathcal{R}^{\star}_{E}(X_{E_0}, t) = \left\{ X \in \mathcal{D} : X = X_{E}(t) \text{ and } X_{E}(\tau) \notin \bigcup_{i=1}^{n} \mathcal{R}^{i}_{P}(X^{i}_{P_0}, \tau), \forall \tau \in [t_0, t] \right\}. \tag{110}$$

From this definition, it is clear that $\mathcal{R}^{\star}_{E}(X_{E_0}, t) \subseteq \mathcal{R}_{E}(X_{E_0}, t)$. The definition

implies that $\mathcal{R}_E^\star(X_{E_0}, t)$ is the set of all terminal points of the evader at time t, whose trajectories do not pass through any reachable sets of the pursuers at *any* time in the interval $[t_0, t]$.

The following theorem is originated from this definition and is used later on to develop an efficient numerical algorithm for solving the pursuit-evasion game between multiple pursuers and one evader in the presence of a known dynamic flow field.

Theorem 5.1 *Let $T = \inf\{t \in [t_0, +\infty) : \mathcal{R}_E^\star(X_{E_0}, t) = \emptyset\}$. If $T < \infty$, then capture is guaranteed for any time greater than T, while the evader can always escape within a time smaller than T. Hence, T is the time-to-capture if both players play optimally. Furthemore, let X_f denote the location where the evader is captured by at least one of the pursuers. Then we have that*

$$X_f \in \mathcal{X} = \left\{ X \in \mathcal{D} : X = X_E(T) \text{ and } X_E(\tau) \notin \bigcup_{i=1}^{n} \mathcal{R}_P^i(X_{P_0}^i, \tau), \forall \tau \in [t_0, T) \right\}. \quad (111)$$

proof 5.3.2 *The proof follows similarly to the proof of Theorem 4.1, and it is omitted.*

The previous theorem gives us a criterion for capture of the evader, that is, capture is guaranteed when $\mathcal{R}_E^\star(X_{E_0}, t) = \emptyset$ for some $t \in [t_0, \infty)$. Also notice from (110) that, in general, $\mathcal{R}_E^\star(X_{E_0}, t)$ can be generated by keeping track of the reachable sets of the pursuers and the evader at all time prior to the capture time. However, when we add some constraints with respect to the speeds of the players, we can replace this criterion with an instantaneous condition which is easier to check and implement. Before we state and prove this result, the following lemma is needed.

Lemma 5.1 *Let $\bar{u} = \bar{u}_i$ for all $i = 1, \ldots, n$ denote the maximum speed of a pursuer, and let $\bar{v}$ denote the maximum speed of the evader, respectively, and assume that $\bar{v} < \bar{u}$. If there exists some time $t_s \geq t_0$ such that $X_E(t_s) \in \mathcal{R}_P(X_{P_0}, t_s)$, then $X_E(t) \in \mathcal{R}_P(X_{P_0}, t)$ for all $t \geq t_s$.*

proof 5.3.3 *The proof is identical to Lemma 4.1.*

This lemma essentially states that when the maximum speed of the evader is smaller than the maximum speed of all the pursuers, then once the evader enters the reachable set of a pursuer, it can never leave this reachable set. We are now ready to present the simplified condition on the capture of the evader as follows.

Proposition 5.2 *When* $\bar{v} \leq \min_{i \in \mathcal{I}} \bar{u}_i$*, the set* $\mathcal{R}^\star_{E}(X_{E_0}, t)$ *satisfies*

$$\mathcal{R}^\star_{E}(X_{E_0}, t) = \mathcal{R}_{E}(X_{E_0}, t) \backslash \bigcup_{i=1}^{n} \mathcal{R}^i_{P}(X^i_{P_0}, t), \tag{112}$$

for all $t \geq t_0$*. In such cases, the condition* $\mathcal{R}^\star_{E}(X_{E_0}, t) = \emptyset$ *is equivalent to the condition*

$$\mathcal{R}_{E}(X_{E_0}, t) \subseteq \bigcup_{i=1}^{n} \mathcal{R}^i_{P}(X^i_{P_0}, t). \tag{113}$$

proof 5.3.4 *The proof is identical to the proof of Proposition 4.1, and is thus omitted.*

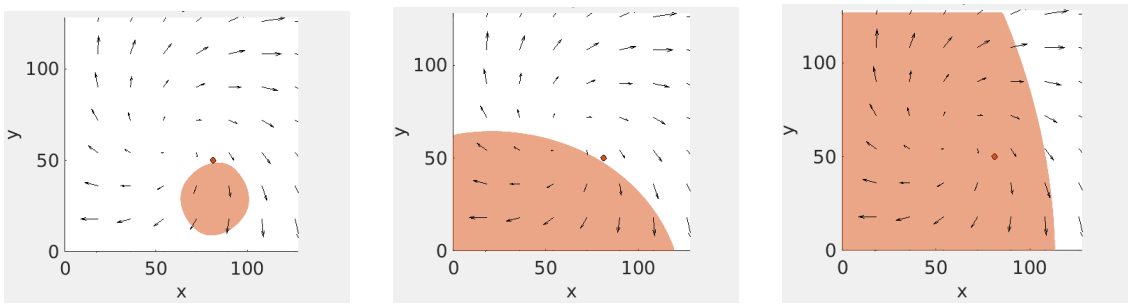
This proposition essentially shows that in the case where $\bar{v} \leq \min_{i \in \mathcal{I}} \bar{u}_i$, the optimal time-to-capture can be determined as the first time instant when the union of the reachable sets of the pursuers $\bigcup_{i=1}^{n} \mathcal{R}^i_{P}(X^i_{P_0}, \tau)$ completely covers the reachable set of the evader $\mathcal{R}_{E}(X_{E_0}, t)$.

5.4 Numerical Construction

5.4.1 Narrow Band Level Set Method

Upon constructing the reachable sets of the pursuers and the evaders by numerically solving their corresponding HJ equations, one may be tempted to utilize the fast marching method [133], which is a well-known and efficient approach to track the evolution of a closed surface as a function of time with speed in the normal direction along the propagating surface. However, one prerequisite to using the fast marching method is that each grid point should not be revisited after the level set propogation.

This prerequisite is not satisfied in our problem due to the existence of the external flow field. Such phenomenon can be demonstrated in the following 2-dimensional example [12]. Reachable sets of the pursuer at time $t = 1, 3, 5$ are depicted in Figure 32. As shown in Figure 32(a), the pursuer start at $P_0 = [81, 50]^\intercal$ at time $t = 0$. Therefore, $P_0 \in \mathcal{R}_P(P_0, 0)$. However, $P_0 \notin \mathcal{R}_P(P_0, 1)$, as is depicted in Figure 32(a), where the orange region represents the reachable set of the pursuer at time $t = 1$. The initial position of the pursuer is still outside the reachable set at $t = 3$, as shown in Figure 32(b). Later on, at $t = 5$, P_0 reenters the reachable set $\mathcal{R}_P(P_0, 5)$, as illustrated in Figure 32(c). This indicates that P_0 is revisited by the reachable set.



(a) (b) (c)

Figure 32: a) Reachable set of pursuer at time $t = 1$. b) Reachable set of pursuer at time $t = 3$. c) Reachable set of pursuer at time $t = 5$.

The previous observation forces us to utilize a more sophisticated method to construct the reachable sets while retaining some level of efficieny. In particular, we apply the narrow band level set method [111,133]. The level set method evolves the reachability front by embedding it as a hyper-surface in a higher dimension, where time is the additional dimension. The level set formulation provides an implicit representation of the front, which offers several advantages over an explicit representation [111,133]. The standard level set method would initialize with a mesh grid and update the value on every grid point to correspond to the motion of the surface. Instead of this full matrix approach, we adopt a more efficient method known as the narrow band approach that perform update of value in a neighborhood of the zero level set. This

approach can save a lot of computation effort with regard to the operation on the entire computational domain. Furthermore, the time step in the narrow band method can be adaptively chosen with respect to the maximum velocity field achieved within the narrow band.

We utilize signed distance function as per choice of implicit function to represent curves. Its definition is given in (93). Assuming that the signed distance function with respect to the i-th pursuer reachable set $\mathcal{R}_{P}^{i}(X_{P_0}^{i}, t)$ at time t is $\phi_{P}^{i}(X, t)$, then the evolution of the reachable front $\partial\mathcal{R}_{P}^{i}(X_{P_0}^{i}, t)$ is governed by the viscosity solution of the Hamilton-Jacobi equation [82, 85, 86]

$$\frac{\partial \phi_{P}^{i}(X,t)}{\partial t} + \bar{u}|\nabla \phi_{P}^{i}(X,t)| + \nabla \phi_{P}^{i}(X,t) w(X,t) = 0, \quad X \in \mathcal{D} \subset \mathbb{R}^3, \tag{114}$$

with initial condition $\phi_{P}^{i}(X, t_0) = |X - X_{P_0}^{i}|$. Note that $\mathcal{R}_{P}^{i}(X_{P_0}^{i}, t) = \{X \in \mathcal{D} : \phi_{P}^{i}(X, t) \leq 0\}$ and $\partial\mathcal{R}_{P}^{i}(X_{P_0}^{i}, t) = \{X \in \mathcal{D} : \phi_{P}^{i}(X, t) = 0\}$.

Similarly, the reachable front $\partial\mathcal{R}_{E}(X_{E_0}, t)$ of the evader is computed by solving the Hamilton-Jacobi equation

$$\frac{\partial \phi_{E}(X,t)}{\partial t} + \bar{v}|\nabla \phi_{E}(X,t)| + \nabla \phi_{E}(X,t) w(X,t) = 0, \quad X \in \mathcal{D} \subset \mathbb{R}^3, \tag{115}$$

with initial condition $\phi_{E}(X, t_0) = |X - X_{E_0}|$, where $\phi_{E}(X, t)$ is the signed distance function with respect to the reachable set $\mathcal{R}_{E}(X_{E_0}, t)$ of the evader at time t.

In the case where the condition in Proposition 5.2 is not satisfied, we need to track $\partial\mathcal{R}_{E}^{\star}(X_{E_0}, t)$ in order to determine the optimal time-to-capture. Instead of propagating $\partial\mathcal{R}_{E}^{\star}(X_{E_0}, t)$ directly, we propagate an intermediate reachable front $\partial\tilde{\mathcal{R}}_{E}(X_{E_0}, t)$ which can be computed by solving the following modified version of the Hamilton-Jacobi equation

$$\frac{\partial \tilde{\phi}_{E}(X,t)}{\partial t} + \tilde{v}(t)\,|\nabla \tilde{\phi}_{E}(X,t)| + \nabla \tilde{\phi}_{E}(X,t) w(X,t) = 0, \quad X \in \mathcal{D} \subset \mathbb{R}^3, \tag{116}$$

where

$$\tilde{v}(t) = \begin{cases} \min\limits_{i \in \mathcal{I}} \bar{u}_i, & \text{if } \bigcup\limits_{i=1}^{n} \phi_P^i(X,t) < 0, \\ \bar{v}, & \text{otherwise,} \end{cases} \tag{117}$$

and initial condition $\tilde{\phi}_E(X, t_0) = |X - X_{E_0}|$.

The main idea here is to treat the reachable sets of the pursuers as moving obstacles and propagate $\tilde{\mathcal{R}}_E(X_{E_0}, t)$ with the maximum speed of the evader $\bar{v}$ for the parts that fall outside the union of the reachable sets of the pursuers, and to keep pace with the propagation of the reachable set of the slowest pursuer when the front of the evader enters any reachable set of the pursuers. By doing this, we can make sure that the front of the evader does not grow out of the union of the reachable sets of the pursuers. The parts of the reachable front of the evader that do not encounter the reachable sets of the pursuers remain unaffected by the change of speed from $\bar{v}$ to $\tilde{v}$, since these changes are only performed for points inside the reachable sets of the pursuers.

Let $\tilde{\mathcal{R}}_E(X_{E_0}, t) = \{X \in \mathcal{D} : \tilde{\phi}_E(X,t) \leq 0\}$. At every time instant t, by construction, $\tilde{\mathcal{R}}_E(X_{E_0}, t)$ excludes all the points X such that $X = X_E(t)$ and $X \notin \bigcup\limits_{i=1}^{n} \mathcal{R}_P^i(X_{P_0}^i, t)$, while $X \in \bigcup\limits_{i=1}^{n} \mathcal{R}_P^i(X_{P_0}^i, \tau)$, for some $\tau \in [t_0, t)$. It follows that

$$\mathcal{R}_E^\star(X_{E_0}, t) = \tilde{\mathcal{R}}_E(X_{E_0}, t) \backslash \bigcup_{i=1}^{n} \mathcal{R}_P^i(X_{P_0}^i, t).$$

Moreover, since $\mathcal{R}_P^i(X_{P_0}^i, t) = \{X \in \mathcal{D} : \phi_P^i(X,t) \leq 0\}$, the usable reachable set of the evader can also be represented in a form that is more suitable for numerical calculations, that is,

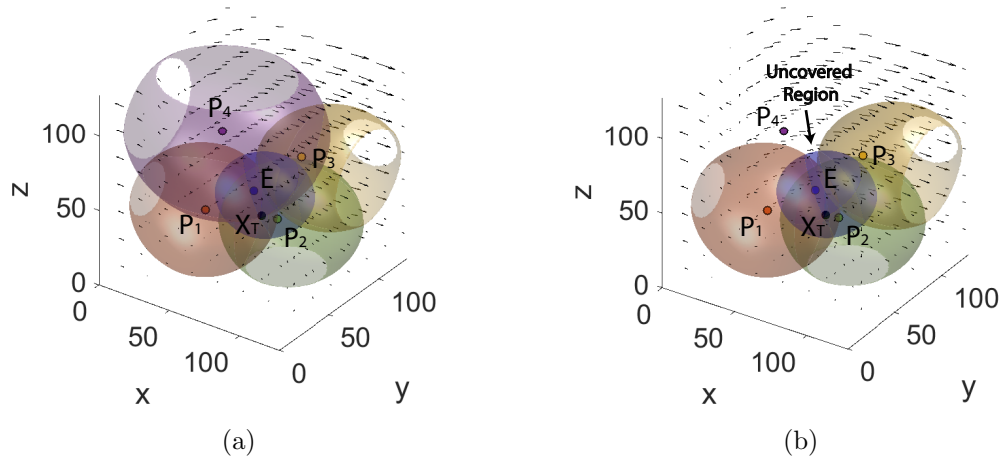
$$\mathcal{R}_E^\star(X_{E_0}, t) = \{X \in \mathcal{D} : \tilde{\phi}_E(X,t) \leq 0 \text{ and } \bigcup_{i=1}^{n} \phi_P^i(X,t) \geq 0\}.$$

5.4.2 Pursuer Classification

For problems with a large number of pursuers it is very likely that not all pursuers are involved in the optimal capture. In certain applications, such as when the pursuers are subject to energy or fuel limitations, or when they play a dual role as pursuers and guards of a certain region of responsibility, it may be beneficial that some of the pursuers remain inactive. In group pursuit problems involving several pursuers, we may therefore classify the pursuers according to their level of involvement as either active or inactive.

In particular, we can divide the pursuer set into two distinct subsets. One subset consists of all the *active* pursuers, while the second subset contains the *inactive* pursuers that do not chase the evader. We also refer to the pursuers in the latter subset as *guards*, since they remain inactive during the optimal pursuit when the evader stays along its optimal trajectory or join the chase if the evader plays suboptimally and deviates from its original trajectory. Once the capture point X_f is found, the active pursuers can be identified as the pursuers whose boundary of the reachable sets at time T intersects X_f, while the rest of the pursuers are guards.

The classification of the pursuer set into active pursuers and guards can be demonstrated by the situation depicted in Figure 33(a). As is shown in this figure, the reachability fronts of pursuers P_1, P_2 and P_3 at the capture time T coincide at the terminal position X_f. These three pursuers need to reach X_f at time T to ensure capture of the evader. Hence, these are the active pursuers. On the other hand, pursuer P_4 cannot reach X_f within time T, but its reachable set covers a portion of the reachable set of the evader. This can been seen in Figure 33(b) where the reachable set of pursuer P_4 is removed to show that the reachable set of the evader is not fully covered by the union of reachable sets of P_1, P_2 and P_3 at time T. Therefore, P_4 acts as a guard.



(a) (b)

Figure 33: a) Level sets of four pursuers and one evader at time T. b) Level sets of three pursuers and one evader at time T. Level set of pursuer P_4 is removed to show the reachable set of the evader is not fully covered by the union of reachable sets of P_1, P_2 and P_3.

5.4.3 Time-Optimal Paths

In this section, we show how the optimal controls of the evaders and the active pursuers, as well as their corresponding optimal trajectories can be retrieved from the computed level sets.

We first consider the active pursuers. Since they can reach X_f at time T, it is clear that X_f resides on the boundary of their reachable sets, otherwise capture would have occurred earlier. Therefore, when the ϕ_P^i's are differentiable, the optimal trajectory for each active pursuer satisfies [85]

$$\frac{\mathrm{d}X_P^{i^*}}{\mathrm{d}t} = \bar{u}^i \frac{\nabla \phi_P^i}{|\nabla \phi_P^i|} + w(X_P^{i^*}, t), \quad X_P^{i^*}(0) = X_{P_0}^i, \quad i \in \mathcal{I}_A, \tag{118}$$

where $\mathcal{I}_A \subseteq \mathcal{I}$ denotes the index set of the active pursuers. The corresponding optimal controls of the active pursuers are thus

$$u_P^{i*} = \bar{u}^i \frac{\nabla \phi_P^i}{|\nabla \phi_P^i|}, \quad i \in \mathcal{I}_A. \tag{119}$$

In order to find the optimal trajectories of the active pursuers, we propagate backward the dynamics (118) starting from X_f until it reaches $X^i_{P_0}, i \in \mathcal{I}_A$.

As for the evader, there are two possible outcomes after the termination of the evolution of the reachable sets of the pursuers and the evader. One possibility is that at the terminal time T, X_f resides on $\partial\tilde{\mathcal{R}}_E(X_{E_0}, T)$ (or $\partial\mathcal{R}_E(X_{E_0}, T)$ when $\bar{v} \leq \min_{i\in\mathcal{I}} \bar{u}^i$). In this case it follows that the boundary of the reachable set of the evader is not fully covered for all $t < T$. When differentiable, the optimal trajectory of the evader is then unique and it satisfies the differential equation

$$\frac{\mathrm{d}X^*_E}{\mathrm{d}t} = \bar{v}\frac{\nabla\phi_E}{|\nabla\phi_E|} + w(X^*_E, t). \tag{120}$$

The corresponding optimal control for the evader is given by

$$u^*_E = \bar{v}\frac{\nabla\phi_E}{|\nabla\phi_E|}. \tag{121}$$

It may also happen that X_f lies in the interior of $\tilde{\mathcal{R}}_E(X_{E_0}, T)$ (or the interior of $\mathcal{R}_E(X_{E_0}, T)$ when $\bar{v} \leq \min_{i\in\mathcal{I}} \bar{u}_i$). This situation occurs when there exists $t_c \in (t_0, T)$ such that $\partial\mathcal{R}_E(X_{E_0}, t) \subset \bigcup_{i=1}^{n} \mathcal{R}^i_P(X^i_{P_0}, t)$, for all $t \in [t_c, T]$. However, some part of the interior of $\mathcal{R}_E(X_{E_0}, t)$ may not be covered until time T. In this case, the optimal control of the evader is not necessarily unique. In particular, the control of the evader can be chosen from the set

$$\mathcal{U}^*_E = \left\{ u_E \in \mathcal{U}_E : X \text{ satisfies (109) and } X(\tau) \notin \bigcup_{i=1}^{n} \mathcal{R}^i_P(X^i_{P_0}, \tau), \forall\tau \in [t_0, T] \right\}. \tag{122}$$

It follows that an optimal control for the evader is valid, as long as it can bring the evader to X_f at time T, without getting captured by any of the pursuers prior to time T.

5.5 Simulation Results

We present in the section simulation results for the multiplayer pursuit-evasion problem under two distinct flow fields.

We first consider a state-dependent wind field that resembles the shape of hurricanes whose wind field snapshot can be found in [108]. Let $X = [x, y, z]^\intercal \in \mathbb{R}^3$, this much simplified approximation model is generated:

$$w(X) = \sqrt{\frac{z}{h}} A(X - X_s), \quad X \in \mathcal{D} \tag{123}$$

where

$$A = \begin{bmatrix} 0.2 & 0.3 & 0 \\ -0.15 & 0.1 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{124}$$

In (123), $X_s = [55, 40, 0]$ denotes the location of the flow singularity and $\mathcal{D} = [0, 128]^3$ represents the 3D space. Also in (123), $h = 128$ denotes the height of the 3D space, and A_i is a 3×3 matrix, whose structure captures the local characteristics of the flow singularity. Notice that due to the designated value of A, the flow field along z axis is zero. Also, for each fixed z, the flow field in the xy-plane approximates the velocity field of a vortex with a linear vector field from the Rankine model [29]. The multiplier $\sqrt{z/h}$ scales the magnitude of the flow field along the z axis so that the flow intensifies as the height increases. The vector field of the external flow field is depicted in Figure 34.

In the first example, we formulate a five-pursuers/two-evaders problem. At time $t_0 = 0$, the three pursuers are located at $X^1_{P_0} = [60, 20, 30]^\intercal$, $X^2_{P_0} = [70, 60, 50]^\intercal$, $X^3_{P_0} = [90, 80, 70]^\intercal$, $X^4_{P_0} = [50, 80, 60]^\intercal$ and $X^5_{P_0} = [25, 70, 45]^\intercal$, respectively. Their corresponding maximum speeds are given by $\bar{u}_1 = 30$, $\bar{u}_2 = 30$, $\bar{u}_3 = 40$, $\bar{u}_4 = 50$

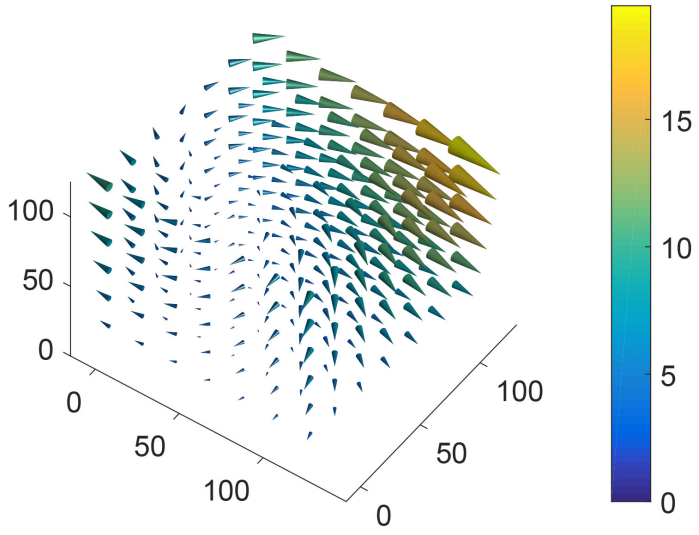


Figure 34: An analytical flow field that approximates a hurricane.

and $\bar{u}_5 = 40$. The initial locations of the evaders are given by $X^1_{E_0} = [50, 50, 40]^\intercal$, $X^2_{E_0} = [70, 90, 75]^\intercal$ and their maximum speed are $\bar{v}_1 = 20$, $\bar{v}_2 = 15$, respectively. Note that, in this example, the maximum speed of the evaders is smaller than the speed of the pursuers. Therefore, we only need to propagate the reachability front of the evaders and pursuers in order to recover $\mathcal{R}^\star_E(X_{E_0}, t)$.

We start the simulation by propagating the reachability front of each pursuer until it coincides with one of the evaders to decide which evader the pursuer is assigned to. In this example, P_1, P_2 and P_5 are assigned to evader E_1 and P_3, P_4 are paired with E_2.

Then for each subgame, we utilize the reachable set algorithm to find the optimal terminal time for the evader to be captured by the pursuers. The optimal time-to-capture for the first subgame between P_1, P_2, P_5 and E_1 is $T_1 = 1.20$ and the optimal time-to-capture of the second subgame is $T_2 = 0.76$. So the overall time-to-capture is $T_o = 1.20$. The reachable fronts of the pursuers and the evaders at their corresponding capture time, as well as the optimal trajectories of the active pursuers and the evaders are shown in Figure 35. The red, yellow and purple color surfaces

represent the reachable fronts of the pursuers P_1, P_2 and P_5 at the terminal time T_1, respectively. The union of the reachable sets of these pursuers fully cover the reachable set of the evader E_1 at time T_1 denoted by the blue surface and the evader is captured at X_{T_1}. The red dashed and blue lines represent the optimal trajectories of the pursuers and evaders respectively. Black arrows on the background represent the external flow field.

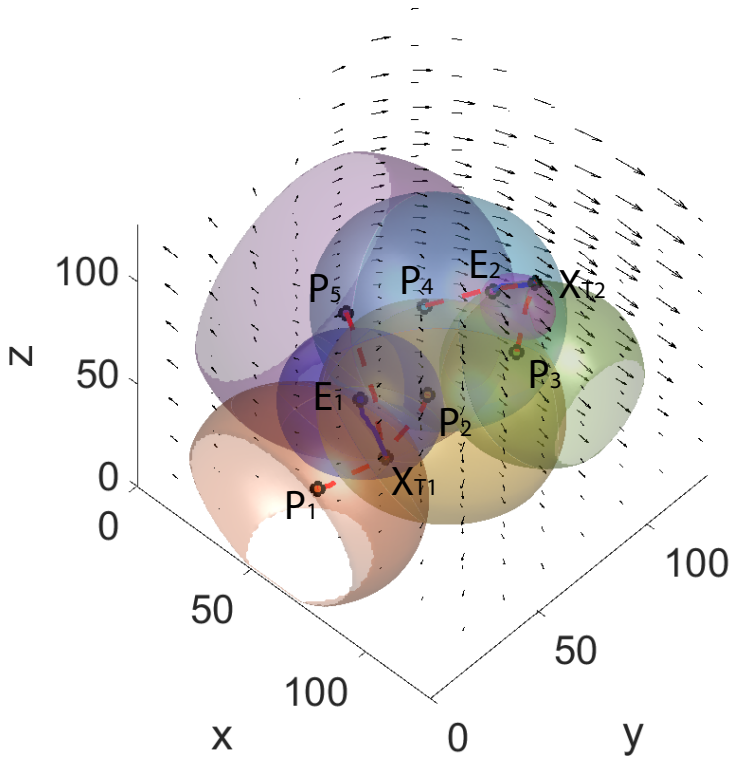


Figure 35: Pursuit-evasion between 5 pursuers and 2 evader in an analytical flow field.

We then apply our algorithm to a pursuit-evasion problem with respect to the Matlab default wind dataset. The external flow field generated from this dataset is depicted in Figure 36. Through this example, we demonstrate that the proposed algorithm can handle scenarios with complex spatial flow fields.

The initial positions of the pursuers and the evaders are set as $X^1_{P_0} = [45, 20, 30]^\intercal$, $X^2_{P_0} = [55, 45, 45]^\intercal$, $X^3_{P_0} = [50, 70, 35]^\intercal$, $X^4_{P_0} = [45, 95, 40]^\intercal$ and $X^1_{E_0} = [70, 40, 40]^\intercal$,

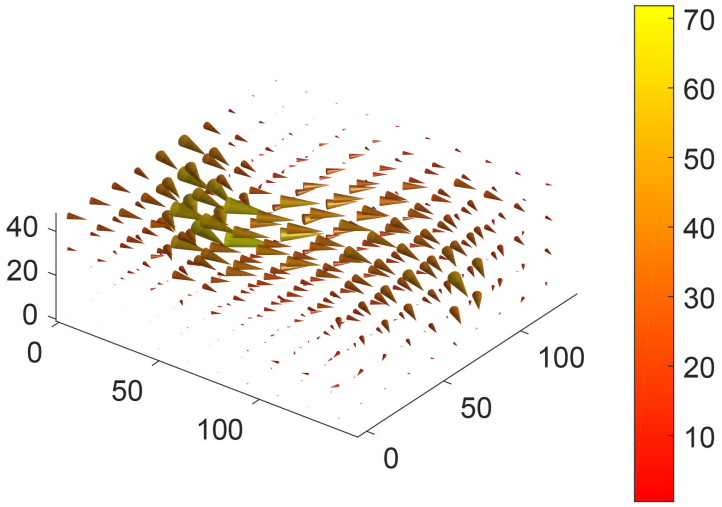


Figure 36: A realistic flow field from Matlab.

$X^2_{E_0} = [60, 90, 30]^\intercal$, respectively. Their corresponding maximum speeds are given by $\bar{u}_1 = 40$, $\bar{u}_2 = 30$, $\bar{u}_3 = 30$, $\bar{u}_4 = 35$ and $\bar{v}_1 = 15$, $\bar{v}_2 = 20$. After propagation of reachable sets of the pursuers, it is determined that P_3, P_4 will go after E_1 while E_2 will be actively chased by P_1 and P_2. Capture of E_1 occurs at time $T_1 = 1.00$ and E_2 is captured at time $T_2 = 0.98$. the optimal paths of the pursuers and the evaders are depicted in Figure 37. The reachability fronts of the pursuers P_1 through P_4 are illustrated in purple, green, red and yellow colors. Similarly, the reachability fronts of the evaders E_1, E_2 are depicted by blue and magenta surfaces. The optimal trajectories of the pursuers and evaders are shown as red dashed and blue lines, respectively.

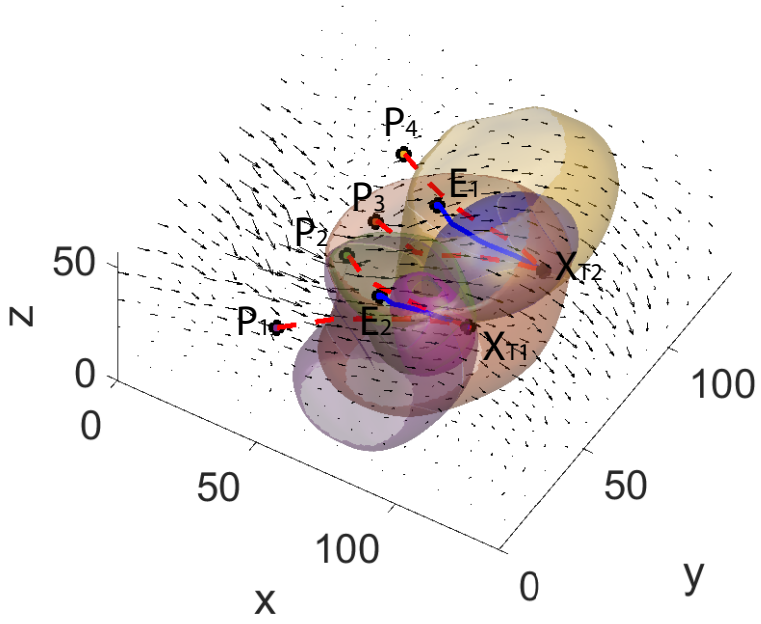


Figure 37: Pursuit-evasion between 4 pursuers and 2 evader in a realistic wind field.

CHAPTER VI

PURSUIT-EVASION GAMES UNDER STOCHASTIC FLOW FIELDS

6.1 Problem Formulation

Consider a pursuit-evasion game in an external stochastic dynamic flow field with a single pursuer P and a single evader E. Henceforth, we will refer to the pursuer and the evader collectively as "agents". The dynamics of the pursuer P is given by

$$\mathrm{d}X_P(t) = u_P(t)\mathrm{d}t + W(X_P(t), \mathrm{d}t, \mathrm{d}w_P), \quad X_P(0) = X_{P_0}, \tag{125}$$

where $X_P(t) = [x_P(t), y_P(t)]^\intercal \in \mathbb{R}^2$ denotes the position of the pursuer, $u_P(t) \in \mathbb{R}^2$ is the control input of the pursuer that satisfies the piecewise constraint $u_P(t) \in U_P$, where $U_P = \{u \in \mathbb{R}^2, |u| \leqslant \bar{u}\}$, and $|\cdot|$ represents the 2-norm. In (125), $W(X_P(t), \mathrm{d}t, \mathrm{d}w_P)$ represents the instantaneous dynamic flow. It is further assumed that the instantaneous flow field takes the form

$$W(X, \mathrm{d}t, \mathrm{d}w) = F(X)\mathrm{d}t + G(X)\mathrm{d}w, \quad X \in \mathbb{R}^2, \tag{126}$$

where $w = [w_1, w_2]^\intercal$ and w_1, w_2 denote two independent standard Wiener processes. $F(X) : \mathbb{R}^2 \to \mathbb{R}^2$ and $G(X) : \mathbb{R}^2 \to \mathbb{R}^{2\times 2}$ are two state dependent functions.

The goal of the pursuer is to capture an evader, whose kinematics is given by

$$\mathrm{d}X_E(t) = u_E(t)\mathrm{d}t + W(X_E(t), \mathrm{d}t, \mathrm{d}w_E), \quad X_E(0) = X_{E_0}, \tag{127}$$

where $X_E(t) = [x_E(t), y_E(t)]^\intercal \in \mathbb{R}^2$ is the position of the evader, and $u_E(t)$ is its control input such that $u_E(t) \in U_E$, where $U_E = \{v \in \mathbb{R}^2, |v| \leqslant \bar{v}\}$. It is assumed that $\bar{v} < \bar{u}$ for simplicity of the analysis. Also note that w_P in (125) and w_E in (127) are independent Wiener processes in the form of w.

Let $\mathcal{X} = [X_E^\intercal, X_P^\intercal]^\intercal \in \mathbb{R}^4$ denote the state of the system. The game begins at initial time $t_0 = 0$ with initial positions $\mathcal{X}_0 = [X_{E_0}^\intercal, X_{P_0}^\intercal]^\intercal$, and terminates when there exists some time T such that $\|X_P(T) - X_E(T)\| \leq \epsilon$ for some $\epsilon > 0$, where $\|\cdot\|$ denotes the Euclidean norm. Here T is called the terminal time.

Let $J(\gamma_P, \gamma_E) = \mathbb{E}\left[\int_0^T 1 \mathrm{d}t\right]$ be the cost function of the game, where $\gamma_P, \gamma_E : \mathbb{R}^+ \times \mathbb{R}^4 \mapsto \mathbb{R}^2$ denote the feedback strategies of the pursuer and the evader, respectively, such that $\gamma_P(t, \mathcal{X}) = u_P(t)$ and $\gamma_E(t, \mathcal{X}) = u_E(t)$. We assume that each player has perfect knowledge of the dynamics of the system represented by (125) and (127), the constraint sets U_P and U_E, the cost function J, as well as the initial state X_0. It is also assumed that the value V of the game [62] exists, that is,

$$V = \min_{\gamma_P} \max_{\gamma_E} J = \max_{\gamma_E} \min_{\gamma_P} J. \tag{128}$$

The objective of this chapter is to characterize the capture condition of the pursuit-evasion game through a reachability-based approach. Henceforth, we consider the open-loop representation of the optimal strategies of the pursuer and the evader. In particular, we consider the control of the pursuer $u_P \in \mathcal{U}_P$, where $\mathcal{U}_P$ consists of all piecewise continuous functions, whose range is included in U_P, and call u_P an *admissible* control of the pursuer. Similarly, the control u_E is an *admissible* control of the evader if $u_E \in \mathcal{U}_E$, which consists of all piecewise continuous functions whose range is included in U_E.

6.2 Problem Analysis

6.2.1 Moment Expansion

Consider the pursuer dynamics given by (125). We can divide the variable X_P to a deterministic part and a random fluctuation [77]. In particular, let

$$X_P = \bar{X}_P + \xi_P, \tag{129}$$

where $\bar{X}_P := \mathbb{E}[X_P]$ denotes the mean of X_P and ξ_P is a random variable with zero mean. The corresponding equations of the time-evolution of X_P and linear approximation of ξ_P are given by

$$\mathrm{d}\bar{X}_P(t) = (F(\bar{X}_P(t)) + u_P)\mathrm{d}t, \quad \bar{X}_P(0) = X_{P_0}, \tag{130}$$

$$\mathrm{d}\xi_P(t) = D(\bar{X}_P(t))\xi_P(t)\mathrm{d}t + G(\bar{X}_P(t))\mathrm{d}w, \quad \xi_P(0) = \mathbf{0}^{2\times 1}, \tag{131}$$

where $D(\bar{X}_P(t)) = \frac{\partial F}{\partial X}(\bar{X}_P(t))$ represents the Jacobian of F. The decomposition of the state variable makes it easy to describe how the mean and variance of the probability distribution of X_P evolves. The evolution of the mean of X_P is readily given by (130) and the change of the covariance matrix Σ_P can be calculated as

$$\frac{\mathrm{d}\Sigma_P}{\mathrm{d}t} = D\Sigma_P + \Sigma_P D^\intercal + GG^\intercal, \quad \Sigma_P(0) = \mathbf{0}^{2\times 2}, \tag{132}$$

where the arguments of the functions Σ_P, D and G are omitted for brevity.

It is worth mentioning that $\xi_P(t)$ follows a Gaussian distribution due to the fact that the linear dynamics (125) preserve Gaussian distributions [68].

We can apply a similar analysis to the evader, and let

$$X_E = \bar{X}_E + \xi_E. \tag{133}$$

Then the ordinary differential equations that govern the mean $\bar{X}_E$ and covariance matrix Σ_E of ξ_E can be obtained as

$$\frac{\mathrm{d}\bar{X}_E(t)}{\mathrm{d}t} = F(\bar{X}_E(t)) + u_E, \quad \bar{X}_E(0) = X_{E_0}, \tag{134}$$

$$\frac{\mathrm{d}\Sigma_E}{\mathrm{d}t} = D\Sigma_E + \Sigma_E D^\intercal + GG^\intercal, \quad \Sigma_E(0) = \mathbf{0}^{2\times 2}. \tag{135}$$

6.2.2 Reachable Sets

In the deterministic setting, a reachable set at a given time is defined as the set of points that can be visited by the agent at a particular time [136]. In particular, the reachable set of the pursuer at time $t \geq 0$, denoted by $\mathcal{R}_P(X_{P_0}, t)$, is the set of all points $X \in \mathbb{R}^2$ such that there exists a trajectory satisfying (125), with initial position X_{P_0} and terminal position X at time t. Similarly, the reachable set $\mathcal{R}_E(X_{E_0}, t)$ of the evader at time $t \geq 0$ is the set of all points $X \in \mathbb{R}^2$ such that there exists a trajectory satisfying (127), with initial position X_{E_0} and terminal position X at time t.

However, in the stochastic setting, that is, when the dynamics of the agent is driven by stochastic differential equations, then this agent has a non-zero probability to reach any point within the domain of interest ($\mathbb{R}^2$ in our case) due to the fact that $w \sim \mathcal{N}(0, I)$ and the support of the normal distribution $\mathcal{N}(0, I)$ is $\mathbb{R}^2$. Therefore, the reachable set defined in the deterministic setting may not be a good indicator for our problem.

Instead, we follow the decomposition approach presented in the previous subsection and describe the reachable set of each agent as the mean reachable set and its random fluctuation. The mean reachable sets of each agent for our problem are defined as the deterministic reachable sets of the pursuer and the evader whose dynamics are given by (130) and (134), respectively. The variance of the reachable sets are determined by (132) and (135), respectively.

Here we give some formal definitions about reachable sets that will be useful in

the follow-up analysis.

Definition 6.1 *(Ref. [136]) The* mean reachable set *of the pursuer at time $t \geq t_0$, denoted by $\bar{\mathcal{R}}_P(X_{P_0}, t)$, is the set of all points $X \in \mathbb{R}^2$, such that there exists a trajectory satisfying (130) for all $\tau \in [t_0, t]$ with $X_P(t_0) = X_{P_0}$ and $X_P(t) = X$.*

In other words, the mean reachable set of the pursuer at time $t \geq t_0$ is the set of all the points that can be reached by the pursuer at time t under the dynamics (130). Similarly, the mean reachable set $\bar{\mathcal{R}}_E(X_{E_0}, t)$ of the evader at time $t \geq t_0$ is the set of all the points that can be reached by the evader at time t under the dynamics (134).

Definition 6.2 *(Ref. [87]) The boundary of the mean reachable set is the* mean reachability front*.*

The mean reachability fronts of the pursuer and the evader at time $t \geq t_0$ are denoted by $\partial\bar{\mathcal{R}}_P(X_{P_0}, t)$ and $\bar{\mathcal{R}}_E(X_{E_0}, t)$, respectively.

Definition 6.3 *The* augmented reachable set *of the pursuer at time $t \geq t_0$ is defined as*

$$\tilde{\mathcal{R}}_P(X_{P_0}, t) = \bigcup_{X \in \bar{\mathcal{R}}_P(X_{P_0}, t)} \mathcal{O}_P^X, \tag{136}$$

where $\mathcal{O}_P^X = \{Y \in \mathbb{R}^2 : g_P(Y) \leq \alpha\}$, and

$$g_P(Y) = \left(\frac{(Y(1) - X(1))\cos(A_P) + (Y(2) - X(2))\sin(A_P)}{\sigma_{P1}} \right)^2 + \left(\frac{(Y(1) - X(1))\sin(A_P) + (Y(2) - X(2))\cos(A_P)}{\sigma_{P2}} \right)^2.$$

Here $\sigma_{P1} = \sqrt{\lambda_{P1}}$ and $\sigma_{P2} = \sqrt{\lambda_{P2}}$, where λ_{P1} and λ_{P2} represent the two eigenvalues of the covariance matrix $\Sigma_P(t)$, and $A_P = \arctan(v_P(2)/v_P(1))$, where v_P is the eigenvector of $\Sigma_P(t)$ that corresponds to the largest eigenvalue. Also, α is a design parameter that can be varied to reflect different levels of confidence intervals.

Similarly, the augmented reachable set of the evader at time $t \geq t_0$ is defined as

$$\tilde{\mathcal{R}}_{E}(X_{E_0}, t) = \bigcup_{X \in \bar{\mathcal{R}}_{E}(X_{E_0}, t)} \mathcal{O}_{E}^{X}, \tag{137}$$

where $\mathcal{O}_{E}^{X} = \{Y \in \mathbb{R}^2 : g_{E}(Y) \leq \alpha\}$, and

$$\begin{aligned} g_{E}(Y) = &\left(\frac{(Y(1) - X(1))\cos(A_{E}) + (Y(2) - X(2))\sin(A_{E})}{\sigma_{E1}} \right)^2 \\ &+ \left(\frac{(Y(1) - X(1))\sin(A_{E}) + (Y(2) - X(2))\cos(A_{E})}{\sigma_{E2}} \right)^2. \end{aligned}$$

Here $\sigma_{E1} = \sqrt{\lambda_{E1}}$ and $\sigma_{E2} = \sqrt{\lambda_{E2}}$, where λ_{E1} and λ_{E2} represent the two eigenvalues of the covariance matrix $\Sigma_{E}(t)$. $A_{E} = \arctan(v_{E}(2)/v_{E}(1))$, where v_{E} is the eigenvector of $\Sigma_{E}(t)$ that corresponds to the largest eigenvalue.

Notice that $\mathcal{O}_{P}^{X}$ in Definition 6.3 is also known as a error ellipse (or confidence ellipse) [58, 135] for a 2-dimensional Gaussian distribution. The error ellipse represents an iso-contour of the Gaussian distribution, and allows one to visualize a 2-dimensional confidence interval. The problem then is how to find α, such that the scale of the resulting ellipse corresponds to a prespecified confidence level. This relationship can be found through the cumulative distribution function of the chi-squared distribution [8]. Each value of α corresponds to a confidence ellipse with probability β. Their relation can be found by checking the chi-squared distribution table [155]. For instance, the value $\alpha = 5.991$ corresponds to a 95% confidence ellipse. That is, pick any $X \in \bar{\mathcal{R}}_{P}(X_{P_0}, t)$, there exists a control $u_{P} \in \mathcal{U}_{P}$ such that X can be reached by the pursuer at time t with dynamics (130) and control u_{P}. Then the terminal position at time t under dynamics (125) and control $\bar{u}_{P}$ will reside in the region $\mathcal{O}_{P}^{X}$ centered at X with 95% probability. Henceforth, we fix the value of α to be 5.991 in all the examples.

The previous discussion is demonstrated in Figure 38. In this example the agents

are subject to the flow field

$$W(X, \mathrm{d}t, \mathrm{d}w) = A(X - X_s)\mathrm{d}t + B\mathrm{d}w, \tag{138}$$

where

$$A = \begin{bmatrix} 0.2 & 0.3 \\ -0.15 & 0.1 \end{bmatrix}, \quad X_s = \begin{bmatrix} 15 \\ 15 \end{bmatrix}, \quad B = \begin{bmatrix} 0.25 & 0 \\ 0 & 0.25 \end{bmatrix}.$$

The vector field in the background depicts the deterministic part of the flow field. The green curve represents the mean reachability front of the pursuer at time $t = 2$, and the gray region corresponds to the augmented reachable set of the pursuer. The magenta ellipse and its interior denotes the 95% error ellipse with respect to X_1.

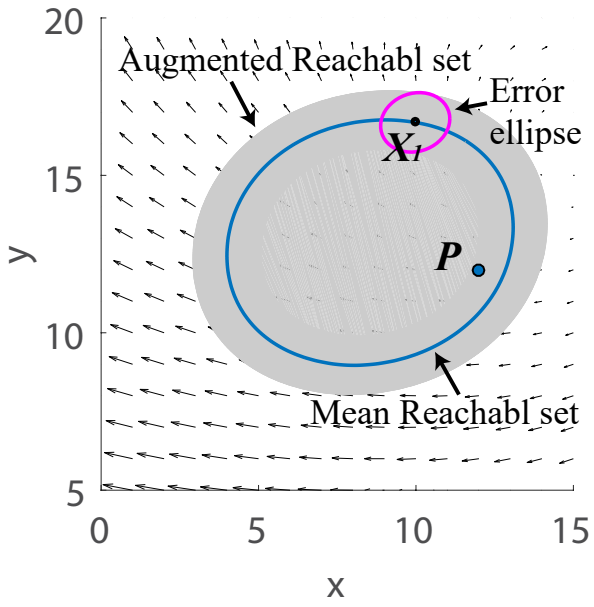


Figure 38: Mean and augmented reachable sets of the pursuer at $t = 2$. A 95% confidence ellipse centered at X_1 is also shown.

Figure 39 shows a Monte Carlo simulation of 100 trajectories with dynamics (125) and the flow field specified by (138). The initial position is at $X_0 = [12, 12]^\intercal$ and the nominal control is given by $u_P = [2\cos(\pi/4), 2\sin(\pi/4)]^\intercal$. A 95% confidence ellipse centered at the terminal position with respect to the mean dynamics (130) is also

illustrated as the gray shaded region. As can be seen in the figure, more than 95% of all the terminal position samples fall in this confidence ellipse, which is consistent with our previous analysis.

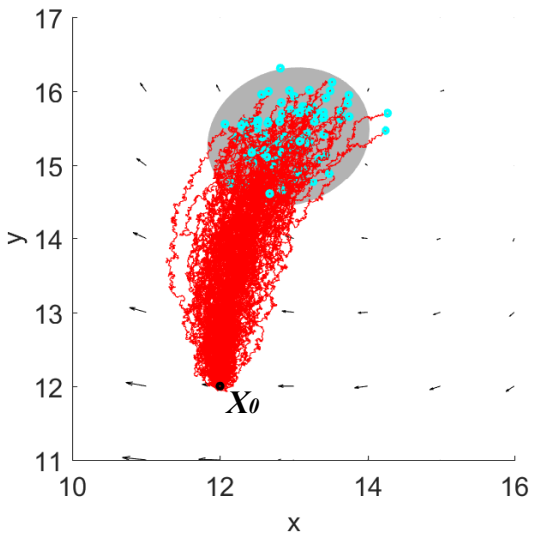


Figure 39: A 95% confidence ellipse in gray that contains most of the terminal positions (cyan points) of the Monte Carlo simulation.

Suppose that at some finite time t_c, the condition $\tilde{\mathcal{R}}_E(X_{E_0}, t_c) \subseteq \tilde{\mathcal{R}}_P(X_{P_0}, t_c)$ holds. Then for any admissible control of the evader $u_E \in \mathcal{U}_E$ in the time interval $[0, t_c]$, the terminal position $X_E(t_c) \in \tilde{\mathcal{R}}_E(X_{E_0}, t_c) \subseteq \tilde{\mathcal{R}}_P(X_{P_0}, t_c)$ with at least 95% probability. This implies that there is a high probability for the existence of an admissible control $u_P \in \mathcal{U}_P$ of the pursuer such that $\|X_P(t_c) - X_E(t_c)\| \leq \epsilon$.

On the other hand, if there exists some finite time t_e, such that $\tilde{\mathcal{R}}_E(X_{E_0}, t_e) \nsubseteq \tilde{\mathcal{R}}_P(X_{P_0}, t_e)$, the evader can find a control $u_E \in \mathcal{U}_E$ to reach the vicinity of $\tilde{\mathcal{R}}_E(X_{E_0}, t_e)$ excluding $\tilde{\mathcal{R}}_P(X_{P_0}, t_e)$ and avoid capture in the time interval $[0, t_e]$ with a high probability, no matter how the pursuer chooses his own admissible control.

These observations lead to the following proposition.

Proposition 6.1 *Let*

$$T = \inf\{t \in \mathbb{R} : \tilde{\mathcal{R}}_E(X_{E_0}, t) \backslash \tilde{\mathcal{R}}_P(X_{P_0}, t) = \emptyset\}.$$

Then T is the expected optimal time-to-capture with probability β^2.

Proof. We start the proof by introducing two events. Let $\mathcal{A} = \{X_{E}(T) \in \tilde{\mathcal{R}}_{E}(X_{E_0}, T)\}$ and $\mathcal{B} = \{X_{P}(T) \in \tilde{\mathcal{R}}_{P}(X_{P_0}, T)\}$. Since $\tilde{\mathcal{R}}_{P}(X_{P_0}, T) = \bigcup_{X \in \bar{\mathcal{R}}_{P}(X_{P_0}, T)} \mathcal{O}_{P}^{X}$ by Definition 6.3, and for each $X \in \bar{\mathcal{R}}_{P}(X_{P_0}, T)$ and its corresponding admissible control $\bar{u}_{P}$ in $[0, T]$, there is a β probability for the terminal position at time t under dynamics (125) and control $\bar{u}_{P}$ to be in the region $\mathcal{O}_{P}^{X}$ centered at X. Thus, it can be deduced that $P(\mathcal{A}) = 95\%$. Similarly, $P(\mathcal{B}) = \beta$. Furthermore, since the construction of $\tilde{\mathcal{R}}_{E}(X_{E_0}, T)$ is independent from the evader and vise versa, it follows that the events $\mathcal{A}$ and $\mathcal{B}$ are independent. Therefore, $P(\mathcal{A} \cap \mathcal{B}) = P(\mathcal{A})P(\mathcal{B}) = \beta^2$.

Now that we have found the probability for events $\mathcal{A}$ and $\mathcal{B}$ to happen at the same time, we can restrict our analysis to the case where $\{X_{E}(T) \in \tilde{\mathcal{R}}_{E}(X_{E_0}, T)\}$ and $\{X_{P}(T) \in \tilde{\mathcal{R}}_{P}(X_{P_0}, T)\}$. Since $\tilde{\mathcal{R}}_{E}(X_{E_0}, t) \backslash \tilde{\mathcal{R}}_{P}(X_{P_0}, t) = \emptyset$, it follows that for any point $X \in \tilde{\mathcal{R}}_{E}(X_{E_0}, T)$ that can be visited by the evader at time T through an admissible evading control $u_{E} \in \mathcal{U}_{E}$, it is also true that $X \in \tilde{\mathcal{R}}_{P}(X_{P_0}, T)$. In other words, there exists an admissible control of the pursuer $u_{P} \in \mathcal{U}_{P}$ such that $\|X_{P}(T) - X_{E}(T)\| = \epsilon$. Therefore, regardless of the strategy it picks, the evader can be captured by the pursuer at some time $t \leq T$.

On the other hand, since $t = T$ is the first time such that $\tilde{\mathcal{R}}_{E}(X_{E_0}, t) \backslash \tilde{\mathcal{R}}_{P}(X_{P_0}, t) = \emptyset$ is satisfied, it follows that $\tilde{\mathcal{R}}_{E}(X_{E_0}, t) \backslash \tilde{\mathcal{R}}_{P}(X_{P_0}, t) \neq \emptyset$ for all $0 \leq t < T$. Hence, for all $t \in [0, T)$, there exists $X_t \in \tilde{\mathcal{R}}_{E}(X_{E_0}, t)$ such that $X_t \notin \tilde{\mathcal{R}}_{P}(X_{P_0}, t)$. In other words, for any time $t \in [0, T)$, there exist an admissible control for the evader to reach X_t such that X_t cannot be visited by the pursuer at time t. It follows that the evader can be expected to avoid capture before time T.

From the previous statements, we can conclude that T is the expected optimal time-to-capture with β^2 probability.

□

6.3 Numerical Solution

6.3.1 Level Set Method

In order to construct the mean reachable sets of the pursuers and the evader and their corresponding augment reachable sets, we utilize the level set method.

The mean reachability front $\partial\bar{\mathcal{R}}_P(X_{P_0}, t)$ of the pursuer is governed by the viscosity solution of the Hamilton-Jacobi (HJ) equation

$$\frac{\partial\phi_P(X,t)}{\partial t} + \bar{u}\,|\nabla\phi_P| + w(X,t)\nabla\phi_P = 0, \tag{139}$$

with initial condition $\phi_P(X,0) = |X - X_{P_0}|$. Moreover, the mean reachable set of the pursuer coincides with the region(s) where ϕ_P is non-positive. Similarly, the mean reachability front $\partial\bar{\mathcal{R}}_E(X_{E_0}, t)$ of the evader is given by the HJ equation

$$\frac{\partial\phi_E(X,t)}{\partial t} + \bar{v}\,|\nabla\phi_E| + w(X,t)\nabla\phi_E = 0, \tag{140}$$

with initial conditions $\phi_E(X,0) = |X - X_{E_0}|$.

After the mean reachable set of the pursuer at some time t is obtained through level set propagation, the augmented reachable set of the pursuer at time t can be subsequently achieved by substituting each mesh grid point X inside the mean reachable set with the error ellipse $\mathcal{O}_P^X$ defined in (136). The covariance matrix $\Sigma_P(t)$ that is required to generated the error ellipse can be calculated from forward integration of equation (132). The aforementioned method to generate the augmented reachable set can be simplified as follows. Instead of extending all mesh grid points inside the mean reachable set with their corresponding error ellipses, we can obtain the augmented reachable set by substituting only the grid points in the mean reachability front with their error ellipse counterparts. The reason this approach works is that the error ellipse with respect to the points in the interior of the mean reachable set will still be

in the interior of the resulting augmented reachable set. The augmented reachable set of the evader at time t can be generated in a similar manner. Afterwards, the expected optimal time-to-capture can be found as the first time such that the augmented reachable set of the evader is fully covered by the augmented reachable set of the pursuer, as described in Proposition 6.1.

6.4 Simulation Results

In this section, we present simulation results of the two-player pursuit-evasion problem under an external stochastic flow field. The external wind field is approximated by

$$W(X, \mathrm{d}t, \mathrm{d}w) = A(X - X_s)\mathrm{d}t + B\mathrm{d}w,$$

where $A \in \mathbb{R}^{2\times 2}$ and $X_s \in \mathbb{R}^2, \sigma \in \mathbb{R}$ are constant values. The deterministic part of this wind field can be seen as a flow generated from a single singularity point located at X_s, with its characteristics captured by A. We set

$$A = \begin{bmatrix} 0.2 & 0.3 \\ -0.15 & 0.1 \end{bmatrix}, \quad X_s = \begin{bmatrix} 15 \\ 15 \end{bmatrix}, \quad B = \begin{bmatrix} 0.25 & 0 \\ 0 & 0.25 \end{bmatrix}.$$

The initial conditions of the pursuer and the evader are given by $X_{P_0} = [12, 12]^\intercal$ and $X_{E_0} = [14, 14]^\intercal$. The maximum speeds of the pursuer and the evader are set to $\bar{u} = 3$ and $\bar{v} = 1$, respectively. At time $t = 1.88$, the augmented reachable set of the evader, depicted as the dark gray region, is fully contained in the augmented reachable set of the pursuer, depicted as the light gray region. The result generated by the method in this paper is shown in Figure 40. Therefore, the expected optimal time-to-capture $T = 1.88$ with a 90.25% probability.

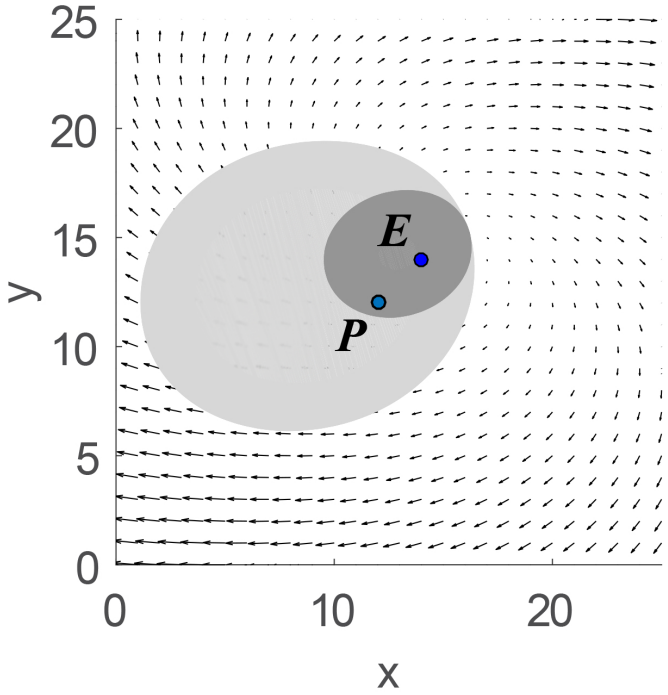


Figure 40: Augmented reachable sets with 95% confidence level of the pursuer and the evader at $t = 1.88$ in light and dark gray, respectively.

CHAPTER VII

GAME THEORETIC CONTINUOUS TIME DIFFERENTIAL DYNAMIC PROGRAMMING

7.1 Introduction

Differential game-theoretic or min-max formulations are important extensions of optimal control having direct connections to robust and H^∞ nonlinear control theory. Despite the plethora of work in this area, min-max algorithms for trajectory optimization have only very recently been derived, and have been applied to humanoid robotic control problems [105], [104]. In addition, although the initial derivation of DDP [63] is in continuous time, most of work on trajectory optimization, including the min-max DDP formulation in [105], [104], relies on either discrete time nonlinear systems or discretized versions of systems that are initially expressed in continuous time.

Given all this existing work in the area of trajectory optimization based on DDP, our contribution in this chapter is the derivation of Game-Theoretic DDP (GT-DDP) in continuous time. We provide a set of backward differential equations that are easy to implement and derive the optimal policies for the two players/controllers. Furthermore, we investigate the effect that the min-max formulation has in the feedforward and feedback parts of the optimal control policies.

With respect to the initial treatment of DDP in the book by D. H. Jacobson and D. Q. Mayne [63] our analysis and derivation of GT-DDP avoids a restrictive assumption of the initial derivation. This assumption was also discussed in a review paper of [63] published in 1971 by Michael K. Sain [32]. In particular, the fundamental assumption in the derivation of continuous-time DDP in [63] is that the nominal control $\bar{\mathbf{u}}$ is close

to the optimal control $\mathbf{u}^*$. This assumption allows the expansion of the terms in the Hamilton-Jacobi-Bellman(HJB) Partial Differential Equation (PDE) around $\mathbf{u}^*$ instead $\bar{\mathbf{u}}$ and results in the cancelation of terms that depend on $\mathcal{H}_{\mathbf{u}^*} = 0$, where $\mathcal{H}_{\mathbf{u}^*}$ stands for the partial derivative of the Hamiltonian with respect to the control input.

GT-DDP does not rely on the assumption regarding the closeness of the nominal controls $\bar{\mathbf{u}}$ and $\bar{\mathbf{v}}$ to $\mathbf{u}^*$ and $\mathbf{v}^*$, respectively, and therefore the quadratic expansions of the terms in the HJB PDE are computed around the nominal controls $\bar{\mathbf{u}}$, $\bar{\mathbf{v}}$ and not the optimal control $\mathbf{u}^*$, $\mathbf{v}^*$. In this case, the term $\mathcal{H}_{\mathbf{u}}$ is not necessarily equal to zero.

7.2 Problem Formulation

We consider the following min-max problem:

$$V(\mathbf{x}(t_0), t_0) = \min_{\mathbf{u}} \max_{\mathbf{v}} \left\{ \phi(\mathbf{x}(t_f), t_f) + \int_{t_0}^{t_f} \mathcal{L}(\mathbf{x}, \mathbf{u}, \mathbf{v}, t) \mathrm{d}t \right\}, \tag{141}$$

subject to the dynamics

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{F}(\mathbf{x}, \mathbf{u}, \mathbf{v}, t), \quad \mathbf{x}(t_0) = \mathbf{x}_0, \tag{142}$$

where V stands for the optimal performance index starting from $\mathbf{x}_0$ at time t_0, $\mathbf{x}(t)$ is an n-dimensional vector function of time describing the state of the dynamic system at $t \in [0, t_f]$. $\mathcal{L}$ and ϕ are scalar functions of their arguments, where $\mathcal{L}(\mathbf{x}, \mathbf{u}, \mathbf{v}, t)$ is the *running cost* and $\phi(\mathbf{x}(t_f), t_f)$ is the *terminal cost*. Finally, $\mathbf{u}$ is an m-dimensional vector function that represents the stabilizing control of the system, whose objective is to minimize the performance index, whereas $\mathbf{v}$ is a q-dimensional vector function representing the destabilizing control of the system that tries to maximize the performance index.

In continuous time, the analysis starts with the Hamilton-Jacobi-Bellman Isaacs

(HJBI) partial differential equation. More precisely, we have:

$$-\frac{\partial V(\mathbf{x},t)}{\partial t} = \min_{\mathbf{u}} \max_{\mathbf{v}} \left\{ \mathcal{L}(\mathbf{x},\mathbf{u},\mathbf{v},t) + V_{\mathbf{x}}(\mathbf{x},t)^{\mathsf{T}} F(\mathbf{x},\mathbf{u},\mathbf{v},t) \right\}, \tag{143}$$

under the boundary condition

$$V(\mathbf{x},t_f) = \phi(\mathbf{x}(t_f),t_f). \tag{144}$$

Given an initial/nominal trajectory of the state and control $(\bar{\mathbf{x}},\bar{\mathbf{u}},\bar{\mathbf{v}})$, and letting $\delta\mathbf{x} = \mathbf{x} - \bar{\mathbf{x}}$, $\delta\mathbf{u} = \mathbf{u} - \bar{\mathbf{u}}$, $\delta\mathbf{v} = \mathbf{v} - \bar{\mathbf{v}}$, the linearized dynamics can be represented as

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = F(\bar{\mathbf{x}} + \delta\mathbf{x}, \bar{\mathbf{u}} + \delta\mathbf{u}, \bar{\mathbf{v}} + \delta\mathbf{v}, t), \tag{145}$$

$$\frac{\mathrm{d}\delta\mathbf{x}}{\mathrm{d}t} = F_{\mathbf{x}}(\bar{\mathbf{x}},\bar{\mathbf{u}},\bar{\mathbf{v}},t)\delta\mathbf{x} + F_{\mathbf{u}}(\bar{\mathbf{x}},\bar{\mathbf{u}},\bar{\mathbf{v}},t)\delta\mathbf{u} + F_{\mathbf{v}}(\bar{\mathbf{x}},\bar{\mathbf{u}},\bar{\mathbf{v}},t)\delta\mathbf{v}. \tag{146}$$

The main idea here is to take expansions of the terms in both sides of the equation (143) around the nominal state and control trajectories $(\bar{\mathbf{x}},\bar{\mathbf{u}},\bar{\mathbf{v}})$ to derive the update law for the stabilizing control, destabilizing control and backward differential equations for the zeroth, first and second order approximation terms of the value function. Starting with the left-hand side of (143) we have:

$$\frac{\partial V(\mathbf{x},t)}{\partial t} = \frac{\partial V(\bar{\mathbf{x}},t)}{\partial t} + \frac{\partial V_{\mathbf{x}}^{\mathsf{T}}}{\partial t}\delta\mathbf{x} + \frac{1}{2}\delta\mathbf{x}^{\mathsf{T}}\frac{\partial V_{\mathbf{xx}}}{\partial t}\delta\mathbf{x}. \tag{147}$$

We also have

$$\frac{\mathrm{d}V(\bar{\mathbf{x}},t)}{\mathrm{d}t} = \frac{\partial V(\bar{\mathbf{x}},t)}{\partial t} + V_{\mathbf{x}}^{\mathsf{T}}\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \frac{\partial V(\bar{\mathbf{x}},t)}{\partial t} + V_{\mathbf{x}}^{\mathsf{T}} F(\bar{\mathbf{x}},\bar{\mathbf{u}},\bar{\mathbf{v}},t). \tag{148}$$

Thus, we get

$$\frac{\partial V(\bar{\mathbf{x}},t)}{\partial t} = \frac{\mathrm{d}V(\bar{\mathbf{x}},t)}{\mathrm{d}t} - V_{\mathbf{x}}^{\mathsf{T}} F(\bar{\mathbf{x}},\bar{\mathbf{u}},\bar{\mathbf{v}},t). \tag{149}$$

Similarly,

$$\frac{\partial V_{\mathbf{x}}(\bar{\mathbf{x}},t)}{\partial t} = \frac{\mathrm{d}V_{\mathbf{x}}(\bar{\mathbf{x}},t)}{\mathrm{d}t} - V_{\mathbf{xx}} F(\bar{\mathbf{x}},\bar{\mathbf{u}},\bar{\mathbf{v}},t). \tag{150}$$

Finally, the partial time derivative of the Hessian of the value function takes the form:

$$\frac{\partial V_{\mathbf{xx}}(\bar{\mathbf{x}},t)}{\partial t} = \frac{\mathrm{d}V_{\mathbf{xx}}(\bar{\mathbf{x}},t)}{\mathrm{d}t} - \sum_{i=1}^{n} V_{\mathbf{xxx}}^{(i)} F^{(i)}, \tag{151}$$

where $V_{\mathbf{xxx}}^{(i)}$ denotes the Hessian matrix of the i-th element of $V_{\mathbf{x}}$ and $F^{(i)}$ denotes the i-th element of $F(\bar{\mathbf{x}},\bar{\mathbf{u}},\bar{\mathbf{v}},t)$. Henceforth, the arguments for the functions V, F, etc, are omitted for brevity, and they are evaluated at the nominal trajectory $(\bar{\mathbf{x}},\bar{\mathbf{u}},\bar{\mathbf{v}})$ unless otherwise specified.

The left-hand side of (143) then becomes

$$\begin{aligned} -\frac{\partial V(\mathbf{x},t)}{\partial t} = &-\frac{\mathrm{d}V}{\mathrm{d}t} - \frac{\mathrm{d}V_{\mathbf{x}}^{\mathsf{T}}}{\mathrm{d}t}\delta\mathbf{x} - \frac{1}{2}\delta\mathbf{x}^{\mathsf{T}}\frac{\mathrm{d}V_{\mathbf{xx}}}{\mathrm{d}t}\delta\mathbf{x} \\ &+ V_{\mathbf{x}}^{\mathsf{T}} F + \delta\mathbf{x}^{\mathsf{T}} V_{\mathbf{xx}} F + \frac{1}{2}\delta\mathbf{x}^{\mathsf{T}}\left(\sum_{i=1}^{n} V_{\mathbf{xxx}}^{(i)} F^{(i)}\right)\delta\mathbf{x}. \end{aligned} \tag{152}$$

Now we turn to the expansion of the right-hand side of (143).

$$\begin{aligned} &\mathcal{L}(\mathbf{x},\mathbf{u},\mathbf{v},t) \\ &= \mathcal{L}(\bar{\mathbf{x}}+\delta\mathbf{x},\bar{\mathbf{u}}+\delta\mathbf{u},\bar{\mathbf{v}}+\delta\mathbf{v},t) \\ &= \mathcal{L} + \mathcal{L}_{\mathbf{x}}^{\mathsf{T}}\delta\mathbf{x} + \mathcal{L}_{\mathbf{u}}^{\mathsf{T}}\delta\mathbf{u} + \mathcal{L}_{\mathbf{v}}^{\mathsf{T}}\delta\mathbf{v} + \frac{1}{2}\begin{bmatrix} \delta\mathbf{x}^{\mathsf{T}} & \delta\mathbf{u}^{\mathsf{T}} & \delta\mathbf{v}^{\mathsf{T}} \end{bmatrix} \begin{bmatrix} \mathcal{L}_{\mathbf{xx}} & \mathcal{L}_{\mathbf{xu}} & \mathcal{L}_{\mathbf{xv}} \\ \mathcal{L}_{\mathbf{ux}} & \mathcal{L}_{\mathbf{uu}} & \mathcal{L}_{\mathbf{uv}} \\ \mathcal{L}_{\mathbf{vx}} & \mathcal{L}_{\mathbf{vu}} & \mathcal{L}_{\mathbf{vv}} \end{bmatrix} \begin{bmatrix} \delta\mathbf{x} \\ \delta\mathbf{u} \\ \delta\mathbf{v} \end{bmatrix}. \end{aligned} \tag{153}$$

By expanding $V_{\mathbf{x}}(\mathbf{x}, t)$, we have

$$V_{\mathbf{x}}(\mathbf{x}, t) = V_{\mathbf{x}}(\bar{\mathbf{x}} + \delta\mathbf{x}, t) = V_{\mathbf{x}} + V_{\mathbf{xx}}\delta\mathbf{x} + \frac{1}{2}\mathcal{U}, \tag{154}$$

where $\mathcal{U} \in \mathbb{R}^n$ and each element of $\mathcal{U}$ is defined as

$$\mathcal{U}^{(i)} = \delta\mathbf{x}^{\intercal} V_{\mathbf{xxx}}^{(i)} \delta\mathbf{x}.$$

The dynamic equation is expanded up to the first order, that is,

$$\begin{aligned} F(\mathbf{x}, \mathbf{u}, \mathbf{v}, t) &= F(\bar{\mathbf{x}} + \delta\mathbf{x}, \bar{\mathbf{u}} + \delta\mathbf{u}, \bar{\mathbf{v}} + \delta\mathbf{v}, t) \\ &= F + F_{\mathbf{x}}\delta\mathbf{x} + F_{\mathbf{u}}\delta\mathbf{u} + F_{\mathbf{v}}\delta\mathbf{v}. \end{aligned} \tag{155}$$

Therefore, the right-hand side of (143) can be expressed as

$$\begin{aligned} &\min_{\mathbf{u}} \max_{\mathbf{v}} \left[\mathcal{L}(\mathbf{x}, \mathbf{u}, \mathbf{v}, t) + V_{\mathbf{x}}^{\intercal} F(\mathbf{x}, \mathbf{u}, \mathbf{v}, t) \right] \\ &= \min_{\delta\mathbf{u}} \max_{\delta\mathbf{v}} \left[\mathcal{L} + \mathcal{L}_{\mathbf{x}}^{\intercal}\delta\mathbf{x} + \mathcal{L}_{\mathbf{u}}^{\intercal}\delta\mathbf{u} + \mathcal{L}_{\mathbf{v}}^{\intercal}\delta\mathbf{v} + \frac{1}{2} \begin{bmatrix} \delta\mathbf{x} \\ \delta\mathbf{u} \\ \delta\mathbf{v} \end{bmatrix}^{\intercal} \begin{bmatrix} \mathcal{L}_{\mathbf{xx}} & \mathcal{L}_{\mathbf{xu}} & \mathcal{L}_{\mathbf{xv}} \\ \mathcal{L}_{\mathbf{ux}} & \mathcal{L}_{\mathbf{uu}} & \mathcal{L}_{\mathbf{uv}} \\ \mathcal{L}_{\mathbf{vx}} & \mathcal{L}_{\mathbf{vu}} & \mathcal{L}_{\mathbf{vv}} \end{bmatrix} \begin{bmatrix} \delta\mathbf{x} \\ \delta\mathbf{u} \\ \delta\mathbf{v} \end{bmatrix} \right. \\ &+ V_{\mathbf{x}}^{\intercal} F + V_{\mathbf{x}}^{\intercal} F_{\mathbf{x}}\delta\mathbf{x} + V_{\mathbf{x}}^{\intercal} F_{\mathbf{u}}\delta\mathbf{u} + V_{\mathbf{x}}^{T} F_{\mathbf{v}}\delta\mathbf{v} + \delta\mathbf{x}^{\intercal} V_{\mathbf{xx}} F + \delta\mathbf{x}^{\intercal} V_{\mathbf{xx}} F_{\mathbf{x}}\delta\mathbf{x} + \delta\mathbf{x}^{\intercal} V_{\mathbf{xx}} F_{\mathbf{u}}\delta\mathbf{u} \\ &\left. + \delta\mathbf{x}^{\intercal} V_{\mathbf{xx}} F_{\mathbf{v}}\delta\mathbf{v} + \frac{1}{2}\mathcal{U}^{\intercal} F + o(\delta\mathbf{x}^{\intercal}\delta\mathbf{x}) \right]. \end{aligned} \tag{156}$$

Note that the term $\frac{1}{2}\mathcal{U}^{\intercal} F$ can be written as follows

$$\frac{1}{2}\mathcal{U}^{\intercal}F = \frac{1}{2}\sum_{i=1}^{n}\left(\delta\mathbf{x}^{\intercal}V_{\mathbf{xxx}}^{(i)}\delta\mathbf{x}F^{(i)}\right)$$
$$= \frac{1}{2}\delta\mathbf{x}^{\intercal}\left(\sum_{i=1}^{n}V_{\mathbf{xxx}}^{(i)}F^{(i)}\right)\delta\mathbf{x}.$$

After equating (152) with (156), and canceling repeated terms, we obtain

$$-\frac{\mathrm{d}V}{\mathrm{d}t} - \delta\mathbf{x}^{\intercal}\frac{\mathrm{d}V_{\mathbf{x}}}{\mathrm{d}t} - \frac{1}{2}\delta\mathbf{x}^{\intercal}\frac{\mathrm{d}V_{\mathbf{xx}}}{\mathrm{d}t}\delta\mathbf{x}$$
$$= \min_{\delta\mathbf{u}}\max_{\delta\mathbf{v}}\left\{\mathcal{L} + \mathcal{L}_{\mathbf{x}}^{\intercal}\delta\mathbf{x} + \mathcal{L}_{\mathbf{u}}^{\intercal}\delta\mathbf{u} + \mathcal{L}_{\mathbf{v}}^{\intercal}\delta\mathbf{v} + \frac{1}{2}\begin{bmatrix}\delta\mathbf{x}\\ \delta\mathbf{u}\\ \delta\mathbf{v}\end{bmatrix}^{\intercal}\begin{bmatrix}\mathcal{L}_{\mathbf{xx}} & \mathcal{L}_{\mathbf{xu}} & \mathcal{L}_{\mathbf{xv}}\\ \mathcal{L}_{\mathbf{ux}} & \mathcal{L}_{\mathbf{uu}} & \mathcal{L}_{\mathbf{uv}}\\ \mathcal{L}_{\mathbf{vx}} & \mathcal{L}_{\mathbf{vu}} & \mathcal{L}_{\mathbf{vv}}\end{bmatrix}\begin{bmatrix}\delta\mathbf{x}\\ \delta\mathbf{u}\\ \delta\mathbf{v}\end{bmatrix}\right.$$
$$\left. + V_{\mathbf{x}}^{\intercal}F_{\mathbf{x}}\delta\mathbf{x} + V_{\mathbf{x}}^{\intercal}F_{\mathbf{u}}\delta\mathbf{u} + V_{\mathbf{x}}^{\intercal}F_{\mathbf{v}}\delta\mathbf{v} + \delta\mathbf{x}^{\intercal}V_{\mathbf{xx}}F_{\mathbf{x}}\delta\mathbf{x} + \delta\mathbf{x}^{\intercal}V_{\mathbf{xx}}F_{\mathbf{u}}\delta\mathbf{u} + \delta\mathbf{x}^{\intercal}V_{\mathbf{xx}}F_{\mathbf{v}}\delta\mathbf{v}\right\}$$
$$= \min_{\delta\mathbf{u}}\max_{\delta\mathbf{v}}\left\{\mathcal{L} + \delta\mathbf{x}^{\intercal}Q_{\mathbf{x}} + \delta\mathbf{u}^{\intercal}Q_{\mathbf{u}} + \delta\mathbf{v}^{\intercal}Q_{\mathbf{v}} + \frac{1}{2}\delta\mathbf{x}^{\intercal}Q_{\mathbf{xx}}\delta\mathbf{x} + \frac{1}{2}\delta\mathbf{u}^{\intercal}Q_{\mathbf{uu}}\delta\mathbf{u}\right.$$
$$\left. + \frac{1}{2}\delta\mathbf{v}^{\intercal}Q_{\mathbf{vv}}\delta\mathbf{v} + \delta\mathbf{u}^{\intercal}Q_{\mathbf{ux}}\delta\mathbf{x} + \delta\mathbf{v}^{\intercal}Q_{\mathbf{vx}}\delta\mathbf{x} + \delta\mathbf{u}^{\intercal}Q_{\mathbf{uv}}\delta\mathbf{v}\right\}, \tag{157}$$

where

$$
\begin{aligned}
Q_{\mathbf{x}} &= F_{\mathbf{x}}^{\mathsf{T}} V_{\mathbf{x}} + \mathcal{L}_{\mathbf{x}}, \\
Q_{\mathbf{u}} &= F_{\mathbf{u}}^{\mathsf{T}} V_{\mathbf{x}} + \mathcal{L}_{\mathbf{u}}, \\
Q_{\mathbf{v}} &= F_{\mathbf{v}}^{\mathsf{T}} V_{\mathbf{x}} + \mathcal{L}_{\mathbf{v}}, \\
Q_{\mathbf{xx}} &= \mathcal{L}_{\mathbf{xx}} + 2V_{\mathbf{xx}} F_{\mathbf{x}}, \\
Q_{\mathbf{uu}} &= \mathcal{L}_{\mathbf{uu}}, \\
Q_{\mathbf{vv}} &= \mathcal{L}_{\mathbf{vv}}, \\
Q_{\mathbf{ux}} &= F_{\mathbf{u}}^{\mathsf{T}} V_{\mathbf{xx}} + \frac{1}{2}\mathcal{L}_{\mathbf{ux}} + \frac{1}{2}\mathcal{L}_{\mathbf{xu}}^{\mathsf{T}}, \\
Q_{\mathbf{vx}} &= F_{\mathbf{v}}^{\mathsf{T}} V_{\mathbf{xx}} + \frac{1}{2}\mathcal{L}_{\mathbf{vx}} + \frac{1}{2}\mathcal{L}_{\mathbf{xv}}^{\mathsf{T}}, \\
Q_{\mathbf{uv}} &= \frac{1}{2}\mathcal{L}_{\mathbf{uv}} + \frac{1}{2}\mathcal{L}_{\mathbf{vu}}^{\mathsf{T}}.
\end{aligned}
\tag{158}
$$

To find the optimal control $\delta\mathbf{u}^*$ and $\delta\mathbf{v}^*$, we compute the gradients of the expression in (157) with respect to $\delta\mathbf{u}$ and $\delta\mathbf{v}$, respectively, and make them equal to zero to obtain:

$$
\delta\mathbf{u}^* = -Q_{\mathbf{uu}}^{-1}\Big(Q_{\mathbf{ux}}\delta\mathbf{x} + Q_{\mathbf{uv}}\delta\mathbf{v} + Q_{\mathbf{u}}\Big), \tag{159}
$$

$$
\delta\mathbf{v}^* = -Q_{\mathbf{vv}}^{-1}\Big(Q_{\mathbf{vx}}\delta\mathbf{x} + Q_{\mathbf{vu}}\delta\mathbf{u} + Q_{\mathbf{v}}\Big), \tag{160}
$$

where $Q_{\mathbf{vu}} = Q_{\mathbf{uv}}^{\mathsf{T}}$.

Notice that $\delta\mathbf{v}$ is still in the previous expression of $\delta\mathbf{u}^*$. We need to replace the $\delta\mathbf{v}$ term in (159) with (160) and solve for $\delta\mathbf{u}^*$. Similarly, we can solve for $\delta\mathbf{v}^*$. The final expressions for $\delta\mathbf{u}^*$ and $\delta\mathbf{v}^*$ are specified as follows:

$$
\delta\mathbf{u}^* = \mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}\delta\mathbf{x} \quad \text{and} \quad \delta\mathbf{v}^* = \mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{v}}\delta\mathbf{x}, \tag{161}
$$

with the feed-forward gains $\mathbf{l_v}, \mathbf{l_u}$ and feedback gains $\mathbf{L_v}, \mathbf{L_u}$ defined as:

$$\mathbf{l_u} = -\Big(Q_{\mathbf{uu}} - Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{vu}}\Big)^{-1}\Big(Q_{\mathbf{u}} - Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{v}}\Big), \tag{162}$$

$$\mathbf{l_v} = -\Big(Q_{\mathbf{vv}} - Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{uv}}\Big)^{-1}\Big(Q_{\mathbf{v}} - Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{u}}\Big), \tag{163}$$

$$\mathbf{L_u} = -\Big(Q_{\mathbf{uu}} - Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{vu}}\Big)^{-1}\Big(Q_{\mathbf{ux}} - Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{vx}}\Big), \tag{164}$$

$$\mathbf{L_v} = -\Big(Q_{\mathbf{vv}} - Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{uv}}\Big)^{-1}\Big(Q_{\mathbf{vx}} - Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{ux}}\Big). \tag{165}$$

In many applications in engineering, we can design the cost function. In order to see the effect that the design of the cost function has in the feed-forward and feedback gains, we recall that $Q_{\mathbf{uu}} = \mathcal{L}_{\mathbf{uu}}$ and $Q_{\mathbf{vv}} = \mathcal{L}_{\mathbf{vv}}$. Moreover, since $\mathcal{L}_{\mathbf{uu}}, \mathcal{L}_{\mathbf{vv}}$ are design parameters, we can choose them such that $\mathcal{L}_{\mathbf{uu}} > 0$ and $\mathcal{L}_{\mathbf{vv}} < 0$. Note also that the positive definiteness of $\mathcal{L}_{\mathbf{uu}}$ and negative definiteness of $\mathcal{L}_{\mathbf{vv}}$ are required since the role of the first controller/player is to minimize the cost while the role of the second controller/player is to maximize it. Given new $Q_{\mathbf{uu}} > 0$ and $Q_{\mathbf{vv}} < 0$ we have the following expressions

$$Q_{\mathbf{uu}} - Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{vu}} > 0 \Rightarrow \Big(Q_{\mathbf{uu}} - Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{vu}}\Big)^{-1} > 0, \tag{166}$$

$$Q_{\mathbf{vv}} - Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{uv}} < 0 \Rightarrow \Big(Q_{\mathbf{vv}} - Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{uv}}\Big)^{-1} < 0. \tag{167}$$

The matrix inequalities (166) and (167) show that the feed-forward and feedback part of the control policies of the two players will operate such that the first player aims to reduce the cost while the second player aims to increase it. An interesting characteristic of trajectory optimization methods such as DDP is that they provide the locally optimal state trajectory, optimal feed-forward control and locally optimal feedback gains. Here we show how the feed-forward and feedback parts of the

correction terms $\delta\mathbf{u}$ and $\delta\mathbf{v}$ depend on the design of the cost function. In the simulation section we demonstrate the effect of the cost function on the feed-forward and feedback parts of the minimizing control policy for different values of $\mathcal{L}_{\mathbf{vv}}$.

7.2.1 Derivation of the Backward Propagation of the Value Function

The next step is to substitute the optimal control (159) and disturbance (destabilizing control) (160) to the HJBI equation (143) in order to find the update law of the value function and its first and second order partial derivatives. Specifically, we have:

$$\begin{aligned}
&-\frac{\mathrm{d}V}{\mathrm{d}t}-\delta\mathbf{x}^{\mathsf{T}}\frac{\mathrm{d}V_{\mathbf{x}}}{\mathrm{d}t}-\frac{1}{2}\delta\mathbf{x}^{\mathsf{T}}\frac{\mathrm{d}V_{\mathbf{xx}}}{\mathrm{d}t}\delta\mathbf{x}\\
&=\mathcal{L}+\delta\mathbf{x}^{\mathsf{T}}Q_{\mathbf{x}}+\delta\mathbf{u}^{*\mathsf{T}}Q_{\mathbf{u}}+\delta\mathbf{v}^{*\mathsf{T}}Q_{\mathbf{v}}+\delta\mathbf{u}^{*\mathsf{T}}Q_{\mathbf{ux}}\delta\mathbf{x}+\frac{1}{2}\delta\mathbf{u}^{*\mathsf{T}}Q_{\mathbf{uu}}\delta\mathbf{u}^{*}+\delta\mathbf{u}^{*\mathsf{T}}Q_{\mathbf{uv}}\delta\mathbf{v}^{*}\\
&+\frac{1}{2}\delta\mathbf{v}^{*\mathsf{T}}Q_{\mathbf{vv}}\delta\mathbf{v}^{*}+\delta\mathbf{v}^{*\mathsf{T}}Q_{\mathbf{vx}}\delta\mathbf{x}+\frac{1}{2}\delta\mathbf{x}^{\mathsf{T}}Q_{\mathbf{xx}}\delta\mathbf{x}\\
&=\mathcal{L}+\delta\mathbf{x}^{\mathsf{T}}Q_{\mathbf{x}}+\Big(\mathbf{l}_{\mathbf{u}}+\mathbf{L}_{\mathbf{u}}\delta\mathbf{x}\Big)^{\mathsf{T}}Q_{\mathbf{u}}+\Big(\mathbf{l}_{\mathbf{v}}+\mathbf{L}_{\mathbf{v}}\delta\mathbf{x}\Big)^{\mathsf{T}}Q_{\mathbf{v}}+\Big(\mathbf{l}_{\mathbf{u}}+\mathbf{L}_{\mathbf{u}}\delta\mathbf{x}\Big)^{\mathsf{T}}Q_{\mathbf{ux}}\delta\mathbf{x}\\
&+\Big(\mathbf{l}_{\mathbf{v}}+\mathbf{L}_{\mathbf{v}}\delta\mathbf{x}\Big)^{\mathsf{T}}Q_{\mathbf{vx}}\delta\mathbf{x}+\frac{1}{2}\delta\mathbf{x}^{\mathsf{T}}Q_{\mathbf{xx}}\delta\mathbf{x}+\Big(\mathbf{l}_{\mathbf{u}}+\mathbf{L}_{\mathbf{u}}\delta\mathbf{x}\Big)^{\mathsf{T}}Q_{\mathbf{uv}}\Big(\mathbf{l}_{\mathbf{v}}+\mathbf{L}_{\mathbf{v}}\delta\mathbf{x}\Big)\\
&+\frac{1}{2}\Big(\mathbf{l}_{\mathbf{u}}+\mathbf{L}_{\mathbf{u}}\delta\mathbf{x}\Big)^{\mathsf{T}}Q_{\mathbf{uu}}\Big(\mathbf{l}_{\mathbf{u}}+\mathbf{L}_{\mathbf{u}}\delta\mathbf{x}\Big)+\frac{1}{2}\Big(\mathbf{l}_{\mathbf{v}}+\mathbf{L}_{\mathbf{v}}\delta\mathbf{x}\Big)^{\mathsf{T}}Q_{\mathbf{vv}}\Big(\mathbf{l}_{\mathbf{v}}+\mathbf{L}_{\mathbf{v}}\delta\mathbf{x}\Big).
\end{aligned}\tag{168}$$

After collecting terms on the right-hand side of (168) as zeroth order, first order and second order expressions of $\delta\mathbf{x}$, we can equate the coefficients of $\delta\mathbf{x}$ on the left-hand side and right-hand side of (168) and readily obtain the backward propagation equations with respect to the value function and its first and second order partial derivatives. These backward differential equations are expressed as follows

$$\begin{aligned}
-\frac{\mathrm{d}V}{\mathrm{d}t}&=\mathcal{L}+\mathbf{l}_{\mathbf{u}}^{\mathsf{T}}Q_{\mathbf{u}}+\mathbf{l}_{\mathbf{v}}^{\mathsf{T}}Q_{\mathbf{v}}+\frac{1}{2}\mathbf{l}_{\mathbf{u}}Q_{\mathbf{uu}}\mathbf{l}_{\mathbf{u}}+\mathbf{l}_{\mathbf{u}}^{\mathsf{T}}Q_{\mathbf{uv}}\mathbf{l}_{\mathbf{v}}+\frac{1}{2}\mathbf{l}_{\mathbf{v}}^{\mathsf{T}}Q_{\mathbf{vv}}\mathbf{l}_{\mathbf{v}},\\
-\frac{\mathrm{d}V_{\mathbf{x}}}{\mathrm{d}t}&=Q_{\mathbf{x}}+\mathbf{L}_{\mathbf{u}}^{\mathsf{T}}Q_{\mathbf{u}}+\mathbf{L}_{\mathbf{v}}^{\mathsf{T}}Q_{\mathbf{v}}+Q_{\mathbf{ux}}^{\mathsf{T}}\mathbf{l}_{\mathbf{u}}+Q_{\mathbf{vx}}^{\mathsf{T}}\mathbf{l}_{\mathbf{v}}+\mathbf{L}_{\mathbf{u}}^{\mathsf{T}}Q_{\mathbf{uu}}\mathbf{l}_{\mathbf{u}}+\mathbf{L}_{\mathbf{u}}^{\mathsf{T}}Q_{\mathbf{uv}}\mathbf{l}_{\mathbf{v}}\\
&\quad+\mathbf{L}_{\mathbf{v}}^{\mathsf{T}}Q_{\mathbf{vu}}\mathbf{l}_{\mathbf{u}}+\mathbf{L}_{\mathbf{v}}^{\mathsf{T}}Q_{\mathbf{vv}}\mathbf{l}_{\mathbf{v}},\\
-\frac{\mathrm{d}V_{\mathbf{xx}}}{\mathrm{d}t}&=2\mathbf{L}_{\mathbf{u}}^{\mathsf{T}}Q_{\mathbf{ux}}+2\mathbf{L}_{\mathbf{v}}^{\mathsf{T}}Q_{\mathbf{vx}}+2\mathbf{L}_{\mathbf{v}}^{\mathsf{T}}Q_{\mathbf{vu}}\mathbf{L}_{\mathbf{u}}+\mathbf{L}_{\mathbf{u}}^{\mathsf{T}}Q_{\mathbf{uu}}\mathbf{L}_{\mathbf{u}}+\mathbf{L}_{\mathbf{v}}^{\mathsf{T}}Q_{\mathbf{vv}}\mathbf{L}_{\mathbf{v}}+Q_{\mathbf{xx}}.
\end{aligned}\tag{169}$$

In many applications in engineering the cost function is designed such that the terms $\mathcal{L}_{\mathbf{vu}} = \mathcal{L}_{\mathbf{uv}}^{\intercal} = 0$. In this case the differential equations for the backward propagation of the value function are simplified as follows

$$\begin{aligned}
-\frac{\mathrm{d}V}{\mathrm{d}t} &= \mathcal{L} + \mathbf{l}_{\mathbf{u}}^{\intercal} Q_{\mathbf{u}} + \mathbf{l}_{\mathbf{v}}^{\intercal} Q_{\mathbf{v}} + \frac{1}{2}\mathbf{l}_{\mathbf{u}} Q_{\mathbf{uu}} \mathbf{l}_{\mathbf{u}} + \frac{1}{2}\mathbf{l}_{\mathbf{v}}^{\intercal} Q_{\mathbf{vv}} \mathbf{l}_{\mathbf{v}}, \\
-\frac{\mathrm{d}V_{\mathbf{x}}}{\mathrm{d}t} &= Q_{\mathbf{x}} + \mathbf{L}_{\mathbf{u}}^{\intercal} Q_{\mathbf{u}} + \mathbf{L}_{\mathbf{v}}^{\intercal} Q_{\mathbf{v}} + Q_{\mathbf{ux}}^{\intercal} \mathbf{l}_{\mathbf{u}} + Q_{\mathbf{vx}}^{\intercal} \mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{u}}^{\intercal} Q_{\mathbf{uu}} \mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{v}}^{\intercal} Q_{\mathbf{vv}} \mathbf{l}_{\mathbf{v}}, \\
-\frac{\mathrm{d}V_{\mathbf{xx}}}{\mathrm{d}t} &= 2\mathbf{L}_{\mathbf{u}}^{\intercal} Q_{\mathbf{ux}} + 2\mathbf{L}_{\mathbf{v}}^{\intercal} Q_{\mathbf{vx}} + \mathbf{L}_{\mathbf{u}}^{\intercal} Q_{\mathbf{uu}} \mathbf{L}_{\mathbf{u}} + \mathbf{L}_{\mathbf{v}}^{\intercal} Q_{\mathbf{vv}} \mathbf{L}_{\mathbf{v}} + Q_{\mathbf{xx}}.
\end{aligned} \tag{170}$$

The backward differential equations in (169) and (170) are different with respect to the corresponding backward equations in the discrete time formulation of min-max DDP in [105] and [104]. Besides the form of the backward differential equations, one of the major differences between the discrete and continuous time formulations is on the specification of the terms $Q_{\mathbf{uu}}$ and $Q_{\mathbf{vv}}$. In the continuous case these terms are specified by $\mathcal{L}_{\mathbf{uu}}$ and $\mathcal{L}_{\mathbf{vv}}$ and therefore they are completely specified by the user. This is not the case with the discrete time formulation of min-max DDP (see equations (10) and (11) in [104]) in which the terms $Q_{\mathbf{uu}}$ and $Q_{\mathbf{vv}}$ are also functions of $V_{\mathbf{xx}}$, besides $\mathcal{L}_{\mathbf{uu}}$ and $\mathcal{L}_{\mathbf{vv}}$. The result of this observation is that for the discrete time case the positive definiteness of $Q_{\mathbf{uu}}$ and the negative definiteness of $Q_{\mathbf{vv}}$ along the nominal trajectories are not guaranteed. As we show in our derivation, this is not the case with the continuous time formulation of GT-DDP and therefore the continuous version is numerically more stable that the discrete time.

7.3 Terminal Condition and the Minimax DDP Algorithm

In this section, we first specify the terminal condition for the backward differential equations with respect to the value function and its first and second order partial derivatives. Then we put all the pieces together and present the algorithm of minimax DDP.

At the final time, we have (144). By taking the Taylor series expansions around

$\bar{\mathbf{x}}(t_f)$ we get

$$\begin{aligned}\phi(\mathbf{x}(t_f), t_f) &= \phi(\bar{\mathbf{x}}(t_f) + \delta\mathbf{x}(t_f), t_f) \\ &\approx \phi(\bar{\mathbf{x}}(t_f), t_f) + \delta\mathbf{x}(t_f)^\intercal \phi_{\mathbf{x}}(\bar{\mathbf{x}}(t_f), t_f) + \delta\mathbf{x}(t_f)^\intercal \phi_{\mathbf{xx}}(\bar{\mathbf{x}}(t_f), t_f)\delta\mathbf{x}(t_f).\end{aligned} \tag{171}$$

Therefore, the boundary conditions at $t = t_f$ for the backward differential equations are

$$\begin{aligned}V(t_f) &= \phi(\bar{\mathbf{x}}(t_f), t_f), \\ V_{\mathbf{x}}(t_f) &= \phi_{\mathbf{x}}(\bar{\mathbf{x}}(t_f), t_f), \\ V_{\mathbf{xx}}(t_f) &= \phi_{\mathbf{xx}}(\bar{\mathbf{x}}(t_f), t_f).\end{aligned} \tag{172}$$

The GT-DDP algorithm is provided in Algorithm 2.

Algorithm 2 GT-DDP Algorithm

Input: Initial condition of the dynamics $\mathbf{x}_0$, initial stabilizing control $\bar{\mathbf{u}}$ and destabilizing control $\bar{\mathbf{v}}$, final time t_f, multiplier γ and a positive constant ϵ.
Output: Optimal stabilizing control $\mathbf{u}^*$, optimal destabilizing control $\mathbf{v}^*$ and the corresponding optimal gains $\mathbf{l_u}, \mathbf{L_u}, \mathbf{l_v}, \mathbf{L_v}$.

procedure UPDATE_CONTROL($\mathbf{x}_0$, $\bar{\mathbf{u}}$, $\bar{\mathbf{v}}$, t_f)
  while $\phi(\bar{\mathbf{x}}(t_f), t_f) > \epsilon$ **do**
    Get the initial trajectory $\bar{\mathbf{x}}$ by integrating controlled dynamics forward with $\mathbf{x}_0$, $\bar{\mathbf{u}}$ and $\bar{\mathbf{v}}$;
    Compute the value of $V, V_{\mathbf{x}}, V_{\mathbf{xx}}$ at t_f according to (172);
    Integrate backward the Riccati equations (169);
    Compute $\mathbf{l_u}, \mathbf{L_u}, \mathbf{l_v}, \mathbf{L_v}$ from (162) through (165);
    Integrate (146) forward by replacing $\delta\mathbf{u}$ and $\delta\mathbf{v}$ with $(\mathbf{l_u} + \mathbf{L_u}\delta\mathbf{x})$ and $(\mathbf{l_v} + \mathbf{L_v}\delta\mathbf{x})$, respectively, to get $\delta\mathbf{x}(t)$;
    Compute $\delta\mathbf{u} = \mathbf{l_u} + \mathbf{L_u}\delta\mathbf{x}$ and $\delta\mathbf{v} = \mathbf{l_v} + \mathbf{L_v}\delta\mathbf{x}$;
    Update control $\mathbf{u}^* = \mathbf{u}^* + \gamma\delta\mathbf{u}$, where $\gamma \in [0, 1]$;
    Set $\bar{\mathbf{u}} = \mathbf{u}^*$ and $\bar{\mathbf{v}} = \bar{\mathbf{v}}^*$;
  end while
  return $\mathbf{u}^*$, $\mathbf{v}^*$, $\mathbf{l_u}, \mathbf{L_u}, \mathbf{l_v}, \mathbf{L_v}$.
end procedure

7.4 Simulation Results

In this section, we apply our algorithm to two systems, namely, the inverted pendulum and the two-player pursuit-evasion game under external flow field. The dynamics of the first problem is affine in control and the cost is quadratic in control, whereas in the second problem, the dynamics is nonlinear in control and the cost function is non-quadratic.

7.4.1 Inverted Pendulum Problem

We first apply our algorithm on the inverted pendulum with conflicting controls. In particular, the dynamics is given by $I\ddot{\theta} + b\dot{\theta} - mg\ell\sin\theta = \mathbf{u} - \mathbf{v}$, where the parameters are chosen in the simulations as $m = 1Kg$, $\ell = 0.5m$, $b = 0.1$, $I = ml^2$, $g = 9.81Kg \cdot m/sec^2$. Our goal is to bring the pendulum from the initial state $[\theta, \dot{\theta}] = [\pi, 0]$ to $[\theta, \dot{\theta}] = [0, 0]$. The cost function is given by $J = \mathbf{x}(t_f)^\intercal Q_f \mathbf{x}(t_f) + \int_0^{t_f} (\mathbf{u}^{TT} R_{\mathbf{u}} \mathbf{u} - \mathbf{v}^\intercal R_{\mathbf{v}} \mathbf{v})$, where $\mathbf{x} = [\theta, \dot{\theta}]^{TT}$, $Q_f = \begin{bmatrix} 100, & 0 \\ 0, & 5 \end{bmatrix}$ and $R_{\mathbf{u}} = 0.1$, $R_{\mathbf{v}} = 0.2$.

We set the initial control to be $\mathbf{u} \equiv 0$, $\mathbf{v} \equiv 0$, the terminal time to be $t_f = 0.5$ and the multiplier $\gamma = 0.8$. As can be seen in Figure 41, the cost converges in 4 iterations. We include 10 iterations to ensure convergence. Figure 42 presents the optimal controls of $\mathbf{u}$ and $\mathbf{v}$ at the 10th iteration, as well as the corresponding optimal trajectories of the states $\theta, \dot{\theta}$.

7.4.2 Inverted Pendulum Problem with Stochastic Disturbances

In this subsection, we utilize GT-DDP to guide the inverted pendulum to the desired state under the presence of stochastic disturbance that acts in the same channel as the control. Our goal is to analyze in simulation how the min-max formulation of GT-DDP affects the resulting feedfoward and feedback parts of the minimizing control policy. To this end, we consider the dynamics of the form $I\ddot{\theta} + b\dot{\theta} - mg\ell\sin\theta = \mathbf{u} + \omega$,

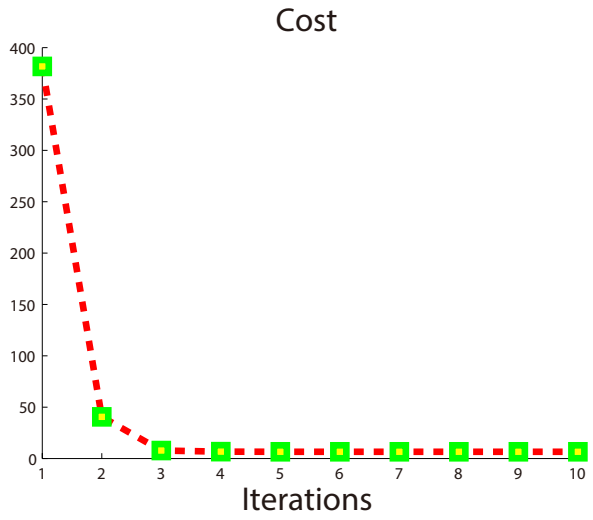


Figure 41: Cost per iteration of the inverted pendulum with conflicting controls.

where ω is a Gaussian noise with mean 0 and variance σ^2. The task for GT-DDP is to drive the inverted pendulum from the initial state $[\theta, \dot{\theta}] = [\pi, 0]$ to the final state $[\theta, \dot{\theta}] = [0, 0]$.

For our simulations, we set $\sigma = 4$ and pick $R_{\mathbf{v}} = 10, 0.2, 0.13$ for comparison. For every value of $R_{\mathbf{v}}$, we run the system with our modified control for 100 times. In Figure 43(a), we have three colored plots, where magenta, blue and cyan plots correspond to the case of $R_{\mathbf{v}} = 10, 0.2$ and 0.13, respectively. The plot of each color depicts the mean of 100 trajectories of θ with respect to time and we draw an error bar at every time step. Each error bar has a distance of the standard variance at that time step above and below the curve. Similarly, in Figure 43(b), we illustrate the mean and standard deviation of 100 trajectories of $\dot{\theta}$ with respect to time for the different values of $R_{\mathbf{v}}$.

Our simulations reveal the role of the min-max formulation of GT-DDP. In particular, Figures 43(a) and 43(b) illustrate that as $R_{\mathbf{v}}$ decreases, both the feed-forward and feedback parts of the control policy change. The feed-forward control steers the mean trajectory towards the desired state early for smaller values of $R_{\mathbf{v}}$. In addition, the locally optimal feedback gains reduce the variability of the trajectories as

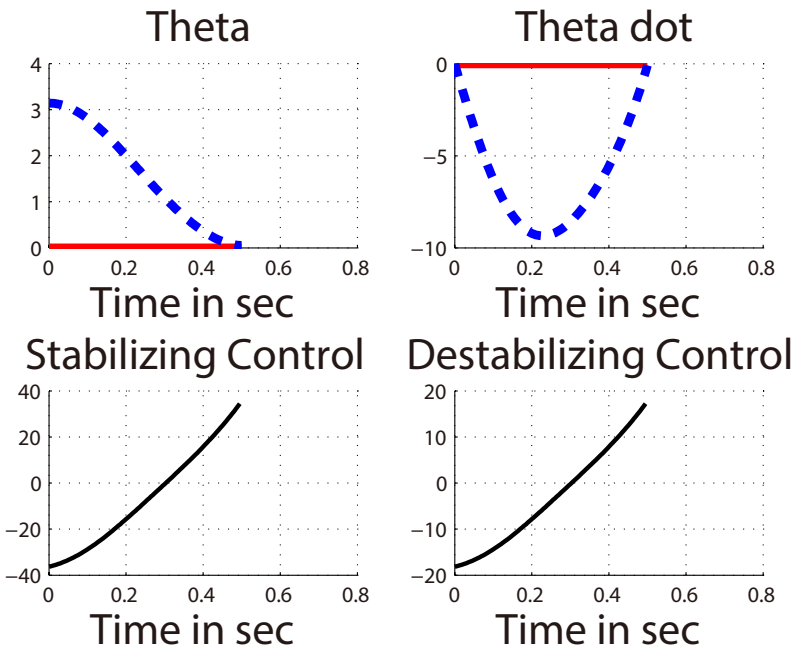
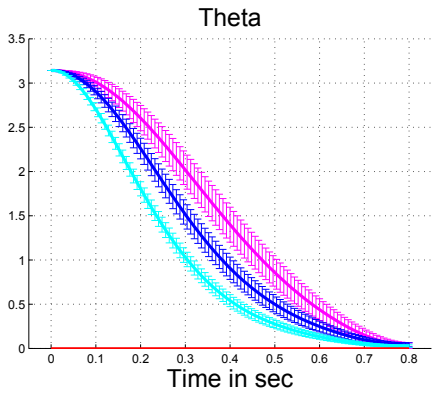


Figure 42: Optimal controls $\mathbf{u}$ and $\mathbf{v}$ in black at the bottom and the corresponding initial trajectories of the states $\theta, \dot{\theta}$ in dashed blue at the top. Red lines represent the desired terminal states.

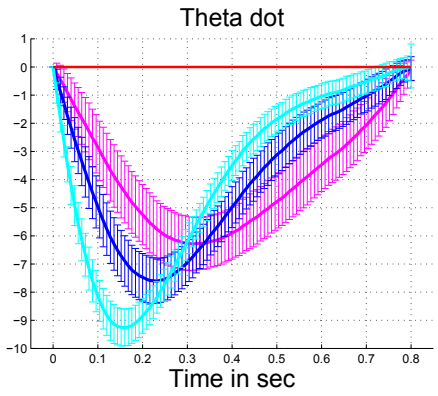
$R_{\mathbf{v}}$ decreases. The aforementioned observations indicate the interplay between the feed-forward and feedback part of the minimizing control policy under GT-DDP formulation and show how this formulation results in robust policies that shape both the mean and variance of optimal trajectories. We believe that these findings are important not only for the areas of engineering and robotics but also for modeling risk sensitive behaviors of bio-mechanical and neuromuscular systems.

7.4.3 Two-player Pursuit Evasion Game Under External Flow Field

Next, we apply the proposed algorithm to a two-player pursuit-evasion (PE) game subject to an external flow field. In this problem, one pursuer and one evader participates in a game on the plane under the environmental disturbance, namely, an external flow field. The terminal time t_f is fixed. The objective of the pursuer is to minimize the terminal distance between the two players, whereas the evader tries to maximize this value. Hence, the controls of the pursuer and the evader, represented



(a) Magenta, blue and cyan plots correspond to the case of $R_{\mathbf{v}} = 10, 0.2$ and 0.13, respectively. Each plot represents mean and standard variance of 100 trajectories of θ with respect to time. The red line at the bottom depicts the desired state $\theta = 0$.



(b) Magenta, blue and cyan plots correspond to the case of $R_{\mathbf{v}} = 10, 0.2$ and 0.13, respectively. Each plot represents mean and standard variance of 100 trajectories of $\dot{\theta}$ with respect to time. The red line depicts the desired state $\dot{\theta} = 0$.

Figure 43: Mean and error bar protrait of the states in the inverted pendulum problem with disturbances.

by $\mathbf{u}$ and $\mathbf{v}$, can be considered as the stabilizing and destabilizing control, respectively. The dynamics of the system is given by

$$\begin{aligned}
\dot{x}_P &= v_P \cos \mathbf{u} + w_1(x_P, y_P, t), \\
\dot{y}_P &= v_P \sin \mathbf{u} + w_2(x_P, y_P, t), \\
\dot{x}_E &= v_E \cos \mathbf{v} + w_1(x_E, y_E, t), \\
\dot{y}_E &= v_E \sin \mathbf{v} + w_2(x_E, y_E, t),
\end{aligned}$$

where (x_P, y_P) and (x_E, y_E) denote the positions of the pursuer and the evader, respectively. v_P and v_E represents the maximum speed of the pursuer and the evader, respectively. w_1 and w_2 are the components of an external spatial-temporal flow field along x-axis and y-axis, respectively. The full state of the systems is represented by $\mathbf{x} = [x_P, y_P, x_E, y_E]$. Let $\Delta x(t) = x_P(t) - x_E(t)$ and $\Delta y(t) = y_P(t) - y_E(t)$. Then the distance between the pursuer and the evader at the terminal time can be specified by

$\sqrt{\Delta x(t_f)^2 + \Delta y(t_f)^2}$. The problem of

$$\min_{\mathbf{u}} \max_{\mathbf{v}} \sqrt{\Delta x(t_f)^2 + \Delta y(t_f)^2}$$

is equivalent to

$$\min_{\mathbf{u}} \max_{\mathbf{v}} \frac{1}{2}\Big[\Delta x(t_f)^2 + \Delta y(t_f)^2\Big],$$

which is further equivalent to

$$\min_{\mathbf{u}} \max_{\mathbf{v}} \frac{1}{2}\Big[\Delta x(t_f)^2 + \Delta y(t_f)^2 - c\Big], \tag{173}$$

for any constant c.

Let $c = (\Delta x(0)^2 + \Delta y(0)^2)$. This c is a constant because the initial states $x(0)$ and $y(0)$ are constants and given a-priori. By substituting c in (173), we can reformulate the cost function as

$$\begin{aligned}
&\min_{\mathbf{u}} \max_{\mathbf{v}} J \\
&= \min_{\mathbf{u}} \max_{\mathbf{v}} \frac{1}{2}\bigg(\Big[\Delta x(t_f)^2 + \Delta y(t_f)^2\Big] - \Big[\Delta x(0)^2 + \Delta y(0)^2\Big]\bigg) \\
&= \min_{\mathbf{u}} \max_{\mathbf{v}} \int_0^{t_f} \frac{\mathrm{d}}{\mathrm{d}t}\Big[\frac{1}{2}[(x_P - x_E)^2 + (y_P - y_E)^2]\Big]\mathrm{d}t \\
&= \min_{\mathbf{u}} \max_{\mathbf{v}} \int_0^{t_f} \Big(\mathcal{L}_1(\mathbf{x}, \mathbf{u}, \mathbf{v}) + \mathcal{L}_2(\mathbf{x}, \mathbf{u}, \mathbf{v})\Big)\mathrm{d}t,
\end{aligned}$$

where

$$\begin{aligned}\mathcal{L}_1(\mathbf{x},\mathbf{u},\mathbf{v}) &= (x_P - x_E)(v_P\cos\mathbf{u} + w_1(x_P,y_P,t)) \\ &\quad - (x_P - x_E)(v_E\cos\mathbf{v} + w_1(x_E,y_E,t)), \\ \mathcal{L}_2(\mathbf{x},\mathbf{u},\mathbf{v}) &= (y_P - y_E)(v_P\sin\mathbf{u} + w_2(x_P,y_P,t)) \\ &\quad - (y_P - y_E)(v_E\sin\mathbf{v} + w_2(x_E,y_E,t)).\end{aligned}$$

The parameters v_P, v_E are specified as $v_P = 1, v_E = 0.8$ in our simulation. The initial condition is $[x_P(0), y_P(0), x_E(0), y_E(0)] = [0, 0, 1, 1]$.

Firstly, we consider the case where no external flow field exists, that is, $w_1(t) = w_2(t) \equiv 0$. It is well known in differential games that without the external flow field, the optimal controls for both players will move along their common line-of-sight (LoS), while the pursuer move towards the evader and the evader moves away from the pursuer [62]. The LoS is defined as the line passing through the pursuer's and evader's instantaneous positions. We want to recover this result in our simulation to verify the correctness of the algorithm. To this end, we start with the nominal control $\bar{\mathbf{u}} \equiv 0$ and $\bar{\mathbf{v}} \equiv 0$. The final time is fixed at $t_f = 2$. In this case, we set $\gamma = 1$. The convergence of the cost is achieved in 4 iterations, as shown in Figure 44(d). The optimal controls of the pursuer and the evader are illustrated in Figures 44(b) and 44(c), respectively. It is shown that $\mathbf{u} = \mathbf{v} \equiv 0.7854$, which coincides with the angle $\pi/4$ such that both players can move along the LoS. The optimal trajectories of both players are depicted in Figure 44(a) and they indeed move along the LoS.

Next we investigate the case where the external wind field is a function of both time and state. In this case, we set

$$\begin{aligned}w_1(x,y,t) &= 0.1\sin(t)x - 0.2y, \\ w_2(x,y,t) &= -0.1x + 0.4\cos(t)y.\end{aligned}$$

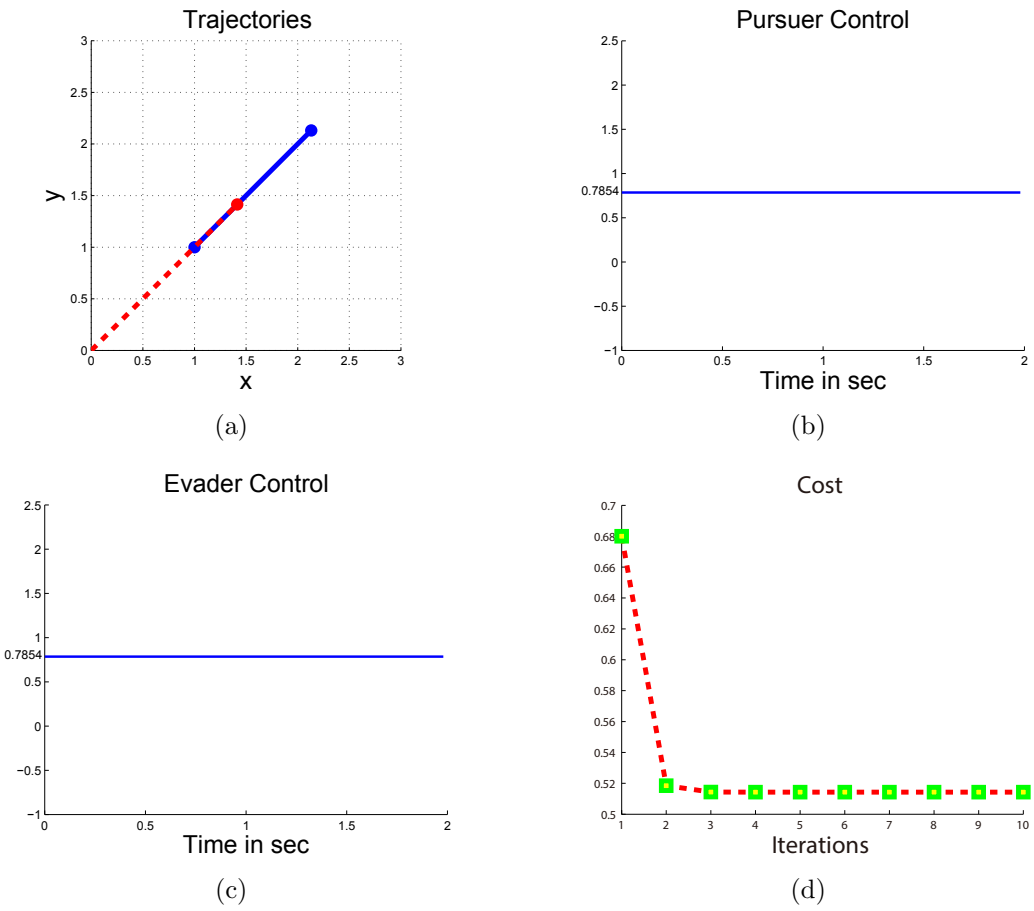


Figure 44: (a) Optimal trajectory of the pursuer in dashed red, and the evader in blue in subfigure 1. (b) Optimal controls $\mathbf{u}$ of pursuer (c) Optimal controls $\mathbf{v}$ of the evader (d) Cost per iteration.

We start with the nominal control $\bar{\mathbf{u}} \equiv 0.7$ and $\bar{\mathbf{v}} \equiv 0.7$. The final time is still fixed at $t_f = 2$. Other parameters also remain the same as in the simulation without the flow field. The optimal controls are presented in Figure 45(b) and 45(c) respectively. Optimal trajectories of the pursuer and the evader are also shown in Figures 45(a). They are different from the trajectories in Figure 44(a) due to the external flow field. Figure 45(d) depicts the cost per iteration. For the reference, we present in Figure 46 the time-varying spacial flow field at time $t = 0, 0.5, 1, 1.5$ and 2.

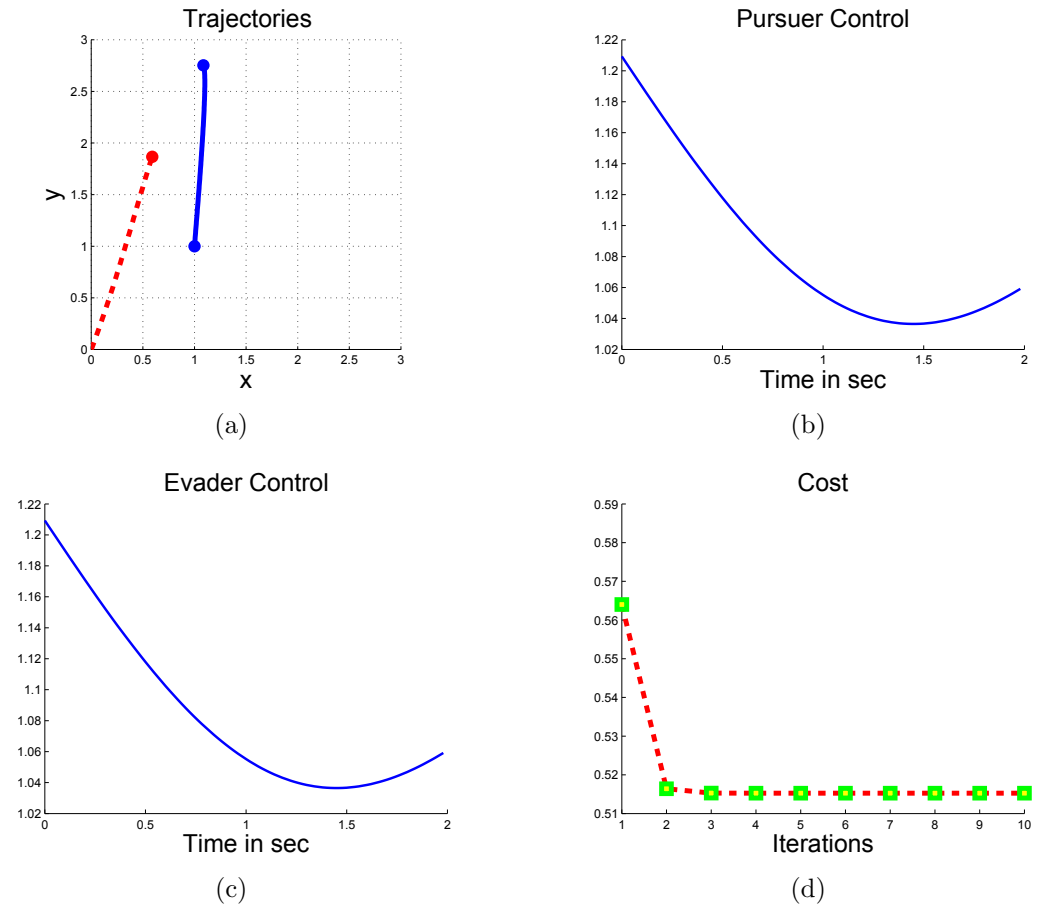
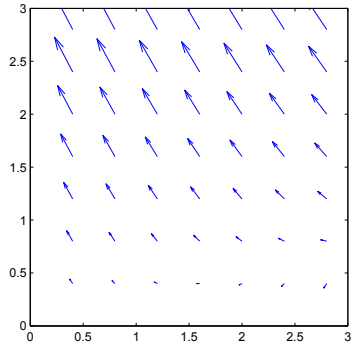
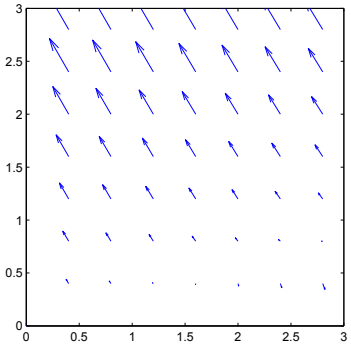


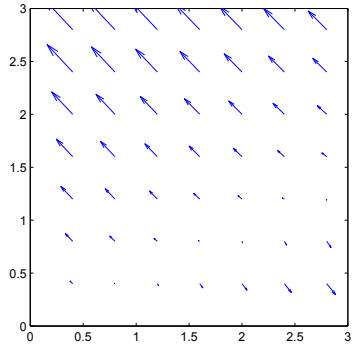
Figure 45: (a) Optimal trajectory of the pursuer in dashed red, and the evader in blue. (b) Optimal controls $\mathbf{u}$ of pursuer (c) Optimal controls $\mathbf{v}$ of the evader (d) Cost per iteration.



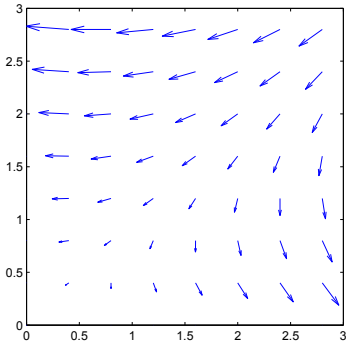
(a) t=0



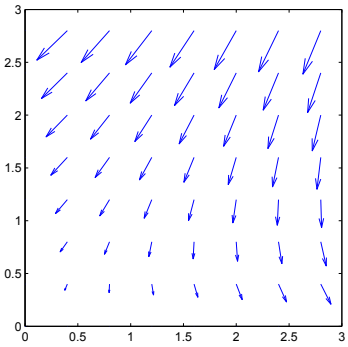
(b) t=0.5



(c) t=1.0



(d) t=1.5



(e) t=2.0

Figure 46: Time-varying spacial flow field at time $t = 0, 0.5, 1, 1.5$ and 2.

CHAPTER VIII

STOCHASTIC GAME THEORETIC CONTINUOUS TIME DIFFERENTIAL DYNAMIC PROGRAMMING

8.1 Introduction

Over the recent years, autonomy has become one of the most active areas of research, with many applications in the areas of robotics, automotive and aerospace systems. From the different computational frameworks used to achieve autonomy in engineered systems, stochastic trajectory optimization plays a key role since it provides a framework for computing the best possible action in the presence of exogenous stochastic disturbances. While there has been an extensive amount of work on stochastic and deterministic trajectory optimization, most of the prior work in this area has been on discrete time representations. In cases where the initial problem formulation is in continuous time, the previous approach is to discretize the problem formulation at hand and then perform optimization in discrete time.

In this Chapter, we derive a method for stochastic trajectory optimization using the framework of Differential Dynamic Programming (DDP) [63]. We address the problem of stochastic trajectory optimization in continuous time and present an algorithm that relies on first order expansion of the dynamics and second order expansion of the value function. In particular, we derive the equations for the backward propagation of the value function for the case of stochastic differential games. The resulting algorithm has the attractive characteristics of DDP in terms of scalability and numerical efficiency, while it also features robustness to deterministic and stochastic disturbances due to stochastic min-max formulation.

8.2 Problem Formulation

We consider the problem of a differential game between two players

$$\begin{aligned} V(\mathbf{x}(t_0), t_0) &= \min_{\mathbf{u}} \max_{\mathbf{v}} J(\mathbf{x}, \mathbf{u}, \mathbf{v}) \\ &= \min_{\mathbf{u}} \max_{\mathbf{v}} \mathbb{E}\left[\phi(\mathbf{x}(t_f)) + \int_{t_0}^{t_f} \mathcal{L}(\mathbf{x}, \mathbf{u}, \mathbf{v}) \mathrm{d}t\right], \end{aligned} \tag{174}$$

subject to the stochastic dynamics

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{v}, t)\mathrm{d}t + \mathbf{G}(\mathbf{x})\mathrm{d}w, \qquad \mathbf{x}(t_0) = \mathbf{x}_0, \tag{175}$$

where V stands for the value function (expected cost-to-go), the term J represents the performance index, and $\mathbf{x} \in \mathbb{R}^n$ represents the state of the dynamical system. The term $\mathbf{u} \in \mathbb{R}^p$ stands for the input of the minimizing player, whose objective is to minimize the performance index. Similarly, $\mathbf{v} \in \mathbb{R}^q$ represents the input of the maximizing player, which tries to maximize the performance index. The function $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^q \mapsto \mathbb{R}$ is the *running cost* and $\phi : \mathbb{R}^n \mapsto \mathbb{R}$ is the *terminal cost*, where the terminal time t_f is a prescribed constant. The term $\mathrm{d}w$ represents an increment of a m-dimensional Wiener process (standard Brownian motion), and $\mathbf{G} : \mathbb{R}^n \mapsto \mathbb{R}^{n\times m}$ is introduced to scale $\mathrm{d}w$ and match the dimension of $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^q \times \mathbb{R} \mapsto \mathbb{R}^n$. It is assumed that $\mathrm{d}w \sim \mathcal{N}(0, I_{m\times m}\mathrm{d}t)$.

Denote by $\mathcal{U}$ the admissible feedback control set of the minimizing player, that is, $\mathcal{U} = \{\mathbf{u} : [t_0, t_f] \times \mathbb{R}^n \mapsto \mathbb{R}^p, \mathbf{u}(\tau, \cdot)$ is $\mathcal{F}_\tau$-measurable, $\forall \tau \in [t, t_f]$, and $\mathbf{u}(\cdot, \mathbf{x})$ is Lebesgue measurable, $\forall \mathbf{x} \in \mathbb{R}^n\}$. Similarly, the admissible feedback control set of the maximizing player is given by $\mathcal{V} = \{\mathbf{v} : [t_0, t_f] \times \mathbb{R}^n \mapsto \mathbb{R}^q, \mathbf{v}(\tau, \cdot)$ is $\mathcal{F}_\tau$-measurable, $\forall \tau \in [t, t_f]$, and $\mathbf{v}(\cdot, \mathbf{x})$ is Lebesgue measurable, $\forall \mathbf{x} \in \mathbb{R}^n\}$. Here $\mathcal{F}_t$ denotes the corresponding filtration with respect to the Brownian motion, which can be interpreted as representing all historical information available up to time t about the stochastic

process.

We assume that the Isaacs' condition is satisfied, that is,

$$V = \min_{\mathbf{u}} \max_{\mathbf{v}} J(\mathbf{x}, \mathbf{u}, \mathbf{v}) = \max_{\mathbf{v}} \min_{\mathbf{u}} J(\mathbf{x}, \mathbf{u}, \mathbf{v}). \tag{176}$$

Next we derive the stochastic min-max DDP framework.

8.3 Optimal Control Variations

Given a nominal mean trajectory of the state and initial controls $(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}})$, and letting $\delta\mathbf{x} = \mathbf{x} - \bar{\mathbf{x}}$, $\delta\mathbf{u} = \mathbf{u} - \bar{\mathbf{u}}$, $\delta\mathbf{v} = \mathbf{v} - \bar{\mathbf{v}}$, from

$$\begin{aligned} \mathrm{d}(\bar{\mathbf{x}} + \delta\mathbf{x}) &= \mathbf{f}(\bar{\mathbf{x}} + \delta\mathbf{x}, \bar{\mathbf{u}} + \delta\mathbf{u}, \bar{\mathbf{v}} + \delta\mathbf{v})\mathrm{d}t + \mathbf{G}(\bar{\mathbf{x}} + \delta\mathbf{x})\mathrm{d}w \\ &\approx (\mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}}) + \nabla_{\mathbf{x}}\mathbf{f}\delta\mathbf{x} + \nabla_{\mathbf{u}}\mathbf{f}\delta\mathbf{u} + \nabla_{\mathbf{v}}\mathbf{f}\delta\mathbf{v})\mathrm{d}t + (\mathbf{G}(\bar{\mathbf{x}}) + \mathbf{G}_{\mathbf{x}}\delta\mathbf{x})\mathrm{d}w, \quad (177) \\ \mathrm{d}\bar{\mathbf{x}} &= \mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}})\mathrm{d}t, \quad (178) \end{aligned}$$

we obtain

$$\mathrm{d}\delta\mathbf{x} = (\nabla_{\mathbf{x}}\mathbf{f}\delta\mathbf{x} + \nabla_{\mathbf{u}}\mathbf{f}\delta\mathbf{u} + \nabla_{\mathbf{v}}\mathbf{f}\delta\mathbf{v})\mathrm{d}t + (\mathbf{G}(\bar{\mathbf{x}}) + \mathbf{G}_{\mathbf{x}}(\delta\mathbf{x}))\mathrm{d}w, \tag{179}$$

where $\mathbf{G}_{\mathbf{x}}(\delta\mathbf{x}) = [\nabla_{\mathbf{x}}\mathbf{G}^{(1)}\delta\mathbf{x}, \ldots, \nabla_{\mathbf{x}}\mathbf{G}^{(m)}\delta\mathbf{x}]$ and $\mathbf{G}^{(j)}$ denotes the j-th column vector of $\mathbf{G}$, $j = 1, \ldots, m$. The arguments of the functions in the previous derivation are omitted when they are evaluated along the nominal trajectory $(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}})$.

In order to derive the update law for the minimizing and maximizing controls, we start our analysis with Bellman/Isaac's principle, which states

$$V(\mathbf{x}_t, t) = \min_{\mathbf{u}} \max_{\mathbf{v}} \mathbb{E}\left[\int_t^{t+\mathrm{d}t} \mathcal{L}(\mathbf{x}, \mathbf{u}, \mathbf{v})\mathrm{d}t + V(\mathbf{x}_{t+\mathrm{d}t}, t + \mathrm{d}t) \middle| \mathbf{x}_t \right], \tag{180}$$

where the subscript t and $t + \mathrm{d}t$ are introduced to denote the evaluation of the variables at time t and $t + \mathrm{d}t$, respectively.

The main idea is to take expansions of the terms in both sides of equation (180) around the nominal state and control trajectories $(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}})$ to find the update equations for the minimizing control, maximizing control and backward differential equations for the zeroth, first and second order approximation terms of the value function. Starting with the left-hand side of (180), the second order expansion of the cost-to-go function around a nominal trajectory $\bar{\mathbf{x}}$ is obtained as follows

$$\begin{aligned} V(\mathbf{x}_t, t) &= V(\mathbf{x}_t + \bar{\mathbf{x}}_t - \bar{\mathbf{x}}_t, t) = V(\bar{\mathbf{x}}_t + \delta\mathbf{x}_t, t) \\ &\approx V_t + \nabla_{\mathbf{x}} V_t \delta\mathbf{x}_t + \frac{1}{2}\delta\mathbf{x}_t^{\mathsf{T}} \nabla_{\mathbf{xx}} V_t \delta\mathbf{x}_t, \end{aligned} \tag{181}$$

where $V_t = V(\bar{\mathbf{x}}_t, t)$. As for the right-hand side of (180), the first term is approximated as follows

$$\begin{aligned} \mathbb{E}\left[\int_t^{t+\mathrm{d}t} \mathcal{L}(\mathbf{x}, \mathbf{u}, \mathbf{v})\mathrm{d}t \middle| \mathbf{x}_t\right] &\approx \mathcal{L}(\mathbf{x}_t, \mathbf{u}_t, \mathbf{v}_t)\mathrm{d}t \\ &= \mathcal{L}(\bar{\mathbf{x}}_t + \delta\mathbf{x}_t, \bar{\mathbf{u}}_t + \delta\mathbf{u}_t, \bar{\mathbf{v}}_t + \delta\mathbf{v}_t)\mathrm{d}t. \end{aligned} \tag{182}$$

This expression can be approximated as

$$\begin{aligned} &\mathcal{L}\mathrm{d}t + (\nabla_{\mathbf{x}}\mathcal{L}\delta\mathbf{x} + \nabla_{\mathbf{u}}\mathcal{L}\delta\mathbf{u} + \nabla_{\mathbf{v}}\mathcal{L}\delta\mathbf{v})\mathrm{d}t \\ &+ \frac{1}{2}\begin{bmatrix} \delta\mathbf{x}_t \\ \delta\mathbf{u}_t \\ \delta\mathbf{v}_t \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \nabla_{\mathbf{xx}}\mathcal{L} & \nabla_{\mathbf{xu}}\mathcal{L} & \nabla_{\mathbf{xv}}\mathcal{L} \\ \nabla_{\mathbf{ux}}\mathcal{L} & \nabla_{\mathbf{uu}}\mathcal{L} & \nabla_{\mathbf{uv}}\mathcal{L} \\ \nabla_{\mathbf{vx}}\mathcal{L} & \nabla_{\mathbf{vu}}\mathcal{L} & \nabla_{\mathbf{vv}}\mathcal{L} \end{bmatrix} \begin{bmatrix} \delta\mathbf{x}_t \\ \delta\mathbf{u}_t \\ \delta\mathbf{v}_t \end{bmatrix} \mathrm{d}t, \end{aligned} \tag{183}$$

where the function $\mathcal{L}$ and its derivatives in the last equation are all evaluated at $(\bar{\mathbf{x}}_t, \bar{\mathbf{u}}_t, \bar{\mathbf{v}}_t)$ and thus omitted for simplicity of notation. Henceforth, all the terms are evaluated at $(\bar{\mathbf{x}}_t, \bar{\mathbf{u}}_t, \bar{\mathbf{v}}_t)$, unless specified otherwise.

Before we expand the term $\mathbb{E}\left[V(\mathbf{x}_{t+\mathrm{d}t}, t+\mathrm{d}t)\right]$ around $\bar{x}_{t+\mathrm{d}t}$ and make it compatible with the left-hand side of (180), we need to find an expression for $\delta\mathbf{x}_{t+\mathrm{d}t}$ in terms of $\delta\mathbf{x}_t$. Indeed, from (179), we get

$$\delta\mathbf{x}_{t+\mathrm{d}t} = \delta\mathbf{x}_t + (\nabla_{\mathbf{x}}\mathbf{f}\delta\mathbf{x}_t + \nabla_{\mathbf{u}}\mathbf{f}\delta\mathbf{u}_t + \nabla_{\mathbf{v}}\mathbf{f}\delta\mathbf{v}_t)\mathrm{d}t + (\mathbf{G} + \mathbf{G}_{\mathbf{x}}(\delta\mathbf{x}_t))\mathrm{d}w, \tag{184}$$

where $\mathbf{G}_{\mathbf{x}}(\delta\mathbf{x}_t) = [\nabla_{\mathbf{x}}\mathbf{G}^{(1)}\delta\mathbf{x}_t, \ldots, \nabla_{\mathbf{x}}\mathbf{G}^{(m)}\delta\mathbf{x}_t]$.

Returning to the expansion of $\mathbb{E}[V(\mathbf{x}_{t+\mathrm{d}t}, t+\mathrm{d}t)|\mathbf{x}_t]$, and letting $V_{t+\mathrm{d}t} = V(\bar{\mathbf{x}}_{t+\mathrm{d}t}, t+\mathrm{d}t)$, we have

$$\mathbb{E}\left[V(\mathbf{x}_{t+\mathrm{d}t}, t+\mathrm{d}t)\middle|\mathbf{x}_t\right] = \mathbb{E}\left[V(\bar{\mathbf{x}}_{t+\mathrm{d}t} + \delta\mathbf{x}_{t+\mathrm{d}t}, t+\mathrm{d}t)\middle|\mathbf{x}_t\right] \tag{185}$$

By expanding the last term, we obtain

$$\begin{aligned}
&\mathbb{E}\big[V_{t+\mathrm{d}t} + \nabla_{\mathbf{x}}V_{t+\mathrm{d}t}\delta\mathbf{x}_{t+\mathrm{d}t} + \frac{1}{2}\delta\mathbf{x}_{t+\mathrm{d}t}^{\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\delta\mathbf{x}_{t+\mathrm{d}t}\big|\mathbf{x}_t\big] \\
&= V_{t+\mathrm{d}t} + \nabla_{\mathbf{x}}V_{t+\mathrm{d}t}\big[\delta\mathbf{x}_t + (\nabla_{\mathbf{x}}\mathbf{f}\delta\mathbf{x}_t + \nabla_{\mathbf{u}}\mathbf{f}\delta\mathbf{u}_t + \nabla_{\mathbf{v}}\mathbf{f}\delta\mathbf{v}_t)\mathrm{d}t\big] \\
&+ \frac{1}{2}\big[\delta\mathbf{x}_t + (\nabla_{\mathbf{x}}\mathbf{f}\delta\mathbf{x}_t + \nabla_{\mathbf{u}}\mathbf{f}\delta\mathbf{u}_t + \nabla_{\mathbf{v}}\mathbf{f}\delta\mathbf{v}_t)\mathrm{d}t\big]^{\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t} \\
&\cdot \big[\delta\mathbf{x}_t + (\nabla_{\mathbf{x}}\mathbf{f}\delta\mathbf{x}_t + \nabla_{\mathbf{u}}\mathbf{f}\delta\mathbf{u}_t + \nabla_{\mathbf{v}}\mathbf{f}\delta\mathbf{v}_t)\mathrm{d}t\big] \\
&+ \frac{1}{2}\mathrm{tr}\big(\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}(\mathbf{G} + \mathbf{G}_{\mathbf{x}}(\delta\mathbf{x}_t))(\mathbf{G} + \mathbf{G}_{\mathbf{x}}(\delta\mathbf{x}_t))^{\intercal}\big)\mathrm{d}t,
\end{aligned} \tag{186}$$

where $\mathrm{tr}(\cdot)$ denotes the trace of a matrix. In the previous derivation, we make use of the fact that $\mathrm{d}w \sim \mathcal{N}(0, I_{m\times m}\mathrm{d}t)$.

We proceed by taking expansions of all the terms. First, note that we can write

$$\begin{aligned}
\mathrm{tr}\big(\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\mathbf{G}_{\mathbf{x}}(\delta\mathbf{x}_t)\mathbf{G}^{\intercal}\big) &= \mathrm{tr}\big(\mathbf{G}^{\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\mathbf{G}_{\mathbf{x}}(\delta\mathbf{x}_t)\big) \\
&= \left(\sum_{j=1}^{m}\mathbf{G}^{(j)\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\nabla_{\mathbf{x}}\mathbf{G}^{(j)}\right)\delta\mathbf{x}_t,
\end{aligned} \tag{187}$$

and similarly, the following expression follows

$$\begin{aligned}
&\mathrm{tr}\big(\nabla_{\mathbf{xx}} V_{t+\mathrm{d}t}\mathbf{G}\mathbf{G}_{\mathbf{x}}(\delta\mathbf{x}_t)^{\intercal}\big) = \bigg(\sum_{j=1}^{m}\mathbf{G}^{(j)\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\nabla_{\mathbf{x}}\mathbf{G}^{(j)}\bigg)\delta\mathbf{x}_t,\\
&\mathrm{tr}\big(\nabla_{\mathbf{xx}} V_{t+\mathrm{d}t}\mathbf{G}_{\mathbf{x}}(\delta\mathbf{x}_t)\mathbf{G}_{\mathbf{x}}(\delta\mathbf{x}_t)^{\intercal}\big) = \delta\mathbf{x}_t^{\intercal}\bigg(\sum_{j=1}^{m}\nabla_{\mathbf{x}}\mathbf{G}^{(j)\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\nabla_{\mathbf{x}}\mathbf{G}^{(j)}\bigg)\delta\mathbf{x}_t.
\end{aligned} \tag{188}$$

From (187) and (188), equation (186) can be rewritten as an explicit function of $\delta\mathbf{x}_t$, $\delta\mathbf{u}_t$ and $\delta\mathbf{v}_t$. After combining (183) with (186) and grouping the terms with respect to $\delta\mathbf{x}_t$, $\delta\mathbf{u}_t$ and $\delta\mathbf{v}_t$, we can represent the right-hand side of (180) in a compact form, that is,

$$\begin{aligned}
&\mathbb{E}\left[\int_t^{t+\mathrm{d}t}\mathcal{L}(\mathbf{x},\mathbf{u},\mathbf{v})\mathrm{d}t + V(\mathbf{x}_{t+\mathrm{d}t}, t+\mathrm{d}t)\middle|\mathbf{x}_t\right]\\
&= V_{t+\mathrm{d}t} + Q_0\mathrm{d}t + \nabla_{\mathbf{x}}V_{t+\mathrm{d}t}\delta\mathbf{x}_t + \big(Q_{\mathbf{x}}\delta\mathbf{x}_t + Q_{\mathbf{u}}\delta\mathbf{u}_t + Q_{\mathbf{v}}\delta\mathbf{v}_t\big)\mathrm{d}t\\
&+ \frac{1}{2}\delta\mathbf{x}_t^{\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\delta\mathbf{x}_t + \frac{1}{2}\begin{bmatrix}\delta\mathbf{x}_t\\ \delta\mathbf{u}_t\\ \delta\mathbf{v}_t\end{bmatrix}^{\intercal}\begin{bmatrix}Q_{\mathbf{xx}} & Q_{\mathbf{xu}} & Q_{\mathbf{xv}}\\ Q_{\mathbf{ux}} & Q_{\mathbf{uu}} & Q_{\mathbf{uv}}\\ Q_{\mathbf{vx}} & Q_{\mathbf{vu}} & Q_{\mathbf{vv}}\end{bmatrix}\begin{bmatrix}\delta\mathbf{x}_t\\ \delta\mathbf{u}_t\\ \delta\mathbf{v}_t\end{bmatrix}\mathrm{d}t,
\end{aligned} \tag{189}$$

where

$$\begin{aligned}
Q_0 &= \mathcal{L} + \frac{1}{2}\mathrm{tr}(\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\mathbf{G}\mathbf{G}^{\intercal}),\\
Q_{\mathbf{x}} &= \nabla_{\mathbf{x}}\mathcal{L} + \nabla_{\mathbf{x}}V_{t+\mathrm{d}t}\nabla_{\mathbf{x}}\mathbf{f} + \sum_{j=1}^{m}\mathbf{G}^{(j)\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\nabla_{\mathbf{x}}\mathbf{G}^{(j)},\\
Q_{\mathbf{u}} &= \nabla_{\mathbf{u}}\mathcal{L} + \nabla_{\mathbf{x}}V_{t+\mathrm{d}t}\nabla_{\mathbf{u}}\mathbf{f},\\
Q_{\mathbf{v}} &= \nabla_{\mathbf{v}}\mathcal{L} + \nabla_{\mathbf{x}}V_{t+\mathrm{d}t}\nabla_{\mathbf{v}}\mathbf{f}
\end{aligned}$$

and the second partials,

$$Q_{\mathbf{xx}} = \nabla_{\mathbf{xx}}\mathcal{L} + \nabla_{\mathbf{x}}\mathbf{f}^{\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\nabla_{\mathbf{x}}\mathbf{f}\mathrm{d}t + 2\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\nabla_{\mathbf{x}}\mathbf{f} + \sum_{j=1}^{m}\nabla_{\mathbf{x}}\mathbf{G}^{(j)\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\nabla_{\mathbf{x}}\mathbf{G}^{(j)},$$

$$Q_{\mathbf{uu}} = \nabla_{\mathbf{uu}}\mathcal{L} + \nabla_{\mathbf{u}}\mathbf{f}^{\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\nabla_{\mathbf{u}}\mathbf{f}\mathrm{d}t,$$

$$Q_{\mathbf{vv}} = \nabla_{\mathbf{vv}}\mathcal{L} + \nabla_{\mathbf{v}}\mathbf{f}^{\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\nabla_{\mathbf{v}}\mathbf{f}$$

and the mixed partials,

$$\begin{aligned}
Q_{\mathbf{ux}} &= \nabla_{\mathbf{ux}}\mathcal{L} + \nabla_{\mathbf{u}}\mathbf{f}^{\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t} + \nabla_{\mathbf{u}}\mathbf{f}^{\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\nabla_{\mathbf{x}}\mathbf{f}\mathrm{d}t,\\
Q_{\mathbf{vx}} &= \nabla_{\mathbf{vx}}\mathcal{L} + \nabla_{\mathbf{v}}\mathbf{f}^{\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t} + \nabla_{\mathbf{v}}\mathbf{f}^{\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\nabla_{\mathbf{x}}\mathbf{f}\mathrm{d}t,\\
Q_{\mathbf{uv}} &= \nabla_{\mathbf{uv}}\mathcal{L} + \nabla_{\mathbf{u}}\mathbf{f}^{\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\nabla_{\mathbf{v}}\mathbf{f}\mathrm{d}t,\\
Q_{\mathbf{xu}} &= Q_{\mathbf{ux}}^{\intercal}, \qquad Q_{\mathbf{xv}} = Q_{\mathbf{vx}}^{\intercal}, \qquad Q_{\mathbf{vu}} = Q_{\mathbf{uv}}^{\intercal}.
\end{aligned}$$

All the parameters in the previous expressions are henceforth denoted as the Q-functions. The reason we single out $V_{t+\mathrm{d}t}$, $\nabla_{\mathbf{x}}V_{t+\mathrm{d}t}\delta\mathbf{x}_t$ and $\frac{1}{2}\delta\mathbf{x}_t^{\intercal}\nabla_{\mathbf{xx}}V_{t+\mathrm{d}t}\delta\mathbf{x}_t$ instead of joining them in the Q-functions will become clear later on, as we derive the backward differential equations with respect to the value function and its derivatives.

In order to find the optimal control updates $\delta\mathbf{u}_t^*$ and $\delta\mathbf{v}_t^*$, we take the derivative of (189) with respect to $\delta\mathbf{u}_t$ and $\delta\mathbf{v}_t$, respectively, and set them equal to zero to obtain

$$\delta\mathbf{u}_t^* = -Q_{\mathbf{uu}}^{-1}\Big(Q_{\mathbf{ux}}\delta\mathbf{x} + Q_{\mathbf{uv}}\delta\mathbf{v}_t + Q_{\mathbf{u}}\Big), \tag{190}$$

$$\delta\mathbf{v}_t^* = -Q_{\mathbf{vv}}^{-1}\Big(Q_{\mathbf{vx}}\delta\mathbf{x} + Q_{\mathbf{vu}}\delta\mathbf{u}_t + Q_{\mathbf{v}}\Big). \tag{191}$$

By replacing the $\delta\mathbf{v}_t$ term in (190) with (191) and solving for $\delta\mathbf{u}_t^*$, we can eliminate $\delta\mathbf{v}_t$ in the expression of $\delta\mathbf{u}_t^*$. We can solve for $\delta\mathbf{v}_t^*$ in a similar manner and obtain

$$\delta\mathbf{u}_t^* = \mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}\delta\mathbf{x} \quad \text{and} \quad \delta\mathbf{v}_t^* = \mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{v}}\delta\mathbf{x}, \tag{192}$$

with the feed-forward gains $\mathbf{l_v}, \mathbf{l_u}$ and feedback gains $\mathbf{L_v}, \mathbf{L_u}$ defined as:

$$\mathbf{l_u} = -\big(Q_{\mathbf{uu}} - Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{vu}}\big)^{-1}\big(Q_{\mathbf{u}} - Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{v}}\big), \tag{193}$$

$$\mathbf{L_u} = -\big(Q_{\mathbf{uu}} - Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{vu}}\big)^{-1}\big(Q_{\mathbf{ux}} - Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{vx}}\big), \tag{194}$$

$$\mathbf{l_v} = -\big(Q_{\mathbf{vv}} - Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{uv}}\big)^{-1}\big(Q_{\mathbf{v}} - Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{u}}\big), \tag{195}$$

$$\mathbf{L_v} = -\big(Q_{\mathbf{vv}} - Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{uv}}\big)^{-1}\big(Q_{\mathbf{vx}} - Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{ux}}\big). \tag{196}$$

8.4 Backward Propagation of the Value Function

Notice that the feed-forward and feedback gains are functions of the value function and its first and second order partial derivatives with respect to $\mathbf{x}$. Therefore, we need to find a way to obtain these values, and this is presented in the following proposition.

Proposition 8.1 *The value function and its first and second order partial derivatives with respect to* $\mathbf{x}$ *can be determined by the backward ordinary differential equations as follows*

$$\begin{aligned}
-\frac{dV}{dt} &= Q_0 + \mathbf{l}_u^{\intercal}Q_{\boldsymbol{u}} + \mathbf{l}_v^{\intercal}Q_v + \frac{1}{2}\mathbf{l}_u^{\intercal}Q_{\boldsymbol{uu}}\mathbf{l}_{\boldsymbol{u}} + \mathbf{l}_u^{\intercal}Q_{\boldsymbol{uv}}\mathbf{l}_v + \frac{1}{2}\mathbf{l}_v^{\intercal}Q_{vv}\mathbf{l}_v,\\
-\frac{d(\nabla_{\mathbf{x}}V)}{dt} &= Q_{\mathbf{x}} + \mathbf{L}_u^{\intercal}Q_{\boldsymbol{u}} + \mathbf{L}_v^{\intercal}Q_v + Q_{\boldsymbol{ux}}^{\intercal}\mathbf{l}_{\boldsymbol{u}} + Q_{\boldsymbol{vx}}^{\intercal}\mathbf{l}_v + \mathbf{L}_u^{\intercal}Q_{\boldsymbol{uu}}\mathbf{l}_{\boldsymbol{u}} + \mathbf{L}_u^{\intercal}Q_{\boldsymbol{uv}}\mathbf{l}_v\\
&\quad + \mathbf{L}_v^{\intercal}Q_{\boldsymbol{vu}}\mathbf{l}_{\boldsymbol{u}} + \mathbf{L}_v^{\intercal}Q_{vv}\mathbf{l}_v,\\
-\frac{d(\nabla_{\mathbf{xx}}V)}{dt} &= Q_{\mathbf{xx}} + 2\mathbf{L}_u^{\intercal}Q_{\boldsymbol{ux}} + 2\mathbf{L}_v^{\intercal}Q_{\boldsymbol{vx}} + 2\mathbf{L}_v^{\intercal}Q_{\boldsymbol{vu}}\mathbf{L}_{\boldsymbol{u}} + \mathbf{L}_u^{\intercal}Q_{\boldsymbol{uu}}\mathbf{L}_{\boldsymbol{u}} + \mathbf{L}_v^{\intercal}Q_{vv}\mathbf{L}_v,
\end{aligned} \tag{197}$$

where the Q-functions are in the form

$$
\begin{aligned}
Q_0 &= \mathcal{L} + \frac{1}{2} tr(\nabla_{\mathbf{xx}} V_t \mathbf{G}\mathbf{G}^\intercal), \\
Q_{\mathbf{x}} &= \nabla_{\mathbf{x}} \mathcal{L} + \nabla_{\mathbf{x}} V_t \nabla_{\mathbf{x}} \mathbf{f} + \sum_{j=1}^{m} \mathbf{G}^{(j)\intercal} \nabla_{\mathbf{xx}} V_t \nabla_{\mathbf{x}} \mathbf{G}^{(j)}, \\
Q_{\boldsymbol{u}} &= \nabla_{\boldsymbol{u}} \mathcal{L} + \nabla_{\mathbf{x}} V_t \nabla_{\boldsymbol{u}} \mathbf{f}, \\
Q_{\boldsymbol{v}} &= \nabla_{\boldsymbol{v}} \mathcal{L} + \nabla_{\mathbf{x}} V_t \nabla_{\boldsymbol{v}} \mathbf{f}, \\
Q_{\mathbf{xx}} &= \nabla_{\mathbf{xx}} \mathcal{L} + 2\nabla_{\mathbf{xx}} V_t \nabla_{\mathbf{x}} \mathbf{f} + \sum_{j=1}^{m} \nabla_{\mathbf{x}} \mathbf{G}^{(j)\intercal} \nabla_{\mathbf{xx}} V_t \nabla_{\mathbf{x}} \mathbf{G}^{(j)}, \\
Q_{\boldsymbol{uu}} &= \nabla_{\boldsymbol{uu}} \mathcal{L}, \\
Q_{\boldsymbol{vv}} &= \nabla_{\boldsymbol{vv}} \mathcal{L}, \\
Q_{\boldsymbol{u}\mathbf{x}} &= \nabla_{\boldsymbol{u}\mathbf{x}} \mathcal{L} + \nabla_{\boldsymbol{u}} \mathbf{f}^\intercal \nabla_{\mathbf{xx}} V_t, \\
Q_{\boldsymbol{v}\mathbf{x}} &= \nabla_{\boldsymbol{v}\mathbf{x}} \mathcal{L} + \nabla_{\boldsymbol{v}} \mathbf{f}^\intercal \nabla_{\mathbf{xx}} V_t, \\
Q_{\boldsymbol{uv}} &= \nabla_{\boldsymbol{uv}} \mathcal{L}, \\
Q_{\mathbf{x}\boldsymbol{u}} &= Q_{\boldsymbol{u}\mathbf{x}}^\intercal, \qquad Q_{\mathbf{x}\boldsymbol{v}} = Q_{\boldsymbol{v}\mathbf{x}}^\intercal, \qquad Q_{\boldsymbol{v}\boldsymbol{u}} = Q_{\boldsymbol{u}\boldsymbol{v}}^\intercal,
\end{aligned} \tag{198}
$$

subject to the terminal conditions

$$
\begin{aligned}
V(t_f) = \phi(\bar{\mathbf{x}}(t_f), t_f), \quad \nabla_{\mathbf{x}} V(t_f) = \nabla_{\mathbf{x}} \phi(\bar{\mathbf{x}}(t_f), t_f), \\
\nabla_{\mathbf{xx}} V(t_f) = \nabla_{\mathbf{xx}} \phi(\bar{\mathbf{x}}(t_f), t_f).
\end{aligned} \tag{199}
$$

Proof. In order to find the update law of the value function and its first and second order partial derivatives, we need to substitute the optimal minimizing control

(190) and maximizing control (191) in the expansion of (180). To be specific, we have:

$$
\begin{aligned}
& V_t + \nabla_{\mathbf{x}} V_t \delta\mathbf{x}_t + \frac{1}{2}\delta\mathbf{x}_t^{\intercal}\nabla_{\mathbf{xx}} V_t \delta\mathbf{x}_t \\
&= V_{t+\mathrm{d}t} + Q_0 \mathrm{d}t + \nabla_{\mathbf{x}} V_{t+\mathrm{d}t}\delta\mathbf{x}_t + Q_{\mathbf{x}}\mathrm{d}t\delta\mathbf{x}_t + Q_{\mathbf{u}}\mathrm{d}t\delta\mathbf{u}_t + Q_{\mathbf{v}}\mathrm{d}t\delta\mathbf{v}_t \\
&+ \frac{1}{2}\delta\mathbf{x}_t^{\intercal}\nabla_{\mathbf{xx}} V_{t+\mathrm{d}t}\delta\mathbf{x}_t + \frac{1}{2}\begin{bmatrix} \delta\mathbf{x}_t \\ \delta\mathbf{u}_t \\ \delta\mathbf{v}_t \end{bmatrix}^{\intercal} \begin{bmatrix} Q_{\mathbf{xx}}\mathrm{d}t & Q_{\mathbf{xu}}\mathrm{d}t & Q_{\mathbf{xv}}\mathrm{d}t \\ Q_{\mathbf{ux}}\mathrm{d}t & Q_{\mathbf{uu}}\mathrm{d}t & Q_{\mathbf{uv}}\mathrm{d}t \\ Q_{\mathbf{vx}}\mathrm{d}t & Q_{\mathbf{vu}}\mathrm{d}t & Q_{\mathbf{vv}}\mathrm{d}t \end{bmatrix} \begin{bmatrix} \delta\mathbf{x}_t \\ \delta\mathbf{u}_t \\ \delta\mathbf{v}_t \end{bmatrix}
\end{aligned} \tag{200}
$$

which is equal to

$$
\begin{aligned}
& V_{t+\mathrm{d}t} + Q_0\mathrm{d}t + \nabla_{\mathbf{x}} V_{t+\mathrm{d}t}\delta\mathbf{x}_t + Q_{\mathbf{x}}\mathrm{d}t\delta\mathbf{x}_t + Q_{\mathbf{u}}\mathrm{d}t\big(\mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}\delta\mathbf{x}_t\big) + Q_{\mathbf{v}}\mathrm{d}t\big(\mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{v}}\delta\mathbf{x}_t\big) \\
& + \frac{1}{2}\delta\mathbf{x}_t^{\intercal}\nabla_{\mathbf{xx}} V_{t+\mathrm{d}t}\delta\mathbf{x}_t + \frac{1}{2}\delta\mathbf{x}_t^{\intercal}Q_{\mathbf{xx}}\mathrm{d}t\delta\mathbf{x}_t + \big(\mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}\delta\mathbf{x}_t\big)^{\intercal}Q_{\mathbf{uv}}\mathrm{d}t\big(\mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{v}}\delta\mathbf{x}_t\big) \\
& + \big(\mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}\delta\mathbf{x}_t\big)^{\intercal}Q_{\mathbf{ux}}\mathrm{d}t\delta\mathbf{x}_t + \big(\mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{v}}\delta\mathbf{x}_t\big)^{\intercal}Q_{\mathbf{vx}}\mathrm{d}t\delta\mathbf{x}_t \\
& + \frac{1}{2}\big(\mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}\delta\mathbf{x}_t\big)^{\intercal}Q_{\mathbf{uu}}\mathrm{d}t\big(\mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}\delta\mathbf{x}_t\big) + \frac{1}{2}\big(\mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{v}}\delta\mathbf{x}_t\big)^{\intercal}Q_{\mathbf{vv}}\mathrm{d}t\big(\mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{v}}\delta\mathbf{x}_t\big).
\end{aligned} \tag{201}
$$

In the previous equation we utilize the conditions $Q_{\mathbf{xu}} = Q_{\mathbf{ux}}^{\intercal}$, $Q_{\mathbf{xv}} = Q_{\mathbf{vx}}^{\intercal}$, and $Q_{\mathbf{vu}} = Q_{\mathbf{uv}}^{\intercal}$.

After grouping terms on the right-hand side of (201) as zeroth order, first order and second order expressions of $\delta\mathbf{x}_t$, we can equate the coefficients on the left-hand side and right-hand side of (201) and get

$$
\begin{aligned}
V_t &= V_{t+\mathrm{d}t} + Q_0\mathrm{d}t + \mathbf{l}_{\mathbf{u}}^{\intercal}Q_{\mathbf{u}}\mathrm{d}t + \mathbf{l}_{\mathbf{v}}^{\intercal}Q_{\mathbf{v}}\mathrm{d}t + \frac{1}{2}\mathbf{l}_{\mathbf{u}}Q_{\mathbf{uu}}\mathbf{l}_{\mathbf{u}}\mathrm{d}t + \mathbf{l}_{\mathbf{u}}^{\intercal}Q_{\mathbf{uv}}\mathbf{l}_{\mathbf{v}}\mathrm{d}t + \frac{1}{2}\mathbf{l}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vv}}\mathbf{l}_{\mathbf{v}}\mathrm{d}t, \\
\nabla_{\mathbf{x}} V_t &= \nabla_{\mathbf{x}} V_{t+\mathrm{d}t} + Q_{\mathbf{x}}\mathrm{d}t + \mathbf{L}_{\mathbf{u}}^{\intercal}Q_{\mathbf{u}}\mathrm{d}t + \mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{v}}\mathrm{d}t + Q_{\mathbf{ux}}^{\intercal}\mathbf{l}_{\mathbf{u}}\mathrm{d}t + Q_{\mathbf{vx}}^{\intercal}\mathbf{l}_{\mathbf{v}}\mathrm{d}t \\
&\quad + \mathbf{L}_{\mathbf{u}}^{\intercal}Q_{\mathbf{uu}}\mathbf{l}_{\mathbf{u}}\mathrm{d}t + \mathbf{L}_{\mathbf{u}}^{\intercal}Q_{\mathbf{uv}}\mathbf{l}_{\mathbf{v}}\mathrm{d}t + \mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vu}}\mathbf{l}_{\mathbf{u}}\mathrm{d}t + \mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vv}}\mathbf{l}_{\mathbf{v}}\mathrm{d}t, \\
\nabla_{\mathbf{xx}} V_t &= \nabla_{\mathbf{xx}} V_{t+\mathrm{d}t} + Q_{\mathbf{xx}}\mathrm{d}t + 2\mathbf{L}_{\mathbf{u}}^{\intercal}Q_{\mathbf{ux}}\mathrm{d}t + 2\mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vx}}\mathrm{d}t + 2\mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vu}}\mathbf{L}_{\mathbf{u}}\mathrm{d}t \\
&\quad + \mathbf{L}_{\mathbf{u}}^{\intercal}Q_{\mathbf{uu}}\mathbf{L}_{\mathbf{u}}\mathrm{d}t + \mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vv}}\mathbf{L}_{\mathbf{v}}\mathrm{d}t.
\end{aligned} \tag{202}
$$

By moving $V_{t+\mathrm{d}t}$, $\nabla_{\mathbf{x}} V_{t+\mathrm{d}t}$ and $\nabla_{\mathbf{xx}} V_{t+\mathrm{d}t}$ to the left-hand side of (202), and dividing both sides with $\mathrm{d}t$, (202) can be rewritten as

$$-\frac{\mathrm{d}V_t}{\mathrm{d}t} = Q_0 + \mathbf{l}_{\mathbf{u}}^{\intercal} Q_{\mathbf{u}} + \mathbf{l}_{\mathbf{v}}^{\intercal} Q_{\mathbf{v}} + \frac{1}{2}\mathbf{l}_{\mathbf{u}} Q_{\mathbf{uu}} \mathbf{l}_{\mathbf{u}} + \mathbf{l}_{\mathbf{u}}^{\intercal} Q_{\mathbf{uv}} \mathbf{l}_{\mathbf{v}} + \frac{1}{2}\mathbf{l}_{\mathbf{v}}^{\intercal} Q_{\mathbf{vv}} \mathbf{l}_{\mathbf{v}}, \tag{203}$$

$$\begin{aligned} -\frac{\mathrm{d}\nabla_{\mathbf{x}} V_t}{\mathrm{d}t} &= Q_{\mathbf{x}} + \mathbf{L}_{\mathbf{u}}^{\intercal} Q_{\mathbf{u}} + \mathbf{L}_{\mathbf{v}}^{\intercal} Q_{\mathbf{v}} + Q_{\mathbf{ux}}^{\intercal} \mathbf{l}_{\mathbf{u}} + Q_{\mathbf{vx}}^{\intercal} \mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{u}}^{\intercal} Q_{\mathbf{uu}} \mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}^{\intercal} Q_{\mathbf{uv}} \mathbf{l}_{\mathbf{v}} \\ &\quad + \mathbf{L}_{\mathbf{v}}^{\intercal} Q_{\mathbf{vu}} \mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{v}}^{\intercal} Q_{\mathbf{vv}} \mathbf{l}_{\mathbf{v}}, \end{aligned} \tag{204}$$

$$-\frac{\mathrm{d}\nabla_{\mathbf{xx}} V_t}{\mathrm{d}t} = Q_{\mathbf{xx}} + 2\mathbf{L}_{\mathbf{u}}^{\intercal} Q_{\mathbf{ux}} + 2\mathbf{L}_{\mathbf{v}}^{\intercal} Q_{\mathbf{vx}} + 2\mathbf{L}_{\mathbf{v}}^{\intercal} Q_{\mathbf{vu}} \mathbf{L}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}^{\intercal} Q_{\mathbf{uu}} \mathbf{L}_{\mathbf{u}} + \mathbf{L}_{\mathbf{v}}^{\intercal} Q_{\mathbf{vv}} \mathbf{L}_{\mathbf{v}}. \tag{205}$$

Letting $\mathrm{d}t$ approach 0 in (203) through (205), we readily obtain (197). Similarly, the expressions of the Q-functions are turned into (198).

At the final time, we have $V(\mathbf{x}(t_f), t_f) = \phi(\mathbf{x}(t_f))$. By taking the expansions around $\bar{\mathbf{x}}(t_f)$ we get

$$\begin{aligned} \phi(\mathbf{x}(t_f)) &= \phi(\bar{\mathbf{x}}(t_f) + \delta\mathbf{x}(t_f)) \\ &\approx \phi(\bar{\mathbf{x}}(t_f)) + \nabla_{\mathbf{x}} \phi(\bar{\mathbf{x}}(t_f)) \delta\mathbf{x}(t_f) + \delta\mathbf{x}(t_f)_{\intercal} \nabla_{\mathbf{xx}} \phi(\bar{\mathbf{x}}(t_f)) \delta\mathbf{x}(t_f). \end{aligned} \tag{206}$$

Therefore, the boundary conditions at $t = t_f$ for the backward differential equations are represented by (199), and this completes the proof. □

Now that we have found a method to obtain the value function and its first and second order partial derivatives with respect to the state through backward propagation, we put all the pieces together and provide the Stochastic Game Theoretic Differential Dynamic Programming (SGT-DDP) algorithm in a pseudocode form shown in Algorithm 3.

The cost function is chosen depending on the application. The roles of minimizing and maximizing controls in the control design are determined by the choices of the Hessian of $\mathcal{L}$ with respect to the controls. In order to see this feature, recall that $Q_{\mathbf{uu}} = \nabla_{\mathbf{uu}} \mathcal{L}$ and $Q_{\mathbf{vv}} = \nabla_{\mathbf{vv}} \mathcal{L}$. Furthermore, since $\nabla_{\mathbf{uu}} \mathcal{L}$ and $\nabla_{\mathbf{vv}} \mathcal{L}$ are design parameters,

Algorithm 3 Pseudocode of the SGT-DDP Algorithm

Given:

- Stochastic dynamics $\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{v}, t)\mathrm{d}t + \mathbf{G}(\mathbf{x})\mathrm{d}w$
- Initial condition of the dynamics $\mathbf{x}_0$
- Initial minimizing control $\bar{\mathbf{u}}$ and maximizing control $\bar{\mathbf{v}}$
- Terminal time t_f
- Multiplier γ
- A constant N

procedure UPDATE_CONTROL($\mathbf{x}_0$, $\bar{\mathbf{u}}$, $\bar{\mathbf{v}}$, t_f, γ, N)
  for i counting from 1 to N **do**
    Get the initial mean trajectory $\bar{\mathbf{x}}$ by integrating the deterministic part of the controlled dynamics forward with $\mathbf{x}_0$, $\bar{\mathbf{u}}$ and $\bar{\mathbf{v}}$;
    Find the value of $V, V_{\mathbf{x}}, V_{\mathbf{xx}}$ at t_f according to (199);
    Compute the quadratic approximation of the value function $V, V_{\mathbf{x}}, V_{\mathbf{xx}}$ in $[0, t_f]$ by integrating backward the equations (197);
    Compute $\mathbf{l}_{\mathbf{u}}, \mathbf{L}_{\mathbf{u}}, \mathbf{l}_{\mathbf{v}}, \mathbf{L}_{\mathbf{v}}$ according to equations (193) through (196) with the Q-functions as computed in (198);
    Get $\delta\mathbf{x}(t)$ through $\delta\mathbf{x}_{t+\mathrm{d}t} = \delta\mathbf{x}_t + (\nabla_{\mathbf{x}}\mathbf{f}\delta\mathbf{x}_t + \nabla_{\mathbf{u}}\mathbf{f}\delta\mathbf{u}_t + \nabla_{\mathbf{v}}\mathbf{f}\delta\mathbf{v}_t)\mathrm{d}t$ while replacing $\delta\mathbf{u}$ and $\delta\mathbf{v}$ with $(\mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}\delta\mathbf{x})$ and $(\mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{v}}\delta\mathbf{x})$, respectively;
    Compute $\delta\mathbf{u} = \mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}\delta\mathbf{x}$ and $\delta\mathbf{v} = \mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{v}}\delta\mathbf{x}$;
    Update control $\mathbf{u}^* = \mathbf{u}^* + \gamma\delta\mathbf{u}$, where $\gamma \in [0, 1]$ is chosen as the learning rate;
    Set $\bar{\mathbf{u}} = \mathbf{u}^*$ and $\bar{\mathbf{v}} = \bar{\mathbf{v}}^*$;
  end for
  return $\mathbf{x}^*$, $\mathbf{u}^*$, $\mathbf{v}^*$, $\mathbf{l}_{\mathbf{u}}, \mathbf{L}_{\mathbf{u}}, \mathbf{l}_{\mathbf{v}}, \mathbf{L}_{\mathbf{v}}$.
end procedure

they can be chosen such that $\nabla_{\mathbf{uu}}\mathcal{L}$ is positive definite and $\nabla_{\mathbf{vv}}\mathcal{L}$ is negative definite. Such design makes sure that the role of the controller $\mathbf{u}$ is to minimize the cost whereas the controller $\mathbf{v}$ aims to maximize it. Since $Q_{\mathbf{uu}} > 0$ and $Q_{\mathbf{vv}} < 0$, we can deduce that $\left(Q_{\mathbf{uu}} - Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{vu}}\right)^{-1} > 0$, and $\left(Q_{\mathbf{vv}} - Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{uv}}\right)^{-1} < 0$. Combining these two matrix inequalities and the form of the feed-forward and feedback gains of the control policies in (193) through (196), it can be seen that the controls are updated such that the control $\mathbf{u}$ tends to reduce the cost while the control $\mathbf{v}$ tends to increase it.

8.5 Simulation Results

In this section, we apply the proposed SGT-DDP algorithm to three systems. The first system is a one-dimensional system, the second one is the inverted pendulum and the third one is the cart pole problem. More precisely, the first system is governed by an equation of the form

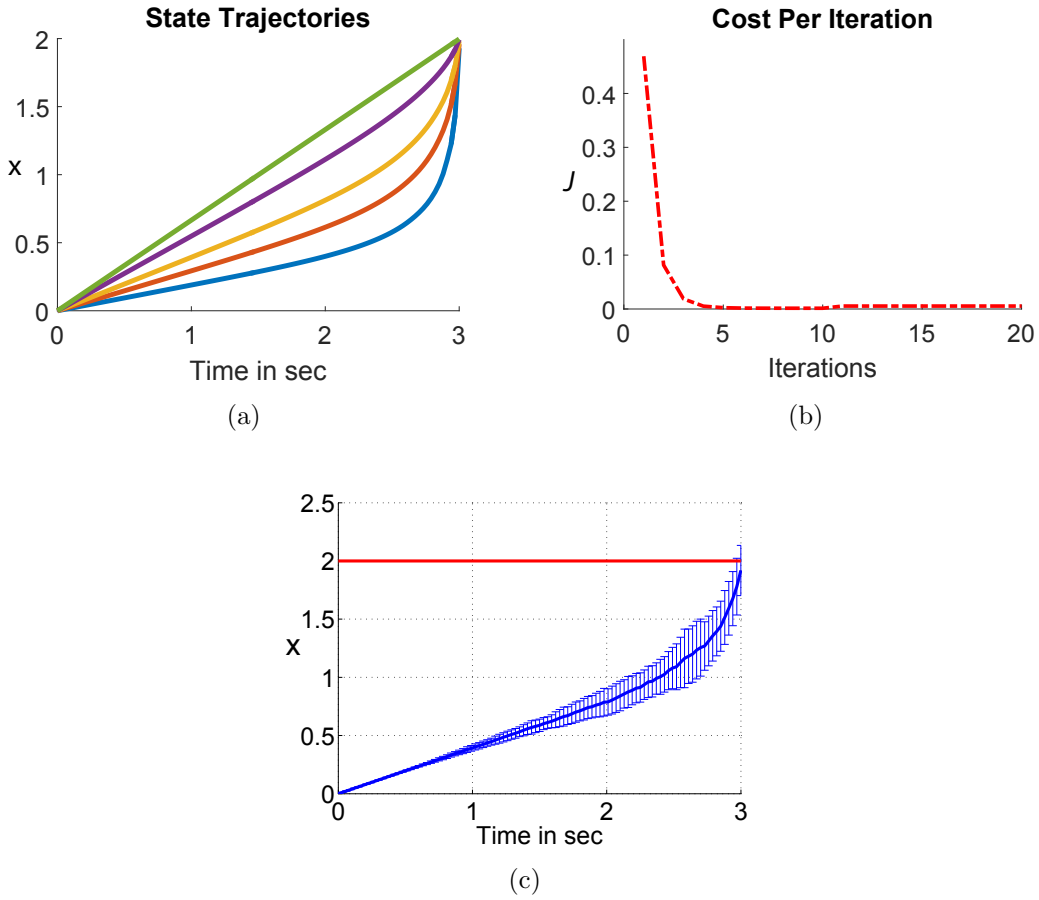
$$\mathrm{d}\mathbf{x} = a\mathbf{x}^2\mathrm{d}t + \mathbf{u}\mathrm{d}t + \mathbf{v}\mathrm{d}t + b\mathbf{x}^2\mathrm{d}w. \tag{207}$$

Our task is to bring the state from $\mathbf{x}_0 = 0$ to the target position $\mathbf{x}_f = 2$ and the cost function can be expressed as

$$V(\mathbf{x}(t_0), t_0) = \min_{\mathbf{u}} \max_{\mathbf{v}} \mathbb{E}\left[\phi + \int_{t_0}^{t_f} (R_{\mathbf{u}}\mathbf{u}^2 - R_{\mathbf{v}}\mathbf{v}^2)\mathrm{d}t\right],$$

where $\phi = (\mathbf{x}(t_f) - \mathbf{x}_f)^2$. The value of the tuning parameters are set as $a = 0.005$, $R_{\mathbf{u}} = 10^{-3}$ and $R_{\mathbf{v}} = 10^{-2}$. We vary the value of b to simulate different intensities of the noise. The left plot in Fig. 47 illustrates the state space trajectories for different values of the variance of the noise while the right plot portraits the convergence behavior of the algorithm. In Fig. 47(a), instead of running the controlled system many times for different realizations of noise and then calculate the mean, we simply

zero the noise term and run the system only once. The mean of 100 trajectories of the state with feedback minimizing and maximizing controls under stochastic disturbance where $b = 0.4$ is depicted in Fig. 47(c). An error bar is also drawn at every time step. Each error bar has a distance of the standard deviation among 100 trajectories at that time step above and below the curve.



(a) (b)

(c)

Figure 47: (a) State trajectories for noise with different value of variance. Green, purple, yellowm orange and light blue curves corresponds to cases where $b = 0, 0.2, 0.4, 0.6, 1$, respectively. (b) Cost per iteration under one of the noise profiles. (c) State trajectories with feedback minimizing and maximizing controls under stochastic disturbance in blue, red line represents the goal position.

In the second system we use the inverted pendulum, the dynamics of which is

given by

$$\mathrm{d}\mathbf{x} = \begin{bmatrix} \mathbf{x}(2) \\ (mg\ell/I)\sin\mathbf{x}(1) - (b/I)\mathbf{x}(2) + (1/I)(\mathbf{u}+\mathbf{v}) \end{bmatrix} \mathrm{d}t + \begin{bmatrix} 0 \\ \alpha\mathbf{x}(1) \end{bmatrix} \mathrm{d}w, \tag{208}$$

where $\mathbf{x} = [\theta, \dot{\theta}]^\intercal$ and the parameters are chosen as $m = 1$ Kg, $\ell = 0.5$ m, $b = 0.1$, $I = ml^2$, $g = 9.81\mathrm{Kg}\cdot\mathrm{m}/\sec^2$ and $\alpha = 1$. Our goal is to bring the pendulum from the initial state $[\theta, \dot{\theta}] = [\pi, 0]$ to the target position $[\theta, \dot{\theta}] = [0, 0]$. The cost function is given by

$$J = \mathbf{x}(t_f)^\intercal Q_f \mathbf{x}(t_f) + \int_0^{t_f} (\mathbf{u}^\intercal R_{\mathbf{u}} \mathbf{u} - \mathbf{v}^\intercal R_{\mathbf{v}} \mathbf{v})\, \mathrm{d}t, \tag{209}$$

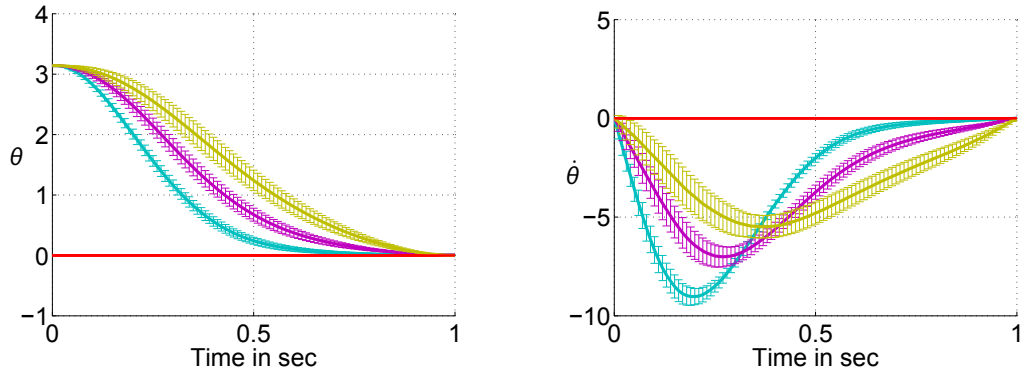
where

$$Q_f = \begin{bmatrix} 100, & 0 \\ 0, & 5 \end{bmatrix}. \tag{210}$$

For the simulation, we set $R_{\mathbf{u}} = 0.1$ and $R_{\mathbf{v}} = 0.12, 0.2, 1$ to observe how the change of control authority of the maximizing control affects the outcome of the simulation. Notice here that in all the cases, we set $R_{\mathbf{u}} < R_{\mathbf{v}}$, which indicates that the minimizing control is penalized less than the maximizing control. In other words, the minimizing control $\mathbf{u}$ has more control authority than the maximizing control $\mathbf{v}$ and thus the minimizing control should be able to bring the expected trajectory to the goal state despite the best effort of the maximizing control. Such phenomenon can be shown in the follow-up simulation.

We set the initial control to be $\bar{\mathbf{u}} \equiv 0$, $\bar{\mathbf{v}} \equiv 0$, the terminal time to be $t_f = 1$ and the multiplier $\gamma = 0.8$. For each value of $R_{\mathbf{v}}$, we run the inverted pendulum system with feedback minimizing control for 1000 times. In Fig. 48(a), we have three colored plots, where cyan, magenta and dark yellow plots correspond to the case of $R_{\mathbf{v}} = 0.13, 0.2$ and 10, respectively. The plot of each color contains the mean of the trajectories of θ with respect to time and an error bar with a distance of the standard

deviation above and below the curve is drawn at every time step. Similarly,, the mean and standard deviation of the trajectories of $\dot{\theta}$ for these values of $R_{\mathbf{v}}$ are shown in Fig. 48(b).



(a) (b)

Figure 48: (a) Plots of mean and standard deviation of 1000 trajectories of θ. Cyan, magenta and yellow plots correspond to the case of $R_{\mathbf{v}} = 0.13, 0.2$ and 10, respectively. (b) Plots of mean and standard deviation of 1000 trajectories of $\dot{\theta}$. Cyan, magenta and yellow plots correspond to the case of $R_{\mathbf{v}} = 0.13, 0.2$ and 10, respectively.

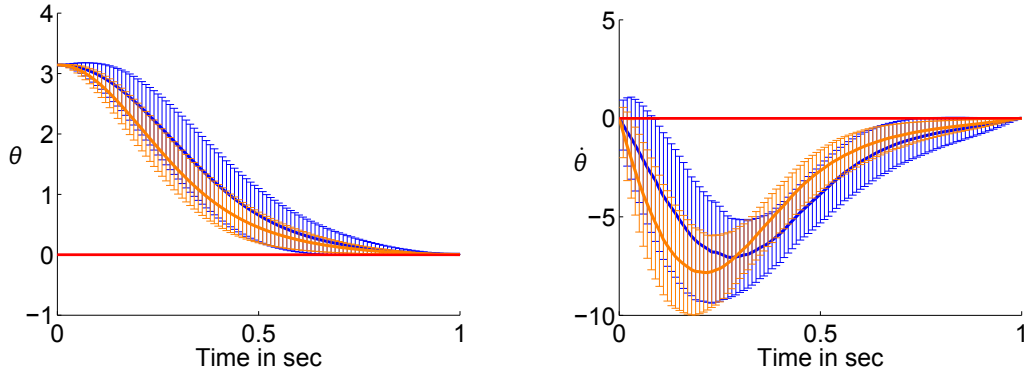
It can be observed from Fig. 48 that the feed-forward and feedback parts of the control policy alters as $R_{\mathbf{v}}$ changes. In particular, as $R_{\mathbf{v}}$ decreases, the feed-forward control steers the mean trajectory towards the desired state earlier. Moreover, the optimal feedback gains reduce the variability of the trajectories when $R_{\mathbf{v}}$ gets small. It indicates that the game theoretic formulation can give rise to robust policies that shape both the mean and variance of optimal trajectories.

Furthermore, we compare the performance of the feedback control emerged from our algorithm with the control that comes from the deterministic game theoretic DDP in [139]. This time, we fix $R_{\mathbf{v}} = 1$ and $\alpha = 4$. The other parameters remain unchanged. The result is shown in Fig. 49, where the orange plots depict the mean and error bar of the state trajectories subject to the feedback control originated from the algorithm in this section, and the blue plots are associated with the deterministic game theoretic DDP. The SGT-DDP algorithm returns a control that drives the

mean trajectory towards the desired state earlier. One explanation of this behavior is presented as follows. Recall from (197) that

$$-\frac{\mathrm{d}(\nabla_{\mathbf{xx}} V)}{\mathrm{d}t} = Q_{\mathbf{xx}} + 2\mathbf{L}_{\mathbf{u}}^{\intercal} Q_{\mathbf{ux}} + 2\mathbf{L}_{\mathbf{v}}^{\intercal} Q_{\mathbf{vx}} + 2\mathbf{L}_{\mathbf{v}}^{\intercal} Q_{\mathbf{vu}} \mathbf{L}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}^{\intercal} Q_{\mathbf{uu}} \mathbf{L}_{\mathbf{u}} + \mathbf{L}_{\mathbf{v}}^{\intercal} Q_{\mathbf{vv}} \mathbf{L}_{\mathbf{v}}. \tag{211}$$

Let $Q_{\mathbf{vv}} = \nabla_{\mathbf{vv}} \mathcal{L}$ be negative definite. Then the more authority the maximizing control has (the smaller $Q_{\mathbf{vv}}$ is), the larger the right-hand side of (211) becomes. Similarly, by the expression of $Q_{\mathbf{xx}}$, as the state dependent noise gets larger, the right-hand side of (211) also increases. Therefore, the noise and the maximizer affects to the update of $\nabla_{\mathbf{xx}} V$ in a similar fashion. Hence, it is expected that the enhancement of the control authority of the maximizing control and the inclusion of noise in SGT-DDP result in similar behavior, as we have shown in Figs. 48 and 49.



(a) (b)

Figure 49: (a) Comparison of plots of mean and standard deviation of 1000 trajectories of θ with respect to the SGT-DDP and the GT-DDP control in orange and blue, respectively. (b) Comparison of plots of mean and standard deviation of 1000 trajectories of $\dot{\theta}$ with respect to the SGT-DDP and the GT-DDP control in orange and blue, respectively.

In the next example, we consider the cart pole problem with conflicting controls under stochastic disturbances. This is an underactuated mechanical system and the

corresponding dynamics is given by

$$\dot{\mathbf{x}} = f(\mathbf{x}) + G(\mathbf{x})(\mathbf{u} + \mathbf{v} + \mathrm{d}w), \tag{212}$$

where

$$f(\mathbf{x}) = \begin{bmatrix} \mathbf{x}(2) \\ \dfrac{m \sin \mathbf{x}(3)(-\ell \dot{\mathbf{x}(4)}^2 + g \cos \mathbf{x}(3))}{M + m \sin^2 \mathbf{x}(3)} \\ \mathbf{x}(4) \\ \dfrac{-m\ell \mathbf{x}(4)^2 \cos \mathbf{x}(3) \sin \mathbf{x}(3) + (M+m) g \sin \mathbf{x}(3)}{\ell(M + m \sin^2 \mathbf{x}(3))} \end{bmatrix}, \tag{213}$$

and

$$G(\mathbf{x}) = \begin{bmatrix} 0 \\ \dfrac{1}{M + m \sin(\mathbf{x}(3))^2} \\ 0 \\ \dfrac{\cos(\mathbf{x}(3))}{\ell(M + m \sin(\mathbf{x}(3))^2)} \end{bmatrix}, \tag{214}$$

The state $\mathbf{x} = [x, \dot{x}, \theta, \dot{\theta}]_\intercal$ where x represents the displacement of the cart and θ stands for the angle of the pole. $\ell = 0.5$ is the length of the pole, $M = 10$ is the mass of the cart and $m = 1$ is the mass of the pole, and $g = 9.8$ is the gravitational constant. The cost function is in the form

$$J = (\mathbf{x}(t_f) - x_f)^\intercal Q_f (\mathbf{x}(t_f) - x_f) + \int_0^{t_f} (\mathbf{u}^\intercal R_{\mathbf{u}} \mathbf{u} - \mathbf{v}^\intercal R_{\mathbf{v}} \mathbf{v}) \mathrm{d}t, \tag{215}$$

where $Q_f = \mathrm{diag}([0, 500, 5000, 50])$. The other parameters in the cost function are given by $R_{\mathbf{u}} = 0.01$, $R_{\mathbf{v}} = 0.1$. The minimizing control $\mathbf{u}$ aims to bring the system from the initial state $\mathbf{x}_0 = [0, 0, \pi, 0]^\intercal$ to the desired state $x_f = [0, 0, 0, 0]^\intercal$, whereas the maximizing control $\mathbf{v}$ attempts to stop this from happening. Note that the terminal displacement is actually not restricted to reach 0 since $Q_f(1, 1) = 0$ when we design

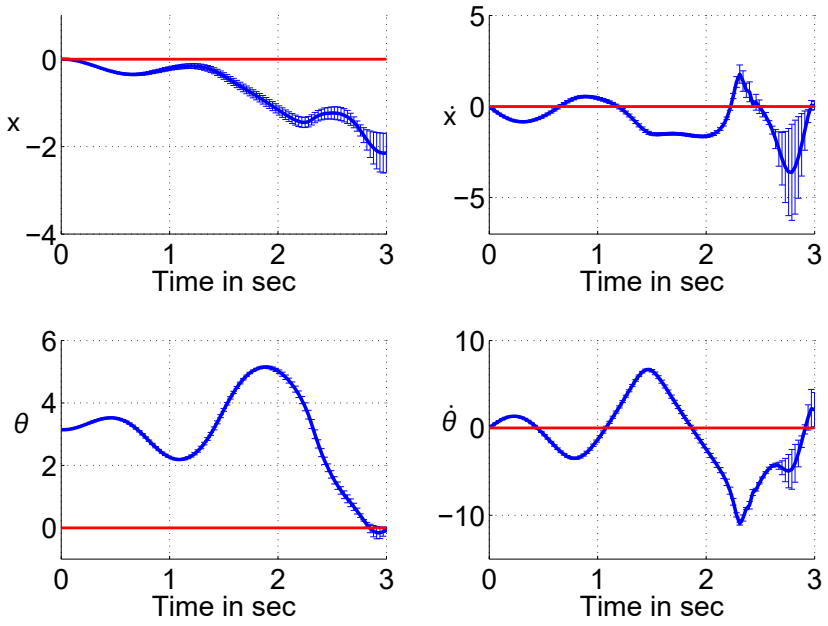


Figure 50: Mean and standard variance of 100 trajectories of the four states with respect to time under conflicting controls in blue. The red lines represents the goal states $\theta = 0$.

the cost function. The initial controls are set to be $\bar{\mathbf{u}} \equiv 0$, $\bar{\mathbf{v}} \equiv 0$, the terminal time $t_f = 3$ and the multiplier $\gamma = 0.3$. The mean of 100 trajectories of the states under conflicting feedback controls and stochastic disturbances are depicted in Fig. 50 in blue and error bars of the standard deviation are drawn around the mean trajectories.

CHAPTER IX

CONCLUSIONS AND FUTURE WORK

9.1 Conclusions

In a differential game setting, decisions made by the players as well as the outcome of the game are susceptible to uncertainties in the game. The uncertainties that enter the system can be divided into three main categories, namely, external/environmental uncertainties, endogenous/dynamical uncertainties and observation uncertainties. Due to the pervasive nature of uncertainties in realistic differential game scenarios, it has become a pressing task to find various methods to deal with differential games under uncertainties. In this dissertation, we present analytical and numerical methods to address pursuit-evasion games under different forms of environmental uncertainties. The area of pursuit-evasion games is an important special case of general differential games and the methods introduced in pursuit-evasion games could also be generalized to other fields of differential games.

In Chapter 2, under the assumption that at most one pursuer is actively chasing a moving target at every instant of time, we have proposed a target-pursuer assignment strategy to capture several moving targets by a set of pursuers in a wind field. We take advantage of the fact that the problem of assigning a pursuer to the moving target can be associated with a dynamically changing Zermelo-Voronoi partitioning problem. This partition assigns to each pursuer the points that can be intercepted faster than any other pursuer, using the minimum-time Zermelo's navigation law. We utilize the Zermelo-Voronoi diagram (ZVD) to dynamically assign the active pursuers at each instant of time.

In Chapter 3, we deal with a differential game between two players in a plane

subject to a linear spatial flow field. Under the assumption that the flow field is approximated by a time-invariant affine function, we reformulate the problem as a game of kind and characterize the initial conditions that secure capture of the evader, as well as the initial conditions that lead to escape of the evader, when both players act optimally. The optimal controls of both players inside the capture zone are derived, and numerical simulations with different parameters of the flow field are presented to illustrate the corresponding capture and escape zones.

A differential game between an evader and multiple pursuers in a general spatiotemporal flow field is discussed in Chapter 4. It is shown that the game terminates when the usable reachable set of the evader becomes the empty set for the first time. A simplified condition for capture of the evader can be derived when the maximum speed of the evader is less than the maximum speed of each pursuer. The level set method is adopted to compute and propagate the reachable sets of all the players. Depending on whether a pursuer contributes to the outcome of the game, whether it chases the evader directly, or whether it guards some part of the reachable set of the evader so that the evader does not detour from its optimal trajectory, the pursuers can be respectively classified into redundant pursuers, active pursuers, or guards. The optimal trajectories and controls of the pursuers and the evader are retrieved by backward propagation along the corresponding levels of the reachable sets. The proposed solution scheme is demonstrated by applying it to multi-player, pursuit-evasion games taking place in realistic strong and time-dependent external flow fields, including a case with an obstacle.

The scheme presented in Chapter 4 is extended in Chapter 5 to address pursuit-evasion games between multiple pursuers and evaders in a 3D space setting. The problem is divided into a pursuer assignment problem, where a partition of the pursuer set is formed to pair the pursuers with the evaders, and the subsequent sub-game between each evader and its assigned pursuers. The reachable set analysis is utilized

for both problems and the level set method is adopted to propagate the reachable sets of all the players. The pursuer assignment problem is achieved by pairing each pursuer with the evader that first enters the reachable set of this pursuer among all the evaders. On the other hand, the sub-game between multiple pursuers and one evader terminates when the usable reachable set of the evader becomes the empty set for the first time. The optimal trajectories and controls are retrieved through backward propagation of the reachable sets. The method is applied to multiplayer pursuit-evasion games in an artificial flow field and a realistic wind field.

In Chapter 6, a differential game between a pursuer and an evader in an external stochastic flow field is considered. A moment decomposition method is utilized to divide the state into a deterministic part and a random fluctuation. Then the mean and augmented reachable sets are introduced to help the analysis of the problem through a reachability-based method. It is shown that the expected optimal time-to-capture can be found as the first time when the augmented reachable set of the pursuer fully contains that of the evader. The level set method is adopted to generate the mean reachable sets of both players, and the augmented reachable sets can be subsequently obtained through an extension method with error ellipses. Our scheme is applied to a pursuit-evasion game in a stochastic flow field.

Most of the system models we are dealing with nowadays, whether the model of an aircraft or a robot, are only approximate models and are thus affected by the dynamical uncertainties. As part of this thesis, we also propose algorithms whose roots can be found in the DDP method in order to deal with some differential games under dynamical uncertainties. In Chapter 7, we consider a differential game problem involving two conflicting controls. By taking a Taylor series expansion of the HJBI equation around a nominal trajectory, we find the update law of both controls/players, as well as the backward propagation equations of the zeroth, first and second order approximation terms of the value function. The resulting GT-DDP algorithm, is

derived by using expansion of the dynamics in continuous time. We test the GT-DDP on two distinct systems: one is the inverted pendulum with conflicting controls where both controls are affine in dynamics and quadratic in running cost, the other is the pursuit-evasion game subject to an external flow field, where both controls enter in trigonometric form in the dynamics as well as the running cost. Finally, we utilize the optimal gains we get from the algorithm to guide an inverted pendulum subject to noise to its desired state and demonstrate the effect of the design of the cost function to the feed-forward and feedback parts of the control policies.

In Chapter 8, we address a differential game problem with two conflicting controls under stochastic dynamics, where the dynamical uncertainty is modeled as a state-dependent Gaussian noise. Starting from the Bellman-Isaacs equation, we take expansions of the value function and its derivatives around a nominal trajectory and find the update law of the minimizing and maximizing controls of both players, as well as the backward differential equations of the approximation of the value function up to the second order. We present the SGT-DDP algorithm and analyze the effect of the game theoretic formulation in the feed-forward and feedback parts of the control policies. The SGT-DDP algorithm is tested on three distinct systems: one is a first-order nonlinear system and the other two are the inverted pendulum and cart pole problems with conflicting controls. We investigate how the intensity of the stochastic noise affects the behavior of the controls and the corresponding trajectories.

Next, we highlight some potential directions for future research that build upon some of the results presented in this dissertation.

9.2 Projected Reachable Set Approach in Pursuit-Evasion Games

Up until now, we have only applied the reachability-based approach to pursuit-evasion games with simple kinematics where the control input is the velocity directly. However, this approach is not limited to such cases. When the dimension of the state

space is larger than that of the *maneuver space*, defined as the 2-dimensional or 3-dimensional Euclidean space in which the players moves, and the termination of the problem is determined by the relative positions between the pursuer and the evader, then we can project the reachable sets of the players in the maneuver space. Capture is guaranteed when the projected reachable set of the evader at some time is fully covered by the projected reachable set of the pursuer.

The projected reachable sets has been utilized in [100] to solve collision avoidance problems, where a pursuit-evasion game involving two identical vehicles moving in the plane is considered. The dynamics of the system is given by $\dot{X} = f(X, u, v)$, where $X \in \mathbb{R}^3$ includes two relative planar locations and one relative heading. Collision occurs if the relative distance between the pursuer and the evader is smaller than a prescribed distance. The *backward* reachable set, which is the set of all initial conditions that leads to collision, is introduced in the paper to help solving the collision avoidance problem. The backward reachable set can be computed by solving a modified Hamilton-Jacobi-Isaacs (HJI) equation

$$\frac{\partial \phi(X, \tau)}{\partial \tau} + \min\{0, H(X, \nabla \phi(X, \tau))\} = 0, \tag{216}$$

where $\tau = -t$ and the Hamiltonian

$$H(X, p) = \min_{u \in \mathcal{U}} \max_{v \in \mathcal{V}} p^{\mathsf{T}} f(X, u, v). \tag{217}$$

The terminal condition is given by $\phi(X, 0) = \phi_0(x)$. Note that (216) is solved from time 0 backwards to some $\tau = -t < 0$. The backward reachable set can be represented as

$$\mathcal{G}(t) = \{X \in \mathbb{R}^3 | \phi(X, -t) \leq 0\}. \tag{218}$$

The previous method requires to solve the HJI equation in three dimension. An

alternative way is to project the true reachable set of a high dimensional system into a collection of lower dimensional subspaces, where computation is less expensive and reconstruct an over-approximation of the full reachable set from these lower dimensional reachable sets. For the problem in [100], a two-dimensional projection of the reachable set with index $ij \in \{12, 13, 23\}$ is governed by the following HJI equation

$$\begin{aligned} \frac{\partial \phi_{ij}(Y_{ij}, \tau)}{\partial \tau} + H(Y_{ij}, \nabla \phi_{ij}(Y_{ij}, t)) = 0, \\ \phi_{ij}(Y_{ij}, 0) = \pi_{ij}[\phi_0](Y_{ij}), \end{aligned} \tag{219}$$

with Hamiltonian

$$H(Y_{ij}, p) = \min_{Y_k \in \mathcal{F}_k(\mathcal{G}, Y_{ij})} p_i f_i(Y_i, Y_j, Y_k) + p_j f_j(Y_i, Y_j, Y_k), \tag{220}$$

where $\pi_{ij}[\cdot]$ is a projection such that $\pi_{ij}[X] = [X_i, X_j]^\intercal$. The function $p(x) = \nabla(\pi_{ij}^{-1}[\phi_{ij}](x, t))$ is the gradient of the projections implicit surface function, and p_i, p_j, p_k are its components. The set valued slice function $\mathcal{F}_k(\mathcal{M}, Y_{ij})$ for some $\mathcal{M} \in \mathbb{R}^3$ and $Y_{ij} \in \mathbb{R}^2$ is defined as

$$\mathcal{F}_k(\mathcal{M}, Y_{ij}) = \{Y_k \in \mathbb{R} | \exists X \in \mathcal{M} \text{ s.t. } \pi_{ij}[X] = Y_{ij} \text{ and } \pi_k[X] = Y_k\}. \tag{221}$$

In the case of pursuit-evasion problems under general flow fields, the dynamics of the system cannot be represented by relative locations in general. As a result, the dimension of the HJI equation for backward propagation of reachable sets is normally too large for the level set method to work efficiently. Hence, the reachability-based approach we proposed is more suitable for this task. For example, consider two Dubins vehicles [117] in a place under external flow fields, each vehicle has three degrees of freedom and the joint state space is of dimension six. Instead of solving

a HJI equation in dimension six directly, we can separate the two players and for each player, we project its reachable set on the plane and propagate the projected reachable set to reduce the computational complexity from propagating the original reachable set. Then the evader is guaranteed to be captured by the pursuer when there exist some time such that the projected reachable set of the evader is fully covered by that of the pursuer.

In the pursuit-evasion game formulation we have been dealing with, the goal is to $\min_{\mathbf{u}} \max_{\mathbf{v}} T$, that is, the pursuer aims to minimize the time-to-capture whereas the evader tries to maximize the time-to-capture. This problem can be solved by a reachability-based approach, where the termination of the game is related to reachable set inclusions. For this purpose, we need to propagate the time-dependent reachable sets of the pursuer and the evader. Now, consider a generalization of the pursuer evasion game, where the objective is to $\min_{\mathbf{u}} \max_{\mathbf{v}} \tau$, where $\tau = \int_0^T p(t)$ and $p(t) \in \mathcal{L}^1(\mathbb{R})$. If $p(t) = 1$, the problem is reduced to the original time-optimal problem. When $p(t) = |\mathbf{u}(t)|$, it becomes a fuel-optimal problem and when $p(t) = \mathbf{u}(t)^{\mathsf{T}}\mathbf{u}(t)$, an energy-optimal problem is formed. Also note that $\frac{\mathrm{d}\tau}{\mathrm{d}t} = p(t)$. To solve this generalized problem, we need to replace the metric t used in the original problem with the generalized metric τ. After such transformation, the boundary of the reachable set of the pursuer at τ can be represented by the zero-level set of an implicit function $\psi(\tau, X)$. We propose to find the partial differential equation that governs this implicit function $\psi(\tau, X)$ and solve the pursuit-evasion game under generalized metric.

9.3 Differential Games under Dynamical Uncertainties with Learned Dynamics

Most of the system models, whether the model of a plane or a robot, are only approximate and are affected by the dynamical uncertainties. A possible direction is to learn the difference between the real model and the analytic model via machine learning methods, such as Locally Weighted Projection Regression (LWPR), and to

apply Game Theoretic Differential Dynamic Programming (GT-DDP) [139, 140] on the learned model to solve differential games under dynamic uncertainties. LWPR is a nonparametric local learning algorithm for nonlinear function approximation in high dimensional spaces through a combination of weighted locally linear regression models.

Different learning methods can be applied to learn the dynamics uncertainties of differential games, such as Gaussian Process [125] and neural networks [53]. Each learning method has its own advantages and disadvantages, so in practice, the learning method should be selected on a case-by-case basis. In order to achieve robust performance, we can augment the learned dynamics with some disturbances apply the GT-DDP algorithm on the new differential game problem to find the optimal controls and feedback gains. The next step is to establish a method that can integrate the learning part and the control part together to enhance the performance of the algorithm.

REFERENCES

[1] Abbeel, P., Coates, A., Quigley, M., and Ng, A. Y., "An application of reinforcement learning to aerobatic helicopter flight," in *In Advances in Neural Information Processing Systems 19*, p. 2007, MIT Press, 2007.

[2] Alpern, S., Fokkink, R., Timmer, M., and Casas, J., "Ambush frequency should increase over time during optimal predator search for prey," *Journal of the Royal Society Interface*, vol. 8, no. 64, pp. 1665–1672, 2011.

[3] Anderson, R. P., Bakolas, E., Milutinović, D., and Tsiotras, P., "Optimal feedback guidance of a small aerial vehicle in a stochastic wind," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 4, pp. 975–985, 2013.

[4] Arslan, G., Marden, J. R., and Shamma, J. S., "Autonomous vehicle-target assignment: A game-theoretical formulation," *Journal of Dynamic Systems, Measurement, and Control*, vol. 129, no. 5, pp. 584–596, 2007.

[5] Atkeson, C. G. and Stephens, B. J., "Random sampling of states in dynamic programming," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 38, pp. 924–929, Aug 2008.

[6] Atkeson, C. G.and Morimoto, J., "Nonparametric representation of policies and value functions: A trajectory-based approach," in *Neural Information Processing Systems*, pp. 1611–1618, MIT Press, 2003.

[7] Bagchi, A. and Olsder, G. J., "Linear-quadratic stochastic pursuit-evasion games," *Applied Mathematics & Optimization*, vol. 7, no. 1, pp. 95–123, 1981.

[8] Bajorski, P., *Statistics for Imaging, Optics, and Photonics*, vol. 808. John Wiley & Sons, 2011.

[9] Bakolas, E. and Tsiotras, P., "The Zermelo-Voronoi diagram: A dynamic partition problem," *Automatica*, vol. 46, no. 12, pp. 2059–2067, 2010.

[10] Bakolas, E. and Tsiotras, P., "Optimal pursuer and moving target assignment using dynamic Voronoi diagrams," in *American Control Conference*, (San Francisco, CA), pp. 5444–5449, June 29 - July 1 2011.

[11] Bakolas, E. and Tsiotras, P., "Feedback navigation in an uncertain flowfield and connections with pursuit strategies," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 35, pp. 1268–1279, July-August 2012.

[12] Bakolas, E. and Tsiotras, P., "Optimal partitioning for spatiotemporal coverage in a drift field," *Automatica*, vol. 49, no. 7, pp. 2064–2073, 2013.

[13] Bakolas, E. and Tsiotras, P., "Minimum-time paths for a light aircraft in the presence of regionally-varying strong winds," in *AIAA Infotech@Aerospace*, (Atlanta, GA), April 20–22, 2010. AIAA Paper 2010-3380.

[14] Bakolas, E. and Tsiotras, P., "Relay pursuit of a maneuvering target using dynamic Voronoi diagrams," *Automatica*, vol. 48, pp. 2213–2220, Aug. 2012.

[15] Bannikov, A. S., "A non-stationary problem of group pursuit," *Journal of Computer and Systems Sciences International*, vol. 48, no. 4, pp. 527–532, 2009.

[16] Başar, T. and Berhard, P., *H-infinity Optimal Control and Related Minimax Design*. Boston: Birkhauser, 1995.

[17] Başar, T. and Olsder, G. J., *Dynamic Noncooperative Game Theory*, vol. 23. Siam, 1999.

[18] Blagodatskikh, A. I., "Group pursuit in Pontryagin's nonstationary example," *Differential Equations*, vol. 44, no. 1, pp. 40–46, 2008.

[19] Blagodatskikh, A. I., "Simultaneous multiple capture in a simple pursuit problem," *Journal of Applied Mathematics and Mechanics*, vol. 73, no. 1, pp. 36–40, 2009.

[20] Book, W. J., "Modeling design, and control of flexible manipulator arms: A tutorial review," in *Proceedings of the 29th IEEE Conference on Decision and Control, Honolulu, Hawaii*, pp. 500–506, 1990.

[21] Bopardikar, S. D., Bullo, F., and Hespanha, J. P., "A cooperative homicidal chauffeur game," *Automatica*, vol. 45, no. 7, pp. 1771–1777, 2009.

[22] Brammer, R. F., "Controllability in linear autonomous systems with positive controllers," *SIAM Journal on Control*, vol. 10, no. 2, pp. 339–353, 1972.

[23] Bryson, A. E. and Ho, Y.-C., *Applied Optimal Control: Optimization, Estimation, and Control*. Taylor & Francis, 1975.

[24] Bullo, F., Frazzoli, E., Pavone, M., Savla, K., and Smith, S. L., "Dynamic vehicle routing for robotic systems," *Proceedings of the IEEE*, vol. 99, no. 9, pp. 1482–1504, 2011.

[25] Chernous' ko, F. L., "A problem of evasion from many pursuers," *Journal of Applied Mathematics and Mechanics*, vol. 40, no. 1, pp. 11–20, 1976.

[26] Chung, C. F. and Furukawa, T., "A reachability-based strategy for the time-optimal control of autonomous pursuers," *Engineering Optimization*, vol. 40, no. 1, pp. 67–93, 2008.

[27] Chung, C. F., Furukawa, T., and Göktogan, A. H., "Coordinated control for capturing a highly maneuverable evader using forward reachable sets,"

in *IEEE International Conference on Robotics and Automation*, (Orlando, FL, USA), pp. 1336–1341, 2006.

[28] Cockayne, E., "Plane pursuit with curvature constraints," *SIAM Journal on Applied Mathematics*, vol. 15, no. 6, pp. 1511–1516, 1967.

[29] Corpetti, T., Memin, E., and Pérez, P., "Extraction of singular points from dense motion fields: an analytic approach," *Journal of Mathematical Imaging and Vision*, vol. 19, no. 3, pp. 175–198, 2003.

[30] Cortes, J., Martinez, S., Karatas, T., and Bullo, F., "Coverage control for mobile sensing networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.

[31] De Berg, M., Cheong, O., Van Kreveld, M., and Overmars, M., *Computational Geometry: Algorithms and Applications*. Springer, 2008.

[32] Denham, W., "Differential dynamic programming," *IEEE Transactions on Automatic Control*, vol. 16, pp. 389–390, Aug 1971.

[33] Devillers, O., "On deletion in Delaunay triangulations," in *Proceedings of the Fifteenth Annual Symposium on Computational Geometry*, (New York, USA), pp. 181–188, 1999.

[34] Devillers, O., "The Delaunay hierarchy," *International Journal of Foundations of Computer Science*, vol. 13, no. 2, pp. 163–180, 2002.

[35] Devillers, O. and Golin, M., "Dog bites postman: Point location in the moving Voronoi diagram and related problems," *Algorithms–ESA '93*, pp. 133–144, 1993.

[36] Devillers, O., Golin, M., Kedem, K., and Schirra, S., "Queries on Voronoi diagrams of moving points," *Computational Geometry*, vol. 6, no. 5, pp. 315–327, 1996.

[37] Devillers, O. and Teillaud, M., "Perturbations and vertex removal in a 3d Delaunay triangulation," in *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, (Philadelphia, USA), pp. 313–319, 2003.

[38] Dockner, E., *Differential Games in Economics and Management Science*. Cambridge University Press, 2000.

[39] Dong, L., Chai, S., Zhang, B., Nguang, S. K., and Li, X., "Cooperative relay tracking strategy for multi-agent systems with assistance of voronoi diagrams," *Journal of the Franklin Institute*, vol. 353, no. 17, pp. 4422–4441, 2016.

[40] Dorato, P. and Kestenbaum, A., "Application of game theory to the sensitivity design of optimal systems," *IEEE Transactions on Automatic Control*, vol. 12, no. 1, pp. 85–87, 1967.

[41] Doyle, J. C., Glover, K., Khargonekar, P. P., and Francis, B. A., "State-space solutions to standard H_2 and H_∞ control problems," *IEEE Transactions on Automatic control*, vol. 34, no. 8, pp. 831–847, 1989.

[42] Elliott, R. J. and Kalton, N. J., "The existence of value in differential games of pursuit and evasion," *Journal of Differential Equations*, vol. 12, no. 3, pp. 504–523, 1972. doi: 10.1016/0022-0396(72)90022-8.

[43] Erez, T., Tassa, Y., and Todorov, E., "Infinite-horizon model predictive control for periodic tasks with contacts," in *Proceedings of Robotics: Science and Systems*, (Los Angeles, CA, USA), June 2011.

[44] Erickson, G. M., "Differential game models of advertising competition," *European Journal of Operational Research*, vol. 83, no. 3, pp. 431–438, 1995.

[45] Etkin, B., *Dynamics of Atmospheric Flight*. Courier Corporation, 2012.

[46] Exahchos, I. and Tsiotras, P., "An asymmetric version of the two car pursuit-evasion game," in *53rd IEEE Conference on Decision and Control*, (Los Angeles, CA), pp. 4272–4277, 2014.

[47] Exarchos, I., Tsiotras, P., and Pachter, M., "On the suicidal pedestrian differential game," *Dynamic Games and Applications*, vol. 5, pp. 297–317, September 2015.

[48] Filippov, A. F., "On certain questions in the theory of optimal control," *Journal of the Society for Industrial & Applied Mathematics, Series A: Control*, vol. 1, no. 1, pp. 76–84, 1962.

[49] Glover, K. and Doyle, J. C., "State-space formulae for all stabilizing controllers that satisfy an H_∞-norm bound and relations to relations to risk sensitivity," *Systems & Control Letters*, vol. 11, no. 3, pp. 167–172, 1988.

[50] Guelman, M., "Control strategies in a planar pursuit evasion game with energy constraints," *Computers & Mathematics with Applications*, vol. 26, no. 6, pp. 33–41, 1993.

[51] Guibas, L. and Russel, D., "An empirical comparison of techniques for updating Delaunay triangulations," in *Proceedings of the Twentieth Annual Symposium on Computational Geometry*, (New York, USA), pp. 170–179, 2004.

[52] Hájek, O., *Pursuit Games: An Introduction to the Theory and Applications of Differential Games of Pursuit and Evasion*. Mineola, New York: Dover Publications, 2nd ed., 2008.

[53] Haykin, S., *Neural Networks: A Comprehensive Foundation*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2nd ed., 1998.

[54] Hernandez, S. and Paley, D. A., "Three-dimensional motion coordination in a spatiotemporal flowfield," *IEEE Transactions on Automatic Control*, vol. 55, no. 12, pp. 2805–2810, 2010.

[55] Hernández-Martínez, E. G. and Aranda-Bricaire, E., *Convergence and Collision Avoidance in Formation Control: A Survey of the Artificial Potential Functions Approach*, ch. 6, pp. 103–126. INTECH Open Access Publisher, 2011.

[56] Hespanha, J. P., Prandini, M., and Sastry, S., "Probabilistic pursuit-evasion games: a one-step nash approach," in *Proceedings of the 39th IEEE Conference on Decision and Control*, vol. 3, (Sydney, Australia), pp. 2272–2277, IEEE, Dec. 12–15 2000.

[57] Ho, Y.-C., Bryson, A. E., and Baron, S., "Differential games and optimal pursuit-evasion strategies," in *Joint Automatic Control Conference*, no. 3, pp. 37–40, 1965.

[58] Hoover, W. E. and Rockville, M. D., *Algorithms for Confidence Circles and Ellipses*. Citeseer, 1984.

[59] Hui, Q. and Haddad, W. M., "Distributed nonlinear control algorithms for network consensus," *Automatica*, vol. 44, no. 9, pp. 2375–2381, 2008.

[60] Ibragimov, G. I., "Optimal pursuit with countably many pursuers and one evader," *Differential Equations*, vol. 41, no. 5, pp. 627–635, 2005.

[61] Ibragimov, G. I., Salimi, M., and Amini, M., "Evasion from many pursuers in simple motion differential game with integral constraints," *European Journal of Operational Research*, vol. 218, no. 2, pp. 505–511, 2012.

[62] Isaacs, R., *Differential Games: A Mathematical Theory with Applications to Warfare and Pursuit, Control and Optimization.* Mineola, NY: Courier Dover Publications, 1999.

[63] Jacobson, D. H. and Mayne, D. Q., *Differential Dynamic Programming.* New York,: American Elsevier Pub. Co., 1970.

[64] Jadbabaie, A., Lin, J., and Morse, A. S., "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.

[65] James, M. R., Nurdin, H. I., and Petersen, I. R., "H^∞ control of linear quantum stochastic systems," *IEEE Transactions on Automatic Control*, vol. 53, no. 8, pp. 1787–1803, 2008.

[66] Jang, J. S. and Tomlin, C. J., "Control strategies in multi-player pursuit and evasion game," *AIAA*, vol. 6239, pp. 15–18, 2005.

[67] Jorgensen, S., Quincampoix, M., and Vincent, T. L., *Advances in Dynamic Game Theory: Numerical Methods, Algorithms, and Applications to Ecology and Economics*, vol. 9. Springer Science & Business Media, 2007.

[68] Kalman, R. E., "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[69] Kao, T., "Dynamic maintenance of Delaunay triangulations," *Work*, vol. 9, p. 32, 1991.

[70] Khaidarov, B. K., "Positional i-capture in the game of a single evader and several pursuers," *Journal of Applied Mathematics and Mechanics*, vol. 48, no. 4, pp. 406–409, 1984.

[71] Khargonekar, P. P., Petersen, I. R., and Rotea, M. A., "H_∞-optimal control with state-feedback," *IEEE Transactions on Automatic Control*, vol. 33, no. 8, pp. 786–788, 1988.

[72] Khargonekar, P. P., Petersen, I. R., and Zhou, K., "Robust stabilization of uncertain linear systems: quadratic stabilizability and H_∞ control theory," *IEEE Transactions on Automatic Control*, vol. 35, no. 3, pp. 356–361, 1990.

[73] Kim, Y., Gu, D.-W., and Postlethwaite, I., "Real-time optimal mission scheduling and flight path selection," *IEEE Transactions on Automatic control*, vol. 52, no. 6, pp. 1119–1123, 2007.

[74] Kopparty, S. and Ravishankar, C. V., "A framework for pursuit evasion games in rn," *Information Processing Letters*, vol. 96, no. 3, pp. 114–122, 2005.

[75] Krasovskiĭ, A. N. and Krasovskiĭ, N. N., *Control under Lack of Information*. Springer Science & Business Media, 1994.

[76] Kushner, H. J., "Numerical approximations for stochastic differential games," *SIAM Journal on Control and Optimization*, vol. 41, no. 2, pp. 457–486, 2002.

[77] Lakatos, E. and Stumpf, M., "Control mechanisms for stochastic biochemical systems via computation of reachable sets," *bioRxiv*, p. 079723, 2016.

[78] Le Ny, J. and Feron, E., "Performance evaluation of a multi-agent risk-sensitive tracking system," in *46th IEEE Conference on Decision and Control*, (New Orleans, LA), pp. 2464–2469, IEEE, Dec. 12–14 2007.

[79] Lee, I. and Gahegan, M., "Interactive analysis using Voronoi diagrams: Algorithms to support dynamic update from a generic triangle-based data structure," *Transactions in GIS*, vol. 6, no. 2, pp. 89–114, 2002.

[80] Li, D. and Cruz, J. B., "A two-player stochastic pursuit-evasion differential game," in *46th IEEE Conference on Decision and Control*, (New Orleans, LA), pp. 4057–4062, IEEE, Dec 12–14 2007.

[81] Lin, Z. and Saberi, A., "Semi-global exponential stabilization of linear discrete-time systems subject to input saturation via linear feedbacks," *Systems & Control Letters*, vol. 24, no. 2, pp. 125–132, 1995.

[82] Lolla, T., *Path Planning and Adaptive Sampling in the Coastal Ocean*. PhD thesis, Department of Mechanical Engineering, Massachusetts Institute of Technology, February 2016.

[83] Lolla, T., Haley Jr., P. J., and Lermusiaux, P. F. J., "Time-optimal path planning in dynamic flows using level set equations: realistic applications," *Ocean Dynamics*, vol. 64, no. 10, pp. 1399–1417, 2014.

[84] Lolla, T., Haley Jr., P. J., and Lermusiaux, P. F. J., "Path planning in multi-scale ocean flows: Coordination and dynamic obstacles," *Ocean Modelling*, vol. 94, pp. 46–66, 2015.

[85] Lolla, T. and Lermusiaux, P. F. J., "A forward reachability equation for minimum-time path planning in strong dynamic flows," *SIAM Journal on Control and Optimization*, 2015. sub–judice.

[86] Lolla, T., Lermusiaux, P. F. J., Ueckermann, M. P., and Haley Jr., P. J., "Time-optimal path planning in dynamic flows using level set equations: theory and schemes," *Ocean Dynamics*, vol. 64, no. 10, pp. 1373–1397, 2014.

[87] Lolla, T., Lermusiaux, P. F. J., Ueckermann, M. P., and Haley Jr., P. J., "Time-optimal path planning in dynamic flows using level set equations: theory and schemes," *Ocean Dynamics*, vol. 64, no. 10, pp. 1373–1397, 2014.

[88] Lolla, T., Ueckermann, M. P., Yiğit, K., Haley Jr, P. J., and Lermusiaux, P. F. J., "Path planning in time dependent flow fields using level set methods," in *IEEE International Conference on Robotics and Automation*, (St.Paul, MN), pp. 166–173, May 14–18 2012.

[89] Lua, C. A., Altenburg, K., and Nygard, K. E., "Synchronized multipoint attack by autonomous reactive vehicles with simple local communication," in *Proceedings of the 2003 IEEE Swarm Intelligence Symposium*, (Indianapolis, IN), pp. 95–102, IEEE, April 24–26 2003.

[90] Luce, R. D. and Raiffa, H., *Games and Decisions: Introduction and Critical Survey*. Courier Corporation, 2012.

[91] Mahoney, L. E., "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *American Journal of Mathematics*, pp. 497–516, 1957.

[92] Manhães de Castro, P. M., Tournois, J., Alliez, P., and Devillers, O., "Filtering relocations on a delaunay triangulation," in *Computer Graphics Forum*, vol. 28, pp. 1465–1474, Wiley Online Library, 2009.

[93] Mayne, D., "A second-order gradient method for determining optimal trajectories of non-linear discrete-time systems," *International Journal of Control*, vol. 3, no. 1, pp. 85–95, 1966.

[94] McLain, T. and Beard, R., "Trajectory planning for coordinated rendezvous of unmanned air vehicles," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, p. 4369, August 2000.

[95] McLain, T. W., Chandler, P. R., Rasmussen, S., and Pachter, M., "Cooperative control of uav rendezvous," in *Proceedings of the 2001 American*

Control Conference, vol. 3, (Arlington, VA), pp. 2309–2314, IEEE, June 25–27 2001.

[96] McNeely, R. L., Iyer, R. V., and Chandler, P. R., "Tour planning for an unmanned air vehicle under wind conditions," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 5, pp. 1299–1306, 2007.

[97] Merz, A. W., "The game of two identical cars," *Journal of Optimization Theory and Applications*, vol. 9, no. 5, pp. 324–343, 1972.

[98] Min, C., "Local level set method in high dimension and codimension," *Journal of Computational Physics*, vol. 200, no. 1, pp. 368–382, 2004.

[99] Mitchell, I. M., Bayen, A. M., and Tomlin, C. J., "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games," *IEEE Transactions on Automatic Control*, vol. 50, no. 7, pp. 947–957, 2005.

[100] Mitchell, I. M. and Tomlin, C. J., "Overapproximating reachable sets by hamilton-jacobi projections," *Journal of Scientific Computing*, vol. 19, no. 1-3, pp. 323–346, 2003.

[101] Mitrovic, D., Klanke, S., and Vijayakumar, S., "Adaptive optimal feedback control with learned internal dynamics models," in *From Motor Learning to Interaction Learning in Robots*, pp. 65–84, Springer, 2010.

[102] Mizukami, K. and Eguchi, K., "A geometrical approach to problems of pursuit-evasion games," *Journal of the Franklin Institute*, vol. 303, no. 4, pp. 371–384, 1977.

[103] Moreau, L., "Stability of multiagent systems with time-dependent communication links," *IEEE Transactions on Automatic Control*, vol. 50, no. 2, pp. 169–182, 2005.

[104] Morimoto, J. and Atkeson, C., "Minimax differential dynamic programming: An application to robust biped walking," in *In Advances in Neural Information Processing Systems 15*, Cambridge, MA: MIT Press, 2002.

[105] Morimoto, J., Zeglin, G., and Atkeson, C. G., "Minimax differential dynamic programming: application to a biped walking robot," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, pp. 1927–1932 vol.2, Oct. 2003.

[106] Morrow, J. D., *Game Theory for Political Scientists*. Princeton University Press Princeton, NJ, 1994.

[107] Nahin, P. J., *Chases and Escapes: the Mathematics of Pursuit and Evasion*. Princeton University Press, 2012.

[108] Observatory, E., "Comparing the winds of Sandy and Katrina," Nov. 6 2012. `https://earthobservatory.nasa.gov/IOTD/view.php?id=79626&src=ve`.

[109] Okabe, A., Boots, B., Sugihara, K., and Chiu, S. N., *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, vol. 501. Wiley, 2009.

[110] Olfati-Saber, R., Fax, J. A., and Murray, R. M., "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

[111] Osher, S. and Fedkiw, R., *Level Set Methods and Dynamic Implicit Surfaces*, vol. 153. Springer Science & Business Media, 2006.

[112] Pachter, M., "Linear-quadratic reversed Stackelberg differential games with incentives," *IEEE Transactions on Automatic Control*, vol. 29, no. 7, pp. 644–647, 1984.

[113] Pachter, M. and Yavin, Y., "A stochastic homicidal chauffeur pursuit-evasion differential game," *Journal of Optimization Theory and Applications*, vol. 34, no. 3, pp. 405–424, 1981.

[114] Paley, D. A. and Peterson, C., "Stabilization of collective motion in a time-invariant flowfield," *Journal of Guidance, Control, and Dynamics*, vol. 32, no. 3, pp. 771–779, 2009.

[115] Pan, Y. and Theodorou, E., "Probabilistic differential dynamic programming," in *Advances in Neural Information Processing Systems*, pp. 1907–1915, 2014.

[116] Pashkov, A. G. and Terekhov, S. D., "A differential game of approach with two pursuers and one evader," *Journal of Optimization Theory and Applications*, vol. 55, no. 2, pp. 303–311, 1987.

[117] Patsko, V. S. and Turova, V. L., "Numerical study of the homicidal chauffeur differential game with the reinforced pursuer," *Game Theory Appl*, vol. 12, no. 8, pp. 123–152, 2007.

[118] Pavone, M., Arsie, A., Frazzoli, E., and Bullo, F., "Distributed algorithms for environment partitioning in mobile robotic networks," *IEEE Transactions on Automatic Control*, vol. 56, no. 8, pp. 1834–1848, 2011.

[119] Petersen, I., "Disturbance attenuation and H^{∞} optimization: A design method based on the algebraic riccati equation," *IEEE Transactions on Automatic Control*, vol. 32, no. 5, pp. 427–429, 1987.

[120] Peterson, C. and Paley, D. A., "Multivehicle coordination in an estimated time-varying flowfield," *Journal of guidance, control, and dynamics*, vol. 34, no. 1, pp. 177–191, 2011.

[121] Petrov, N. N. and Shuravina, I. N., "On the "soft" capture in one group pursuit problem," *Journal of Computer and Systems Sciences International*, vol. 48, no. 4, pp. 521–526, 2009.

[122] Pittsyk, M. and Chikrii, A. A., "On a group pursuit problem," *Journal of Applied Mathematics and Mechanics*, vol. 46, no. 5, pp. 584–589, 1982.

[123] Pshenichnyi, B. N., "Simple pursuit by several objects," *Cybernetics and Systems Analysis*, vol. 12, no. 3, pp. 484–485, 1976.

[124] Ramana, M. B. V. and Kothari, M., "A cooperative pursuit-evasion game of a high speed evader," in *54th IEEE Conference on Decision and Control*, (Osaka, Japan), pp. 2969–2974, Dec. 15–18 2015.

[125] Rasmussen, C. E. and Williams, C. K. I., *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[126] Ren, W., Beard, R. W., and Atkins, E. M., "Information consensus in multivehicle cooperative control," *IEEE Control Systems*, vol. 27, no. 2, pp. 71–82, 2007.

[127] Reynolds, C. W., "Steering behaviors for autonomous characters," in *Game developers conference*, pp. 763–782, 1999.

[128] Reynolds, C. W., "Flocks, herds and schools: A distributed behavioral model," *ACM SIGGRAPH computer graphics*, vol. 21, no. 4, pp. 25–34, 1987.

[129] Roos, T., "Voronoi diagrams over dynamic scenes," *Discrete Applied Mathematics*, vol. 43, no. 3, pp. 243–259, 1993.

[130] Rublein, G. T., "On Pursuit with Curvature Constraints," *SIAM Journal on Control*, vol. 10, no. 1, pp. 37–39, 1972.

[131] Salmon, D. M. and Heine, W., "Reachable sets analysis-an efficient technique for performing missile/sensor tradeoff studies," *AIAA Journal*, vol. 11, no. 7, pp. 927–931, 1973.

[132] Schmitendorf, W. E. and Barmish, B. R., "Null controllability of linear systems with constrained controls," *SIAM Journal on Control and Optimization*, vol. 18, no. 4, pp. 327–345, 1980.

[133] Sethian, J. A., *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*, vol. 3. Cambridge University Press, 1999.

[134] Sgall, J., "Solution of David Gale's Lion and Man Problem," *Theoretical Computer Science*, vol. 259, no. 1, pp. 663–670, 2001.

[135] Shi, W. and Liu, W., "A stochastic process-based model for the positional error of line segments in gis," *International Journal of Geographical Information Science*, vol. 14, no. 1, pp. 51–66, 2000.

[136] Sontag, E. D., *Mathematical Control Theory: Deterministic Finite Dimensional Systems*, vol. 6. Springer Science & Business Media, 2013.

[137] Starr, A. W. and Ho, Y.-C., "Nonzero-sum differential games," *Journal of Optimization Theory and Applications*, vol. 3, no. 3, pp. 184–206, 1969.

[138] Stulp, F., Theodorou, E. A., and Schaal, S., "Reinforcement learning with sequences of motion primitives for robust manipulation," *IEEE Transactions on Robotics*, vol. 28, no. 6, pp. 1360–1370, 2012.

[139] Sun, W., Theodorou, E., and Tsiotras, P., "Game theoretic continuous time differential dynamic programming," in *American Control Conference*, (Chicago, IL), pp. 5593–5598, July 1–3 2015.

[140] Sun, W., Theodorou, E. A., and Tsiotras, P., "Stochastic game theoretic trajectory optimization in continuous time," in *55th Conference on Decision and Control (CDC)*, (Las Vegas, USA), pp. 6167–6172, IEEE, Dec. 12–14 2016.

[141] Sun, W. and Tsiotras, P., "A sequential pursuer-target assignment problem under external disturbances," in *52nd Annual Conference on Decision and Control (CDC)*, (Firenze, Italy), pp. 3994–3999, IEEE, Dec. 10 – 13 2013.

[142] Sun, W. and Tsiotras, P., "Pursuit evasion game of two players under an external flow field," in *American Control Conference*, (Chicago, IL, USA), pp. 5617–5622, July 1–3 2015.

[143] Sussmann, H., Sontag, E., and Yang, Y., "A general result on the stabilization of linear systems using bounded controls," *IEEE Transactions on Automatic Control*, vol. 39, no. 12, pp. 2411–2425, 1994.

[144] Tassa, Y., Erez, T., and Smart, W. D., "Receding horizon differential dynamic programming," in *Advances in Neural Information Processing Systems* (Platt, J. C., Koller, D., Singer, Y., and Roweis, S. T., eds.), Curran Associates, Inc., 2007.

[145] Tassa, Y., Erez, T., and Todorov, E., "Synthesis and stabilization of complex behaviors through online trajectory optimization," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pp. 4906–4913, Oct 2012.

[146] Tassa, Y., Mansard, N., and Todorov, E., "Control-limited differential dynamic programming," in *Proceedings International Conference on Robotics and Automation*, (Seattle, WA), 2014.

[147] Theodorou, E., Tassa, Y., and Todorov, E., "Stochastic differential dynamic programming," in *American Control Conference*, (Baltimore, MD), pp. 1125–1132, June 30 - July 2 2010.

[148] Todorov, E. and Li, W., "A generalized iterative LQG method for locally-optimal feedback control of constrained nonlinear stochastic systems," in *American Control Conference*, (Portland, OR), pp. 300–306, June 8–10 2005.

[149] Van Der Schaft, A. J., "L_2-gain analysis of nonlinear systems and nonlinear state-feedback H_∞ control," *IEEE Transactions on Automatic Control*, vol. 37, no. 6, pp. 770–784, 1992.

[150] Vidal, R., Shakernia, O., Kim, H. J., Shim, D. H., and Sastry, S., "Probabilistic pursuit-evasion games: theory, implementation, and experimental evaluation," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 662–669, 2002.

[151] Von Neumann, J. and Morgenstern, O., *Theory of Games and Economic Behavior*. Princeton University Press, 2007.

[152] Voronoï, G., "Nouvelles applications des paramètres continus à la théorie des formes quadratiques. deuxième mémoire. recherches sur les parallélloèdres primitifs.," *Journal für die reine und angewandte Mathematik*, vol. 134, pp. 198–287, 1908.

[153] Vukobratovic, M., Borovac, B., Surla, D., and Stokic, D., *Biped Locomotion: Dynamics, Stability, Control and Application*, vol. 7. Springer Science & Business Media, 2012.

[154] Wie, B., Liu, Q., and Byun, K.-W., "Robust H_∞ control synthesis method and its application to benchmark problems," *Journal of guidance, control, and dynamics*, vol. 15, no. 5, pp. 1140–1148, 1992.

[155] Wilson, E. B. and Hilferty, M. M., "The distribution of chi-square," *Proceedings of the National Academy of Sciences*, vol. 17, no. 12, pp. 684–688, 1931.

[156] Xie, L. and de Souza Carlos, E., "Robust H_∞ control for linear systems with norm-bounded time-varying uncertainty," *IEEE Transactions on Automatic Control*, vol. 37, no. 8, pp. 1188–1191, 1992.

[157] Yeung, D. W. K. and Petrosjan, L. A., *Cooperative Stochastic Differential Games*. Springer Science & Business Media, 2006.

[158] Zanardi, C., Hervé, J.-Y., and Cohen, P., "Escape strategy for a mobile robot under pursuit," in *IEEE International Conference on Systems, Man and Cybernetics*, vol. 4, (Vancouver, BC, Canada), pp. 3304–3309, 1995.

[159] Zermelo, E., "Über das Navigationsproblem bei ruhender oder veränderlicher Windverteilung," *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 11, no. 2, pp. 114–124, 1931.

[160] Zhou, K., Doyle, J. C., and Glover, K., *Robust and Optimal Control*, vol. 40. Prentice Hall New Jersey, 1996.

[161] Zhu, Q., Tembine, H., and Başar, T., "Hybrid risk-sensitive mean-field stochastic differential games with application to molecular biology," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, (Orlando, FL, USA), pp. 4491–4497, Dec. 12–15 2011.

VITA

Wei Sun received his BS. degree in Mathematics from the Peking University, China, in 2010 and his MS. degrees in both Aerospace Engineering and Mathematics from the Georgia Institute of Technology, Atlanta, in 2015. He is currently a Ph.D. candidate at the Georgia Institute of Technology. His current research interests include optimal control, differential games, reinforcement learning and trajectory optimization with applications in motion planning, and pursuit-evasion games.