Harvest-Time Optimal Path Planning in Dynamic Flows

by

Manmeet Singh Bhabra

B.Eng., McGill University (2018)

Submitted to the Department of Mechanical Engineering
and
Center for Computational Science & Engineering

in partial fulfillment of the requirements for the degrees of

Master of Science in Mechanical Engineering
and
Master of Science in Computational Science & Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2021

Author . . . .
Department of Mechanical Engineering
and
Center for Computational Science & Engineering
August 13, 2021

Certified by . . . .
Pierre F.J. Lermusiaux
Professor, Department of Mechanical Engineering
Thesis Supervisor

Accepted by . . . .
Youssef M. Marzouk
Co-Director, Center for Computational Science and Engineering

Accepted by . . . .
Nicolas Hadjiconstantinou
Chairman, Department Committee on Graduate Theses

Harvest-Time Optimal Path Planning in Dynamic Flows

by

Manmeet Singh Bhabra

Submitted to the Department of Mechanical Engineering
and
Center for Computational Science & Engineering
on August 13, 2021, in partial fulfillment of the
requirements for the degrees of
Master of Science in Mechanical Engineering
and
Master of Science in Computational Science & Engineering

Abstract

The past decade has seen an increasing use of autonomous vehicles (propelled AUVs, ocean gliders, solar-vehicles, etc.) in marine applications. For the operation of these vehicles, efficient methods for path planning are critical. Path planning, in the most general sense, corresponds to a set of rules to be provided to an autonomous robot for navigating from one configuration to another in some optimal fashion. Increasingly, having autonomous vehicles that optimally collect/harvest external fields from highly dynamic environments has grown in relevance. Autonomously maximizing the harvest in minimum time is our present path planning objective. Such optimization has numerous impactful applications. For instance, in the case of energy optimal path planning where long endurance and low power are crucial, it is important to be able to optimally harvest energy (solar, wind, wave, thermal, etc.) along the way and/or leverage the environment (winds, currents, etc.) to reduce energy expenditure. Similarly, autonomous marine cleanup or collection vehicles, tasked with harvesting plastic waste, oil spills, or seaweed fields, need to be able to plan paths that maximize the amount of material harvested in order to optimize the cleanup or collection process.

In this work, we develop an exact partial differential equation-based methodology that predicts harvest-time optimal paths for autonomous vehicles navigating in dynamic environments. The governing differential equations solve the multi-objective optimization problem of navigating a vehicle autonomously in a highly dynamic flow field to any destination with the goal of minimizing travel time while also maximizing the amount harvested by the vehicle. Using Hamilton-Jacobi theory for reachability, our methodology computes the exact set of Pareto optimal solutions to the multi-objective path planning problem. This is completed by numerically solving a reachability problem for the autonomous vehicle in an augmented state space consisting of the vehicle's position in physical space as well as its harvest state. Our approach is applicable to path planning in various environments, however we primarily present examples of navigating in dynamic ocean flows. The following cases, in particular, are studied. First, we validate our methodology using a benchmark case of planning paths through a region with a harvesting field present in a halfspace, as this case admits a semi-analytical solution that we compare to the results of our method. We next consider a more complex unsteady environment as we solve for harvest-time optimal missions in a quasi-geostrophic double-gyre ocean flow field. Following this, we provide harvest-time optimal paths to the highly relevant issue of collecting harmful algae blooms. Our final case considers an application to next generation offshore aquaculture technologies. In particular, we consider in this case path planning of an offshore moving fish farm that accounts for optimizing fish growth. Overall, we find that our exact planning equations and efficient schemes are promising to address several pressing challenges for our planet.

Thesis Supervisor: Pierre F.J. Lermusiaux
Title: Professor, Department of Mechanical Engineering

Acknowledgments

This work would not have been possible without the contributions from a number of people.

First and foremost, I want to express my gratitude to my advisor, Prof. Pierre Lermusiaux. Thank you Pierre for your continued support and guidance throughout my graduate studies, and for giving me the freedom and flexibility to tackle research problems of my interest. Working under Pierre's mentorship has allowed me to grow as a researcher, as he has taught me how to properly approach open scientific problems in a precise and systematic manner. I will definitely miss his quirky humor, especially all the jokes about Canada and its freezing temperatures! Pierre, I am truly grateful for having had the opportunity to work with you.

Next, I don't think I can thank enough the members of the MSEAS group, both past and present, who have become a family away from home! Manan, this thesis should have your name on it alongside mine as this work would definitely not have been possible with you. Working together to build up the path planning idea we fortuitously stumbled across 2 years ago has been great. Spending time with you talking about topics ranging from research to just life in general has been a highlight of my time at MIT. Abhinav, thank you for all the great times, your wonderful friendship and for being the dependable anchor everyone at MSEAS can look to. I look back fondly upon the times we had together and will miss your hilarious jokes as well as your great Indian food. Corbin, thank you for all the "Chon" runs, poker nights, glow-in-the-dark ultimate frisbee games and shared love for songs that rip. Working with you has not just been great, it has also been, as you would say, "a silly goose time". Thank you Akis, for not only being such a great friend, but for also being a big source of support throughout my graduate studies. I'm going to definitely miss our late afternoon chats in the lab as well as your sailing lessons on the Charles River. Thank you Aaron for all the great laughs, for sharing my enthusiasm for all things hip-hop/rap music related, and, of course, introducing me to "fire-ice". Thanks Wael for your friendship and for being a great source of support when I first joined

MSEAS. I definitely will miss working with you on all things acoustics related, and helping you make sense of seeing "ghosts" of Pat during all-nighters before deadlines. Tony, thanks for being a wonderful friend and for all the fun times playing pickup soccer together. Mike and Jacob, thanks for all the great times and for regaling me with all your jokes about Canada. Clara, thanks for your camaraderie and for helping me review big parts of this thesis. To Chinmay, Aman, Aditya, Zach, Ben and all other lab members, thank you for making MSEAS such a fun and welcoming place. Thank you also Dr. Pat Haley, Dr. Chris Mirabito and Lisa (aka Linda) Mayer for all your help and support.

I want to also thank all my friends outside of MSEAS for helping make my time at MIT a truly memorable one. Thanks Mark for your amazing friendship. I'll definitely miss the times we spent camping out for cheap last minute resale tickets to Celtics games at the Garden! Tony T., Vidit, Marek and all the rest of the MIT pickup soccer crew, thanks for all the fun times. Marc-Andre and Alvin, thanks for all the great times at weekend brunches. To all my friends in Canada, especially the "Goon Squad" and the Kuper crew, thanks for all the messages, Zoom calls and laughs – you guys are the best.

I would like to also thank Prof. Siva Nadarajah, my undergraduate advisor during my studies at McGill University. I have been able to pursue research in graduate school in the field of numerical methods and computational sciences largely as a result of your mentorship and guidance early in my academic career.

Moreover, we are grateful to the Office of Naval Research (ONR) for research support under grants N00014-14-1-0476 (Science of Autonomy-LEARNS) and N00014-19-1-2693 (IN-BDA), and to the DOE - Office of ARPA-E for support under grant DE-AR0000910 (MARINER) to MIT. I would also like to thank the Fonds de recherche du Québec – Nature et technologies (FRQNT) program for their financial support. Additionally, I am very grateful to grantor of the Epp and Ain Sonin Fellowship for their generous financial support during my first semester at MIT.

Lastly, I would like to thank my family. Thank you Mom, Dad and Avneet. I can't imagine what I would have done without our frequent Facetime calls and text messages. None of this would have been possible without your continued love and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The past decade has seen a surge of interest in the field of autonomy. Engineering advances ranging from faster computer processors to more advanced algorithms in fields such as controls and machine learning have truly accelerated the pace of development of autonomous vehicles. While applications on land (e.g. self driving cars) and air (e.g. autonomous drones) have received quite a bit of attention, autonomous marine vehicles such as autonomous underwater vehicles (AUVs) and autonomous surface vehicles (ASVs) have also seen a proliferation of use. These marine vehicles have been used in numerous applications and research areas including military operations, surveillance, seafloor mapping, and environmental monitoring to name a few (Roberts and Sutton, 2006; Griffiths, 2002).

A central task in the operation of these vehicles is in designing accurate and efficient means to navigate between specified waypoints. This problem falls under the purview of path planning. Path planning, in its most general sense, refers to the determination of a path for a vehicle to follow such that a specific criteria (e.g. travel time, energy usage, safety) is optimized (Lolla et al., 2014b; Lermusiaux et al., 2016, 2017b). As technological advances have been pushed further, an additional objective to be considered in the path planning problem is taking shape – harvesting of background environmental fields (Bhabra et al., 2020; Lermusiaux et al., 2019). In particular, vehicles are being tasked with planning paths while additionally accounting for the harvesting of external fields. This has a number of areas of applicability:

energy constrained autonomous vehicles, increasingly equipped with energy harvesting capabilities, will want to navigate intelligently to harvest renewable energy from the environment and wind, solar, buoyant, thermal, and/or wave sources; waste collection vehicles, tasked with collecting residual waste such as plastic pollution or even oil from spills, need to plan paths while accounting for how much waste can be harvested from the surroundings prior to reaching their destination; autonomously moving offshore farms, a next generation offshore aquaculture concept, will want to plan paths to optimize the amount of nutrients harvested in order to maximize fish or algae growth.

While path planning in static environments has been studied extensively (Latombe, 2012; Leonard and Bahr, 2016), the ocean, on the other hand, is a highly dynamic system that proves to be a more complex environment in which to plan paths. The dynamic currents in oceans have a significant influence on the marine vehicle, with these currents oftentimes having velocities comparable to or much larger than the speed of the vehicle itself (Schmidt et al., 1996; Rudnick et al., 2004; Lolla et al., 2014b). Several methods have been investigated for path planning of vehicles in these dynamic marine environments, including graph search methods, Rapidly Exploring Random Trees (RRTs) and PDE-based approaches (Panda et al., 2020; Li et al., 2018; Zeng et al., 2016; Lermusiaux et al., 2016, 2017b).

The focus of this work is optimal path planning of autonomous vehicles for harvesting applications in highly dynamic ocean environments. In particular, we address the multi-objective optimization problem of planning paths for autonomous vehicles navigating in deterministic dynamic environments that minimize the travel time while simultaneously maximizing the amount of material harvested by the vehicle. We denote this as *harvest-time optimal path planning*. Our approach to address this planning problem is motivated by (Lolla et al., 2014b,a; Doshi, 2021) and builds upon Hamilton-Jacobi reachability. Furthermore, while cases from ocean applications are emphasized and presented in this work, it is important to note that the theory we introduce is not unique to marine applications and is in fact equally applicable to different dynamic environments.

1.1 Layout of Thesis

This thesis is organized as follows. We start in Chapter 2 by presenting a literature review of both algorithms for path planning and different applications for our harvest-time optimal path planning problem. In Chapter 3, we outline the general problem statement that is the focus of this work as well as introduce the key notation that is used throughout this thesis. Chapter 4 is devoted entirely to the theory behind our path planning approach. Here, we start by reviewing the fundamentals of optimal control, following which we review forward reachability. Next, a review is conducted of the theoretical properties of viscosity solutions of Hamilton-Jacobi equations. We then show how this all can be applied to compute globally optimal paths for our harvest-time path planning problem. In Chapter 5, we present numerical details regarding the implementation of our algorithm. A variety of applications of our path planning methodology is illustrated in Chapter 6. We first consider a benchmark example problem which admits a semi-analytical solution in order to validate our algorithm. This is followed by a number of different harvesting applications in more complex and realistic ocean flows. In Chapter 7, preliminary work is presented for how our approach can be extended in the case of vehicles navigating in uncertain environments. Finally, conclusions and possible avenues for future work are discussed in Chapter 8.

Chapter 2

Literature Review

In this chapter, we review common methods and algorithms for optimal path planning, and we outline some different relevant applications for harvest-time optimal path planning, with an emphasis on marine autonomy applications. We note that many of the existing algorithms provide approximate solutions to the optimal path planning problem and that none consider a rigorous comprehensive approach to predicting generic harvest optimal paths in dynamic environments. This is a motivation for our work.

2.1 Path Planning Algorithms in Marine Applications

In this section, we present a brief overview of some of the most popular approaches applied to path planning for autonomous marine vehicles. For more details, we refer the reader to (Panda et al., 2020; Li et al., 2018; Zeng et al., 2016; Lermusiaux et al., 2016; Lolla, 2012) and the references cited therein.

2.1.1 Graph Search Algorithms

Some of the earliest developed algorithms for path planning were graph search schemes. These algorithms are adapted for discrete search spaces composed of vertices along with edges specifying the cost of travelling between vertex neighbors. For navigating from a start node to an end node, Dijkstra's algorithm is one popular approach. The original algorithm addresses the problem of finding the shortest path between two nodes on a graph. This is done by assigning some initial distance value from each node to the start node on the graph, and then iteratively improving on this distance step by step by maintaining a set of visited and unvisited nodes (LaValle, 2006). The A* algorithm can be seen as an extension of Dijkstra's algorithm, and is another quintessential scheme used in discrete path planning. Here, a path cost of any node x in the graph is given in the form

$$f(x) = g(x) + h(x), \tag{2.1}$$

where $g(x)$ is the cost of the path from the start node to x and $h(x)$ is a heuristic function estimating the cost to go from x to the goal. The A* algorithm then aims to use this estimate of the cost of candidate paths when checking from a current node which neighbors to consider in forming the final path (LaValle, 2006).

Graph search schemes – such as A*, Dijkstra's algorithm, and others – have been tried in a variety of marine path planning applications. Carroll et al. (1992) was one of the first to do this by applying A* on a search space adapted to the ocean current field. In Garau et al. (2005), the method from Carroll et al. (1992) is built upon further to allow for the consideration of energy costs in determining paths in environments consisting of eddies and currents. Kularatne et al. (2016) generate time and energy optimal paths, while accounting for kinematic constraints on vehicle operation, using graph search based methods. In Huynh et al. (2015), nonlinear robust Model Predictive Control is used, where optimal paths are determined using an A*-like algorithm. Koay and Chitre (2013) investigate the effectiveness of current-aware energy-efficient paths, found using an A* algorithm, for AUVs. While graph search methods in general have proven to be relatively effective in marine path planning applications, a key challenge of this approach is that the discrete nature of these schemes result in the consideration of only discrete state transitions that unnaturally

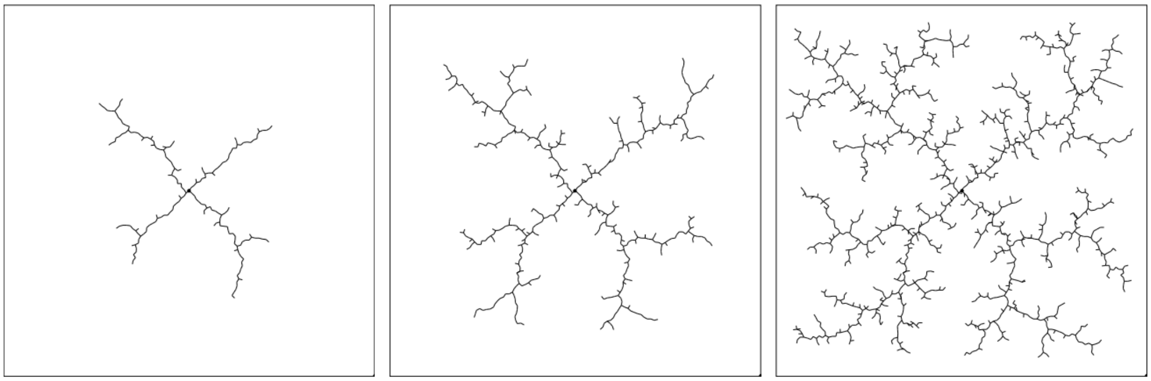


Figure 2-1: An example application showing the tree growth (at three different snapshots) resulting from applying RRT from a start point at the center of the domain (adapted from (LaValle et al., 1998)).

constrain the motion of a vehicle to limited directions (Zeng et al., 2016).

2.1.2 Rapidly exploring Random Trees (RRTs)

Rapidly exploring Random Trees (RRTs) is a probabilistic path planning approach. The method is noted for its ability to efficiently search for paths in nonconvex and high-dimensional spaces (LaValle, 2006; LaValle et al., 1998). RRT attacks the path planning problem by incrementally building a tree from the start point. At each step of the original algorithm, a random point in the state space is sampled and the vertex on the current tree closest to this state is found. A new branch of the tree is then grown from the closest vertex a certain fraction of the distance to the random sample. This process is then repeated, as can be seen in Figure 2-1, until a path in the tree reaches the desired destination. Numerous extensions to RRT have also been developed including RRT* (Karaman and Frazzoli, 2011) and LQR-RRT* (Perez et al., 2012). RRT has been studied in the context of marine path planning such as in the case of navigating AUVs (Tan et al., 2005). It has also been employed in energy-efficient path planning, such as in Rao and Williams (2009), where RRTs are used to find feasible paths for gliders, following which an A* search is used to identify a path from this collection that minimizes an energy-based metric. While RRT has proven to be advantageous in large dimensional state space settings, it should be noted that a key drawback is that the generated path is not globally optimal (LaValle, 2006).

2.1.3 Evolutionary Algorithms

Evolutionary algorithms for path planning have also been developed of which the genetic algorithm (GA) and particle swarm optimization (PSO) algorithm are two of the most popular (Zeng et al., 2016). GA and PSO are population based schemes to solve optimization problems. Specifically, GA aims to model Darwin's genetic selection and natural elimination in the process of biological evolution, while PSO is a heuristic algorithm inspired by the predation and return of bird populations (Li et al., 2018). Both of these approaches have been used in the context of marine path planning. In Alvarez et al. (2004), a genetic algorithm approach on a grid is used to find minimum energy cost paths in dynamic ocean environments. Li et al. (2019) use PSO in the context of path optimization for AUV docking. Cao et al. (2016) address planning for underwater gliders by using a modified 3D Dubins curve to obtain candidate paths and then generating optimal trajectories that minimize energy consumption using a genetic algorithm. Guo et al. (2020) use a variant of the PSO algorithm to compute paths for unmanned surface vehicles that aim to take full advantage of currents while avoiding collisions. It should be noted that while evolutionary algorithms have been successfully applied, a drawback is that they may produce suboptimal solutions within finite time (Zeng et al., 2016).

2.1.4 PDE-Based Algorithms

PDE-based schemes, that operate in continuous spaces, have also seen a proliferation of use. The Fast Marching algorithm is one such example that can be regarded as a continuous version of Dijkstra's algorithm and has relatively recently been studied in the context of underwater path planning (Petres et al., 2005). The level set method has also proven to be a quite versatile approach to path planning. Introduced first in Lolla et al. (2012); Lolla (2012); Lolla et al. (2014b,a), this method allows for the ability of computing globally time optimal paths in continuous, strong and dynamic ocean environments by solving a Hamilton-Jacobi-Bellman PDE. This was followed by further work in extending these schemes to several important applications including underwater autonomous vehicles navigating in three-dimensional flow fields (Kulkarni and Lermusiaux, 2020), path planning in uncertain flow fields (Subramani et al., 2018) or with multiple destinations and vehicles (Ferris et al., 2018), energy-optimal path planning with stochastic optimization (Subramani and Lermusiaux, 2016), energy-optimal path planning using state augmentation (Doshi, 2021) and risk-optimal path planning in stochastic environments (Subramani and Lermusiaux, 2019).

2.2 Applications of Harvest-Time Optimal Path Planning

As addressed in Chapter 1, our proposed rigorous harvest-time optimal path planning would have broad applicability to various marine collection problems. In the following, we delve deeper into several different scenarios and cases where a new accurate path planning methodology would be highly desirable.

2.2.1 Energy Harvesting

As users require autonomous marine vehicles to operate for longer and longer time periods at sea, energy consumption becomes increasingly important to track. The energy requirements for autonomous vehicles are quite diverse, with the bulk of the energy being used for propulsion while additional power – what is known as the "hotel" load – is allotted for non-propulsive purposes for components such as communication systems (Reader et al., 2002). To allow for longer operation at sea, various research advances have been made including the design of more efficient vehicle components and more advanced batteries (Sharkh et al., 2002). Recently, however, there has also been increased attention to design marine vehicles that are no longer restricted to use whatever energy was initially charged into their batteries, but that can instead harvest energy from the environment. The ocean environment is replete with energy sources including solar energy at the water surface, wind energy, wave energy, buoyancy energy, and tidal energy, to name a few (Charlier and Justus, 1993). With this abundance of energy sources, newer marine vehicle designs have looked at ways to collect or harvest this energy so that vehicles can operate at sea for much longer periods of time.

A new design with good potential is the Solar powered AUV (SAUV) (Blidberg and Ageev, 2002). Several different models have been tested and analyzed, all of which allow for extended endurance of the vehicle by having it recharge its onboard energy system by returning to the surface (Ageev, 1995, 2000; Jalbert et al., 2003; Crimmins et al., 2006). Several companies have also designed their own vehicles with energy harvesting capabilities which have started to be used in practical applications. The C-Enduro is a long endurance autonomous surface vehicle fitted with the ability to harvest solar, wind and wave power from its environment. Capable of enduring missions up to 3 months of length, the autonomous vehicle has been used in numerous applications ranging from environmental surveys to military data gathering (Rees, 2013; L3 ASV, 2021; Unmanned Marine Systems, 2013). The AutoNaut is an unmanned surface vessel (USV) whose propulsion is powered entirely by wave energy, while all non-propulsive equipment is operated using solar power (AutoNaut Ltd, 2019). The "Wave Glider", developed by the company Liquid Robotics, is another ocean surface robot powered entirely by renewable sources of energy. Using wave and solar energy, the vehicle can operate for up to 12 months with no fuel, emission, or crew (Liquid Robotics Inc., 2020).

When navigating in marine environments, these new energy harvesting marine vehicles allow for improved energy efficiency through intelligent path planning that optimizes energy use. The strong currents, for one, provide tremendous potential. Autonomous vehicles can save significant energy by leveraging the external flow field by "hitchhiking" on naturally developed currents while reducing the vehicle thrust so that energy can be saved. Energy harvesting capabilities additionally open the door for planning vehicle paths where energy can be efficiently gathered along the way. For instance, in the case where solar energy can be harvested from the environment, an autonomous vehicle can choose to navigate in regions with minimal cloud cover and high solar radiation so as to recharge its power supply and thus increase its endurance.

Figure 2-2: Examples of various autonomous marine vehicles equipped with energy harvesting capabilities. (Top left) The SAUV (Bekey et al., 2006). (Top right) The C-Enduro (Unmanned Marine Systems, 2013). (Bottom left) The Wave Glider (Coxworth, 2013). (Bottom right) The AutoNaut (Ocean News and Technology, 2019).

Similarly, wind powered vehicles can naturally choose to remain in regions with large wind gusts and wave powered vehicles can navigate intelligently to go through regions with higher wave energy.

2.2.2 Ocean Plastic Waste Cleanup

The growth in plastic production has been staggering. From the 1950s when mass production of plastics began (global production of resins and fibers was 2 million metric tons), there has been an approximately 8.4% compound annual growth rate in production, resulting in approximately 380 million metric tons produced in 2015 (Gibb, 2019; Geyer et al., 2017). Cumulatively, this translates to almost 8.3 billion metric tons of resins and fibers manufactured from 1950 through 2015 (Gibb, 2019; Geyer et al., 2017; Ritchie and Roser, 2018). The explosive growth in plastic use of course isn't surprising given its widespread use in numerous applications including

Figure 2-3: (Left) The Manta, a vessel being developed by SeaCleaners, for actively harvesting plastic waste pollution (The SeaCleaners, n.d.). (Right) An autonomous surface vessel called the Wasteshark, developed by Ranmarine Technology, designed for removing floating debris, including plastics, from marine environments (RanMarine Technology, 2021).

packaging, electronics, automotive design and construction, to name a few (Gibb, 2019; Andrady, 2003). The challenge, however, is that of the total plastic waste generated, it is estimated that 9% has been recycled, 12% has been incinerated and a shocking 79% has accumulated in landfills or been released into the natural environment (Rhodes, 2018; Geyer et al., 2017; Ritchie and Roser, 2018). Of this discarded plastic in the natural environment, a large portion ultimately ends up in our seas and oceans (Law, 2017; Ostle et al., 2019), with one of the largest masses present in the North Pacific Ocean – the "Great Pacific Garbage Patch" (Lebreton et al., 2018). This growth of plastic waste in the ocean, moreover, is not slowing, with statistical models projecting that at present rates of production and pollution, by 2050 there will be more plastic than fish in the sea (by mass) (Rhodes, 2018). Once present in the ocean, plastics have devastating consequences on the marine ecosystem. Ingested plastic by wildlife can reduce stomach capacity, reduce growth and cause internal injuries (Sigler, 2014; Plot and Georges, 2010). Plastic entanglement of species with fish nets or other ring shaped materials is also a pressing issue, as this can lead to strangulation and in some cases even drowning (Sigler, 2014).

Tackling the proliferation of ocean plastic pollution requires not only mitigation efforts to reduce our overall consumption of plastic, but also active efforts to collect waste from ocean environments (Vegter et al., 2014; Prata et al., 2019; Borrelle et al., 2020). Several initiatives are in development for plastic waste collection (Rojas, 2018). The Ocean Cleanup, for instance, has developed a surface plastic collection system consisting of a floater that traps advecting plastic (The Ocean Cleanup, 2021). The growth of marine robots and autonomous vessels has resulted in increasing research in their potential use too in this field of active pollution collection. SeaCleaners, a French organization, is developing a vessel – named Manta – powered by renewable energy and equipped with a collection system with the capability to collect up to 10,000 tonnes of plastic per year (The SeaCleaners, n.d.). Ranmarine Technology is a company designing Wasteshark, a lightweight autonomous vessel inspired by whale sharks. It is designed to harvest plastic, micro-plastics, alien vegetation and floating debris, while simultaneously performing data collection including temperature and water quality tracking (RanMarine Technology, 2021). There has additionally been increasing amounts of research in recent years in the academic community in developing autonomous collection vessels capable of detecting, classifying and harvesting plastic pollution (Kong et al., 2020; Flores et al., 2020). All in all, the development of autonomous marine vessels in tackling this problem makes it evident that effective path planning that can optimize for the amount of plastic waste harvested from the environment would be of great use. Of course, such planning would require inputs from accurate plastic transport modeling and ocean modeling, both of which are making progress, e.g., Lermusiaux et al. (2019).

2.2.3 Offshore Aquaculture

To address the growing food demands of the world, offshore aquaculture use has spread and seen increasing development (Soto and Wurmann, 2019; Garlock et al., 2020; Subasinghe et al., 2009). Also often referred to as open water aquaculture or open ocean aquaculture, this is an approach to growing marine species by positioning farms in deeper waters further from land (Benetti et al., 2010; Gentry et al., 2017). An exciting new development in this field is research into novel, next generation technology on autonomously moving fish farms. The principle idea here would be for

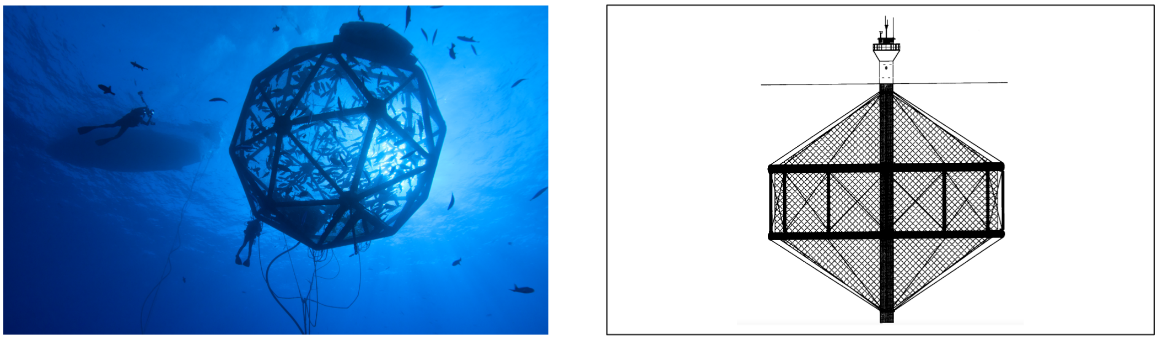


Figure 2-4: (Left) An offshore fish farm operating in the Gulf of Mexico (Schwaab, 2020). (Right) A conceptual design for the Ocean Drifter, a novel system for an offshore fish farm that continuously moves in the open ocean (Goudey, 1998).

such farms to be launched with lab-bred baby fish and then autonomously navigate along ocean currents to ultimately arrive at a destination with fully mature animals ready for market (Mann, 2004). Governments are actively funding research labs to help move this idea from the drawing board to reality (Cohen, 2008; Goudey, 1998). Furthermore, our harvest-time path planning methodology is directly applicable to these autonomous farms in helping plan paths that not only allow to reach the destination time optimally, but also optimize nutrient harvesting so as to maximize fish growth or to minimize risks of diseases by avoiding specific water regions.

2.2.4 Additional Applications

In addition to the aforementioned applications, there are numerous other situations where accurate optimal harvesting in dynamic flows could have applicability in the near or long-term future as technology progresses (Doshi et al., 2019; Bhabra et al., 2020; Hoteit et al., 2021).

Oil Spill Cleanup The ubiquitous use of oil and petroleum products has made oil spills a pressing issue to address due to their damaging effect on the environment. In marine environments, these spills can occur from shipping incidents or problems on offshore rigs (Fingas, 2016; Doerffer, 2013). While numerous cleanup strategies have been developed for containment and recovery of oil on water (Fingas, 2012; Bayat et al., 2005), autonomous cleanup vehicles and optimal harvesting path planning can

Figure 2-5: (Left) A ship actively monitoring oil from a spill (Scammell, 2018). Sargassum seaweed off the coast of St. Martin (Leatherman, 2021).

also be of use to help address problems in the future.

Sargassum Cleanup Sargassum, a form of macroalgae, has in recent years inundated coastal areas in the Caribbean, disturbing the tourism industries of affected countries (Yong, 2019; Louime et al., 2017; Schell et al., 2015). Autonomous optimal harvesting vehicles would be quite beneficial in the overall cleanup strategy by addressing the collection of this sargassum in the ocean prior to reaching land (SOS Carbon, 2021).

Chapter 3

Problem Statement

In this chapter, we outline in detail the specific path planning problem that is the central focus of this work.

3.1 Problem Parameters

Consider a vehicle assigned with the task of navigating in a physical domain $\Omega \subseteq \mathbb{R}^d$ from a start point $\boldsymbol{x}_s \in \mathbb{R}^d$ to a defined target point $\boldsymbol{x}_f \in \mathbb{R}^d$. The vehicle operates in an environment with a background dynamic flow $\boldsymbol{V}(\boldsymbol{x}, t) : \Omega \times [0, \infty) \to \mathbb{R}^d$. Furthermore, as it traverses to the target, the vehicle has two controls at its disposal. The first is the time-dependent relative speed function $F(t)$ given by $F(t) : [0, \infty) \to [0, F_{max}],\ F_{max} \in \mathbb{R}_{>0}$. The second is the sequence of headings $\hat{\boldsymbol{h}}(t)$ that it can choose to move in, which also is in general time-dependent. We represent these headings with a unit vector:

$$\hat{\boldsymbol{h}}(t) : [0, \infty) \to \mathbb{R}^d \ \text{ such that } \ \left\| \hat{\boldsymbol{h}}(t) \right\| = 1,\ \forall t. \tag{3.1}$$

Using only these controls, the vehicle's primary goal is to plan optimal paths that optimize a user specified metric, e.g., travel time, energy usage, and here harvest and time optimization.

In addition to its physical position in space, we consider the case where the vehicle may have additional state variables of interest related to harvesting. For instance, in

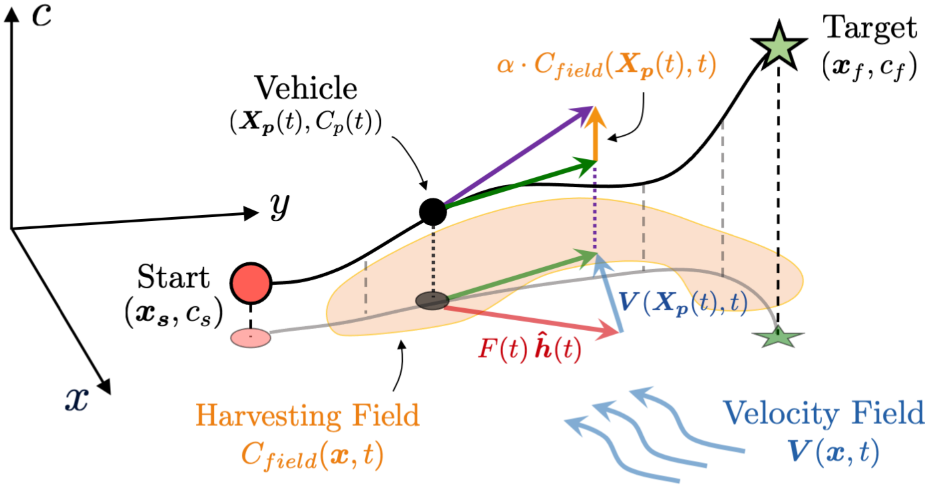


Figure 3-1: A schematic showing the primary components of the harvest-time optimal path planning problem that is the focus of this work. This schematic depicts the particular case of a vehicle navigating in a 2D physical domain tasked with collecting a tracer field (the harvesting dynamics is given by Equation (3.4)). Shown is the trajectory of the vehicle through the augmented state space (x, y, c) consisting the physical position $\boldsymbol{x}$ and harvest state c (the amount of the tracer field harvested/collected).

the case of energy harvesting, the energy of the vehicle would be an additional state while for the case where a vehicle may be collecting some passively advecting field, the amount of the field collected would be an additional state to track. To allow for complete generality, we consider the case where a vehicle is tasked with harvesting a number d_h of dynamic fields, and we denote the state of the harvested fields for the vehicle with a vector $\boldsymbol{c} \in \mathbb{R}^{d_h}$. Moreover, we denote the harvest state of the vehicle at the start point with $\boldsymbol{c}_s$.

Altogether, we have an augmented vehicle state space $\boldsymbol{x}^a = [\boldsymbol{x}, \boldsymbol{c}]^T$, a subset of $\mathbb{R}^{d+d_h}$, consisting of the physical position $\boldsymbol{x}$ and harvest state $\boldsymbol{c}$. Let a general continuous trajectory of the position and harvesting states of the vehicle through the augmented state space be given as $\boldsymbol{X}_p(t)$ and $\boldsymbol{C}_p(t)$. The full dynamics of the vehicle's state is assumed to be of the following form:

$$\frac{\mathrm{d}\boldsymbol{X}_p}{\mathrm{d}t} = \boldsymbol{V}\left(\boldsymbol{X}_p(t), t\right) + F(t)\,\hat{\boldsymbol{h}}(t), \tag{3.2a}$$

$$\frac{\mathrm{d}C_{p,i}}{\mathrm{d}t} = \mathcal{H}_i(\boldsymbol{X}_p(t), \boldsymbol{C}_p(t), F(t), \hat{\boldsymbol{h}}(t), t), \quad i = 1, ..., d_h. \tag{3.2b}$$

Equation (3.2a) governs the motion of the vehicle through the physical space. The vehicle, in this work, is modeled as a point particle and its trajectory is governed by its relative motion due to its controls – the speed and heading – and advection from the external flow field.

Equation (3.2b) corresponds to the general form of the dynamics of the harvesting states considered in this work. In particular, the evolution of each of the harvesting states is assumed to be given by an ODE which may additionally be coupled to other harvesting fields or to the physical state, or may be externally forced by other variables. The class of problems contained in this formulation is thus quite wide. Specifically, some of the common cases considered in this work are given as follows:

Energy Harvesting In the case of path planning with energy considerations, the harvesting state of interest would be e – the energy of the vehicle. In this case, the energy dynamics of the vehicle would be dictated by the first law of thermodynamics, where the vehicle would expend energy as it uses power for various needs – most notably propulsion – however it may also have the capability of collecting energy from the environment through, for instance, solar panels. Denote the spatially varying – and possibly dynamic – energy input field as $\dot{Q}(\boldsymbol{x}, t) : \Omega \times [0, \infty) \to \mathbb{R}$. The work expended by the vehicle can be assumed to be dominated by energy used for propulsion and would be a function of the thrust of the vehicle in the general form $\dot{W}(F)$. Several models exist for the thrust power function including $\dot{W}(t) \propto F(t)$ for the case of constant drag-force, $\dot{W}(t) \propto F(t)^2$ for linear drag, or $\dot{W}(t) \propto F(t)^3$ for quadratic drag (Subramani, 2014). Consider an arbitrary trajectory of the vehicle through the physical space given by $\boldsymbol{X}_p(t)$. Furthermore, let $E_p(t)$ denote the continuous trajectory through the energy state space for the vehicle. The energy evolution would then be governed by the following equation

$$\frac{\mathrm{d}E_p}{\mathrm{d}t} = \dot{Q}(\boldsymbol{X}_p(t), t) - \dot{W}(F(t)), \tag{3.3}$$

where it is clear that the dynamics of the harvesting state – the vehicle energy – is indeed of the form of Equation (3.2b).

Collecting a Tracer Field Consider an arbitrary tracer field in the domain given by $C_{field}(\boldsymbol{x}, t) : \Omega \times [0, \infty) \to \mathbb{R}$. This field's time evolution is assumed to be governed by its transport by advection and diffusion due to the surrounding flow field as would be the case, for instance, with a plastic waste field. If the harvesting state of the vehicle consisted of the amount of the background field collected, a natural model for this collection process would be to model the amount collected at any given point in space to be proportional to the amount of the background field present. Consider an arbitrary trajectory of the vehicle through the physical space given by $\boldsymbol{X}_p(t)$. Furthermore, let $C_p(t)$ denote the continuous trajectory through the "collection" state space. The collection dynamics would then be of the form

$$\frac{\mathrm{d}C_p}{\mathrm{d}t} = \alpha \cdot C_{field}(\boldsymbol{X}_p(t), t), \tag{3.4}$$

where $\alpha \in \mathbb{R}_{>0}$ is a constant of proportionality. It can be noted that Equation (3.4) is once again of the same form as Equation (3.2b). It is important to note that the model of the collection dynamics in Equation (3.4) does not account for the influence of the vehicle on the field itself. This model would thus be valid in the limit where the vehicle is so small relative to the length scales of the background field that it does not have a noticeable influence on it.

With the dynamics of the vehicle and its harvesting state(s) specified, we next address the specific optimal path planning problem that is the principal focus of this work.

3.2 Multi-Objective Harvest-Time Optimal Path Planning

The optimal path planning problem of interest in this work is a multi-objective one involving optimizing not only the final harvested state of the vehicle but also factoring in the time taken to traverse from the start point to the destination. Specifically, we consider paths obtained by solving the following optimization problem:

$$\begin{aligned}
\underset{F(\cdot),\,\hat{\boldsymbol{h}}(\cdot)}{\text{minimize}} \quad & \left[t_f, -C_{p,1}(t_f), -C_{p,2}(t_f), ..., -C_{p,d_h}(t_f)\right] \\
\text{subject to} \quad & \boldsymbol{X}_p(0) = \boldsymbol{x}_s \\
& \boldsymbol{C}_p(0) = \boldsymbol{c}_s \\
& \boldsymbol{X}_p(t_f) = \boldsymbol{x}_f \\
& \boldsymbol{C}_p(t) > 0, \quad \forall t \\
& \frac{\mathrm{d}\boldsymbol{X}_p}{\mathrm{d}t} = \boldsymbol{V}\left(\boldsymbol{X}_p(t), t\right) + F(t)\,\hat{\boldsymbol{h}}(t) \\
& \frac{\mathrm{d}C_{p,i}}{\mathrm{d}t} = \mathcal{H}_i(\boldsymbol{X}_p(t), \boldsymbol{C}_p(t), F(t), \hat{\boldsymbol{h}}(t), t), \quad i = 1, ..., d_h
\end{aligned} \quad . \tag{3.5}$$

The optimization problem in Equation (3.5) involves *both* minimizing the time taken to arrive at the destination and maximizing the vehicle's harvesting states at the destination. Mathematically equivalent to this maximization is the minimization of the negative of the final harvesting states as shown in Equation (3.5).

Multi-objective optimization is a rich field with high relevance to numerous industries such as engineering and economics (Ehrgott, 2008). The key challenge in such an optimization problem is that, in the vast majority of cases, no globally optimal solution exists that simultaneously satisfies all the competing objectives. In these cases, one looks for what are known as *Pareto optimal* solutions. Consider the general multi-objective optimization problem

$$\begin{aligned}
\underset{\boldsymbol{x}}{\text{minimize}} \quad & [f_1(\boldsymbol{x}), f_2(\boldsymbol{x})..., f_k(\boldsymbol{x})] \\
\text{subject to} \quad & \boldsymbol{x} \in \mathcal{S}
\end{aligned} \quad , \tag{3.6}$$

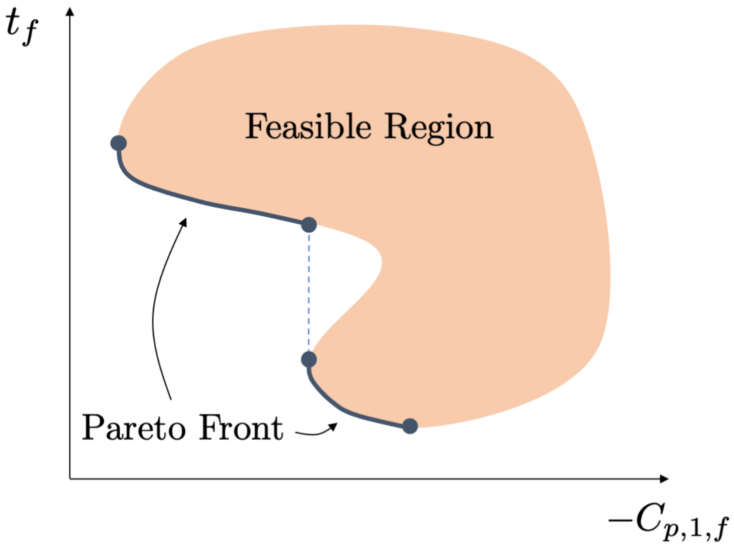


Figure 3-2: A schematic illustrating the concept of the Pareto front in multi-objective optimization. Shown here is a hypothetical solution to the optimization problem given by Equation (3.5) for the case of only one harvest state (we define $C_{p,1,f} \equiv C_{p,1}(t_f)$). Pareto solutions are those whose performance in one objective cannot be improved without worsening the performance on another objective. Shown in this figure is a set of feasible solutions (in orange) for the two objectives we wish to minimize. Indicated in blue is the set of Pareto optimal solutions (the Pareto front).

where we search for the optimal decision vector $\boldsymbol{x}$ belonging to the feasible set $\mathcal{S}$. We then have the following formal definition of Pareto optimality from Miettinen (2012):

Definition 3.2.1. *A decision vector $\boldsymbol{x}^* \in \mathcal{S}$ is Pareto optimal if there does not exist another decision vector $\boldsymbol{x} \in \mathcal{S}$ such that $f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}^*)$ for all $i = 1, ..., k$ and $f_j(\boldsymbol{x}) < f_j(\boldsymbol{x}^*)$ for at least one index j.*

In other words, a Pareto optimal solution is one which cannot be improved upon in any objective without degrading the performance of at least one other objective. Collectively, these Pareto optimal solutions form a set Θ_p known as the *Pareto front*. In the past decades, numerous algorithms have been formulated with the central goal of searching for the Pareto front (Marler and Arora, 2004; Xiujuan and Zhongke, 2004).

In this work, we look to compute the set of Pareto optimal solutions – the Pareto front – for the optimization problem given in Equation (3.5). That is, we aim to determine optimal trajectories of the vehicle through the full *augmented* state space consisting of the physical and harvesting states. In the next chapters, we show how this can be completed efficiently using Hamilton-Jacobi theory for reachability and our efficient numerical schemes for computing their solutions.

3.3 Assumptions

We now summarize the key assumptions used throughout this work.

- In all cases, the flow field $\boldsymbol{V}(\boldsymbol{x}, t)$ as well as any harvesting dynamics equations are assumed to be deterministic and exactly known. That is, we do not consider the impact of how uncertainties in the dynamics equations would affect the final solution. Very preliminary results from an approach to handle uncertainties in the environment that we can build on in future work are shown in Chapter 7.
- We consider the flow field and the path planning at scales (e.g., grid resolutions of 100 meters and more, with ocean domains of hundreds to thousands of kilometers) that are much larger than the size of the vehicles (e.g., a few meters). Without any loss of generality, the vehicle is then modeled to behave as a point mass that exhibits no inertia. As a result, the interaction between the flow field and vehicle is solely kinematic.
- We search for continuous optimal trajectories of the vehicle through the full state space.
- In all cases, the influence of the vehicle on the field it is harvesting is not considered. In many cases, such as energy harvesting, this is not an issue (e.g., the vehicle has no influence on the amount of solar energy remaining after it has harvested some). In other cases, such as for collecting tracers or pollutants (e.g., plastic waste), this assumption is valid in the limit of the vehicle being so small relative to the length scales of the field that it has a negligible influence on it. We expand upon this assumption and its implications further in Chapter 6 when we consider different applications of our methodology.

Chapter 4

Theory: Multi-Objective Harvest-Time Optimal Path Planning

In this chapter, we present the theory behind our harvest-time optimal path planning algorithm. We start first by diving into the fundamentals of optimal control theory and Hamilton-Jacobi-Bellman equations, as this forms the base of our approach. We then review Hamilton-Jacobi forward reachability, following which we make the connection with how reachability can be used in the context of multi-objective harvest-time optimal path planning.

In Sections 4.1 to 4.3, we review optimal control theory, the method of dynamic programming, the concept of forward reachability and the theory of viscosity solutions of Hamilton-Jacobi equations, respectively. The material here follows the presentations of classic papers and books in these fields, and we include derivations from these works for completeness. In Section 4.4, we apply these concepts to formulate a novel approach to solve our multi-objective harvest-time optimal path planning problem. We derive the relevant PDEs that form the base of our method and prove, in particular, how forward reachability can be used to form the full Pareto front of optimal solutions. This section contains the main contribution of this work. We use our methodology, presented in Section 4.4, to derive the governing equations for exact harvest-time optimal path planning for certain applications in Section 4.5. Specifically, we consider the case of how globally optimal paths can be computed for energy-time optimal path planning and in the case of harvest-time optimal path planning when harvesting a tracer field.

4.1 Optimal Control

We start first in this section by introducing fundamental concepts from the rich field of optimal control theory. The key idea in optimal control is to pose the control design process as an optimization problem, where the central objective is to determine a control signal that will cause some process to satisfy its physical constraints while also minimizing some performance criterion (i.e. cost function) (Kirk, 2004; Tedrake, 2021).

First, we define the optimal control problem, explaining its various components (Kirk, 2004; Brogan, 1991; Bardi and Capuzzo-Dolcetta, 2008; Evans, 2010). We then present the powerful technique of solving these optimization problems using the method of dynamic programming (Kirk, 2004; Bertsekas, 2000).

4.1.1 Statement of the Optimal Control Problem

We start by defining the optimal control problem, following primarily the presentation of (Kirk, 2004). Our focus in this section is primarily on optimal control of continuous dynamical systems, and we refer the reader to (Kirk, 2004) for information specific to their discrete counterparts.

The Dynamical System

The first component of any optimal control problem is the dynamical system. We consider, in particular, a system state $\boldsymbol{x} \in \mathbb{R}^n$ governed by an ordinary differential equation of the form

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t), \quad \boldsymbol{x}(t_0) = \boldsymbol{x}_0, \quad t \in [t_0, t_f], \tag{4.1}$$

where $\boldsymbol{u}(\cdot)$, the control, is from a set $\mathbb{U}$ of measurable function of $t \in [t_0, t_f]$ with values in $\mathcal{U}$:

$$\mathbb{U} = \{\boldsymbol{u} : [t_0, t_f] \to \mathcal{U} \mid \boldsymbol{u}(\cdot) \text{ is measurable}\}. \tag{4.2}$$

$\mathcal{U}$ is typically referred to as the control space and is often a closed bounded set in $\mathbb{R}^{n_u}$, where n_u corresponds to the number of control inputs. The system dynamics $\boldsymbol{f} : \mathbb{R}^n \times \mathcal{U} \times \mathbb{R} \to \mathbb{R}^n$ is further assumed to be continuously differentiable in the variables $\boldsymbol{x}$, $\boldsymbol{u}$ and t, for all $t \in [t_0, t_f]$ and all $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{u} \in \mathcal{U}$. With these smoothness assumptions on the dynamics, we have that there is a unique solution to the initial value problem in Equation (4.1) for any control sequence $\boldsymbol{u}(\cdot)$ (Kurzhanski and Varaiya, 2001). We denote such a trajectory of the system from an initial state $\boldsymbol{x}_0$ at time t_0 using the control sequence $\boldsymbol{u}(\cdot)$ as $\boldsymbol{x}(t; t_0, \boldsymbol{x}_0, \boldsymbol{u}(\cdot))$.

Remark 1. *In the proceeding presentation, we will use $\boldsymbol{u}(\cdot)$ to denote the control function when the domain of the function can be easily inferred. In cases where we want to explicitly highlight what segment (i.e. part of the domain) of the control function is being evaluated, we will use the notation $\underset{t_0 \le s \le t_f}{\boldsymbol{u}(s)}$. The same convention will be used in analyzing the state trajectory $\boldsymbol{x}(\cdot)$.*

The Performance Measure

The next key component of the optimal control problem is the performance measure (also commonly referred to as the cost function). The performance measure is created by a designer and is used as a quantitative metric to evaluate the performance of a system. While numerous possible measures are available, dramatic simplifications arise when using those that have an additive cost structure, as these often yield recursive solutions (Tedrake, 2021). As a result of this, one of the most commonly used cost functions or performance measures (and the one also used in this work) is of the form (Kirk, 2004):

$$J(\boldsymbol{x}_0, t_0, \underset{t_0 \le \tau \le t_f}{\boldsymbol{u}(\tau)}) = h(\boldsymbol{x}(t_f), t_f) + \int_{t_0}^{t_f} g(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), \tau) d\tau. \tag{4.3}$$

In Equation (4.3), $J(\boldsymbol{x}_0, t_0, \underset{t_0 \le \tau \le t_f}{\boldsymbol{u}(\tau)})$ takes as input the start state, start time and an admissible control sequence, and ultimately outputs a scalar. Furthermore, $\boldsymbol{x}(\cdot)$ on the right hand side of Equation (4.3) is implied to be $\boldsymbol{x}(t) = \boldsymbol{x}(t; t_0, \boldsymbol{x}_0, \boldsymbol{u}(\cdot))$ – that is, it is the state trajectory that results from starting at the initial state $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$ and using the specified control signal $\boldsymbol{u}(\cdot)$ defined for times $t \in [t_0, t_f]$. Additionally, the performance measure is composed of $h : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$, a scalar function known as the *terminal cost*, and $g : \mathbb{R}^n \times \boldsymbol{\mathcal{U}} \times \mathbb{R} \to \mathbb{R}$, a scalar function known as the *running cost* (also sometimes called the instantaneous cost) (Tedrake, 2021).

The Optimal Control Problem

With the dynamical system and performance measure defined, we can now present the central objective of optimal control (Kirk, 2004):

Problem 4.1.1. *Find an admissible control function* $\boldsymbol{u}^*(\cdot)$ *which causes the system governed by*

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t), \quad \boldsymbol{x}(t_0) = \boldsymbol{x}_0, \quad t \in [t_0, t_f],$$

to follow a trajectory $\boldsymbol{x}^*(\cdot)$ *that minimizes the performance measure (or cost function)*

$$J(\boldsymbol{x}_0, t_0, \underset{t_0 \le \tau \le t_f}{\boldsymbol{u}(\tau)}) = h(\boldsymbol{x}(t_f), t_f) + \int_{t_0}^{t_f} g(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), \tau) d\tau.$$

The resulting $\boldsymbol{u}^*(\cdot)$ *is called the optimal control and* $\boldsymbol{x}^*(\cdot)$ *is referred to as the optimal state trajectory*

Various types of optimal control functions $\boldsymbol{u}^*(\cdot)$ exist. In particular, we have the following definitions from (Kirk, 2004):

Definition 4.1.1. *If a functional relationship of the form*

$$\boldsymbol{u}^*(t) = \boldsymbol{\pi}(\boldsymbol{x}(t), t), \tag{4.4}$$

can be found for the optimal control at time t*, then the function* $\boldsymbol{\pi}$ *is called the optimal control law or the optimal policy (the terms optimal feedback control and closed-loop* *optimal control are also used often in literature).*

Definition 4.1.2. *If the optimal control is determined as a function of time for a specified initial state value, that is,*

$$\boldsymbol{u}^*(t) = \boldsymbol{e}(\boldsymbol{x}(t_0), t), \tag{4.5}$$

then the optimal control is said to be in open-loop form.

In essence, an optimal policy or closed-loop optimal control is able to determine the optimal control at any time t and for any state $\boldsymbol{x}$. In the open-loop case, the control is optimal only for a particular initial state value.

It is instructive to see a few examples of different optimal control problems and how their corresponding optimization problem is formulated. The following are two examples:

> **Example: Minimum-Time Problems**
>
> Consider the problem of a system at an initial state $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$ with the goal of reaching a target set $\mathcal{S}$ in minimum time. This can be formulated as an optimal control problem using the performance measure
>
> $$J(\boldsymbol{x}_0, t_0, \underset{t_0 \leq \tau \leq t_f}{\boldsymbol{u}(\tau)}) = \int_{t_0}^{t_f} d\tau, \tag{4.6}$$
>
> where t_f is the first instant in time when $\boldsymbol{x}(t_f) \in \mathcal{S}$ (Kirk, 2004).
>
> **Example: Tracking Problems**
>
> Consider the problem of having to keep a system state $\boldsymbol{x}(t)$, starting at $\boldsymbol{x}_0$ at time t_0, as close as possible to a desired trajectory through the state space $\boldsymbol{r}(t)$ over the time interval $[t_0, t_f]$. This can be formulated as an optimal control problem using the cost function
>
> $$J(\boldsymbol{x}_0, t_0, \underset{t_0 \leq \tau \leq t_f}{\boldsymbol{u}(\tau)}) = \int_{t_0}^{t_f} \|\boldsymbol{x}(\tau) - \boldsymbol{r}(\tau)\|^2_{\boldsymbol{Q}(\tau)}\, d\tau, \tag{4.7}$$

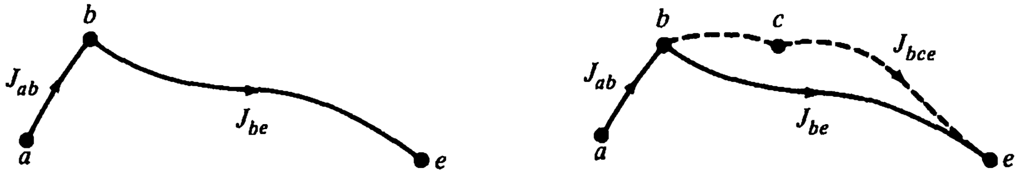


Figure 4-1: A schematic (adapted from (Kirk, 2004)) showing a visual representation of the principle of optimality. If $a - b - e$ is the optimal path in the state space, then the principle of optimality states that $b - e$ must be the optimal path from b to e.

where $\|\cdot\|_{\boldsymbol{Q}(\tau)}$ is a weighted norm and $\boldsymbol{Q}(t)$ is a real and symmetric positive semi-definite matrix for all $t \in [t_0, t_f]$ (Kirk, 2004).

4.1.2 Dynamic Programming

Once the optimal control problem has been formulated, the next task is the most important: solving the optimization problem for the optimal control function. Two primary approaches exist to accomplish this. The first is a variational approach, following from the minimum principle of Pontryagin, that leads to a nonlinear two-point boundary value problem to be solved (Kirk, 2004). The second is the method of dynamic programming developed by Richard Bellman (Bellman and Kalaba, 1965; Bellman, 1966). In this section, we introduce the method of dynamic programming, as this forms the basis of our path planning methodology.

The Principle of Optimality

Dynamic programming is a method for solving problems with a multistage decision process. The basis of this method is Bellman's principle of optimality which states: "An optimal policy has the property that whatever the initial state and the initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision" (Brogan, 1991).

Figure 4-1, adapted from (Kirk, 2004), is a sketch that gives a good visual representation of the principle of optimality. Here, trajectories through a state space are shown with the associated cost of each segment also listed. Assume that the optimal path is given by segment $a - b - e$ which yields the minimum cost J^*_{ae} for going from a to e given by

$$J^*_{ae} = J_{ab} + J_{be}.$$

It follows from the principle of optimality then that if this is the optimal path for $a - e$, then $b - e$ is the optimal path from b to e. That is, there is no other possible path, such as $b - c - e$, which produces a cost $J_{bce} < J_{be}$ (Kirk, 2004).

The principle of optimality can ultimately be used in the process of decision making. Figure 4-2, adapted from (Kirk, 2004), provides a good illustration of this concept. Consider the problem of planning an optimal trajectory through the state space for a process starting at state b with the requirement of reaching the terminal state f. The path segments resulting from all different possible decisions (i.e. controls) from b are shown in the dashed lines connecting b to the corresponding subsequent states (c, d and e). Assume further that the optimal trajectories from each of these intermediary states to the terminal state f are known and have associated costs of J^*_{cf}, J^*_{df} and J^*_{ef}. It follows then, from the principle of optimality, that if $b - c$ formed the initial segment of the optimal path for $b - f$, then $c - f$ constitutes the terminal segment of this path. The same reasoning applies if the optimal path were to go through state d or state e. The optimal trajectory from b to f is then found by considering the cost of each of these candidate paths

$$\begin{aligned} C^*_{bcf} &= J_{bc} + J^*_{cf} \\ C^*_{bdf} &= J_{bd} + J^*_{df} \\ C^*_{bef} &= J_{be} + J^*_{ef}, \end{aligned} \tag{4.8}$$

and then choosing the path with the minimum of these costs. This resulting path furthermore determines the optimal decision (i.e. control) to take at the state b (Kirk, 2004).

Dynamic programming ultimately builds upon this decision-making process by applying it to sequences of decisions which in turn determine the optimal policy and

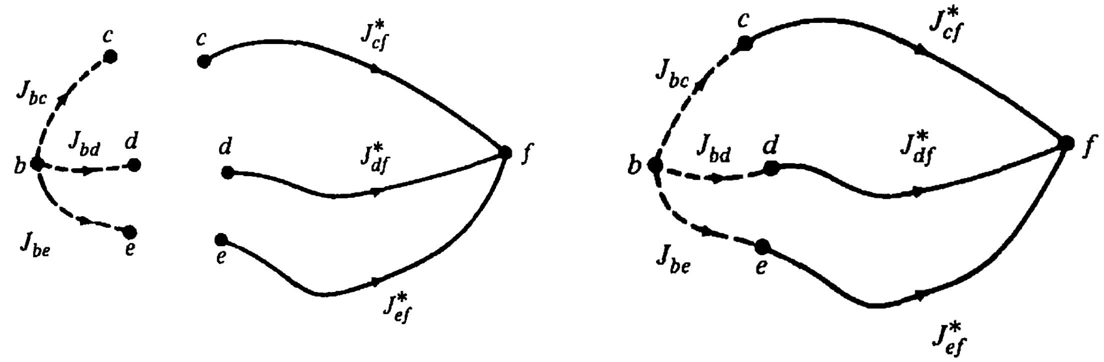


Figure 4-2: A schematic (adapted from (Kirk, 2004)) showing how the principle optimality can be used in the context of decision making for multistage design processes.

trajectory for the process (Kirk, 2004).

The Hamilton-Jacobi-Bellman Equation

We now to turn to using dynamic programming to solve the general optimal control problem given by Problem 4.1.1. To solve this specific problem, we consider the more general problem of finding the optimal controls to minimize the cost function

$$J(\boldsymbol{x}, t, \underset{t \le \tau \le t_f}{\boldsymbol{u}(\tau)}) = h(\boldsymbol{x}(t_f), t_f) + \int_t^{t_f} g(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), \tau) d\tau. \tag{4.9}$$

Equation (4.9) considers the problem of finding the optimal trajectory starting from any arbitrary state $\boldsymbol{x}$ and at any arbitrary time $t \in [t_0, t_f]$. We consider in particular this broader problem since embedded in this larger problem is our specific optimal control problem of interest (of computing trajectories starting from $\boldsymbol{x}_0$ at time t_0).

To solve for the optimal trajectory from any $\boldsymbol{x}$ and $t \in [t_0, t_f]$ in Equation (4.9), we turn our attention to a critical component known as the *value function* which is defined as (Kirk, 2004):

$$\begin{aligned} J^*(\boldsymbol{x}, t) &= \min_{\substack{\boldsymbol{u}(\tau) \\ t \le \tau \le t_f}} \left\{ J(\boldsymbol{x}, t, \underset{t \le \tau \le t_f}{\boldsymbol{u}(\tau)}) \right\}, \\ &= \min_{\substack{\boldsymbol{u}(\tau) \\ t \le \tau \le t_f}} \left\{ h(\boldsymbol{x}(t_f), t_f) + \int_t^{t_f} g(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), \tau) d\tau \right\}. \end{aligned} \tag{4.10}$$

The value function holds the minimum possible cost that can be attained for our system of interest starting at any arbitrary state $\boldsymbol{x}$ and time $t \in [t_0, t_f]$. As we will see, the value function $J^*(\boldsymbol{x}, t)$ can be efficiently computed as a solution to a PDE, and this solution can then be ultimately used to determine the optimal control policy (Evans, 2010).

Consider an arbitrary $\boldsymbol{x}$ and $t \in [t_0, t_f]$. Subdividing intervals, the value function can be written as:

$$\begin{aligned} J^*(\boldsymbol{x}, t) = \min_{\substack{\boldsymbol{u}(\tau) \\ t \leq \tau \leq t_f}} \Bigg\{ \int_t^{t+\Delta t} & g(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), \tau) d\tau \\ & + \int_{t+\Delta t}^{t_f} g(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), \tau) d\tau + h(\boldsymbol{x}(t_f), t_f) \Bigg\}. \end{aligned} \tag{4.11}$$

Using now the principle of optimality, Equation (4.11) can be equivalently expressed as

$$J^*(\boldsymbol{x}, t) = \min_{\substack{\boldsymbol{u}(\tau) \\ t \leq \tau \leq t+\Delta t}} \Bigg\{ \int_t^{t+\Delta t} g(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), \tau) d\tau + J^*(\boldsymbol{x}', t + \Delta t) \Bigg\}. \tag{4.12}$$

Here, the state $\boldsymbol{x}'$ is that which is reached at time $t + \Delta t$ (i.e. $\boldsymbol{x}(t + \Delta t) = \boldsymbol{x}'$) when starting from $\boldsymbol{x}(t) = \boldsymbol{x}$ at time t using the control $\underset{t \leq \tau \leq t+\Delta t}{\boldsymbol{u}(\tau)}$. In what follows, denote by $\boldsymbol{u}$ the instantaneous control at time t (i.e. $\boldsymbol{u}(t) = \boldsymbol{u}$). Now, assuming that the value function J^* is smooth enough that second order partial derivatives exist and are bounded, we can express $J^*(\boldsymbol{x}', t + \Delta t)$ using a Taylor series about the point $(\boldsymbol{x}, t)$:

$$\begin{aligned} J^*(\boldsymbol{x}', t + \Delta t) = J^*(\boldsymbol{x}, t) & + \left[\frac{\partial J^*}{\partial t} \bigg|_{(\boldsymbol{x}, t)} \right] \cdot \Delta t \\ & + \left[\frac{\partial J^*}{\partial \boldsymbol{x}} \bigg|_{(\boldsymbol{x}, t)} \right]^T \cdot \Big[\boldsymbol{x}' - \boldsymbol{x} \Big] \\ & + \text{Higher Order Terms}. \end{aligned} \tag{4.13}$$

The dynamics equation additionally allows for the ability to express $\boldsymbol{x}'$ as:

$$\boldsymbol{x}' = \boldsymbol{x} + \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \cdot \Delta t + \mathcal{O}(\Delta t^2), \tag{4.14}$$

which can be substituted into Equation (4.13) to yield:

$$\begin{aligned} J^*(\boldsymbol{x}', t + \Delta t) = J^*(\boldsymbol{x}, t) &+ \left[\frac{\partial J^*}{\partial t}\bigg|_{(\boldsymbol{x},t)} \right] \cdot \Delta t \\ &+ \left[\frac{\partial J^*}{\partial \boldsymbol{x}}\bigg|_{(\boldsymbol{x},t)} \right]^T \cdot \Big[\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \Big] \cdot \Delta t \\ &+ \mathcal{O}(\Delta t^2). \end{aligned} \tag{4.15}$$

Furthermore, the integral

$$\int_t^{t+\Delta t} g(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), \tau) d\tau, \tag{4.16}$$

can be approximated using a Taylor series as well to give

$$\int_t^{t+\Delta t} g(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), \tau) d\tau = g(\boldsymbol{x}, \boldsymbol{u}, t) \cdot \Delta t + \mathcal{O}(\Delta t^2). \tag{4.17}$$

Substituting the expansions from Equation (4.15) and (4.17) into Equation (4.12), we obtain:

$$\begin{aligned} J^*(\boldsymbol{x}, t) = \min_{\substack{\boldsymbol{u}(\tau) \\ t \le \tau \le t + \Delta t}} \Bigg\{ & g(\boldsymbol{x}, \boldsymbol{u}, t) \cdot \Delta t + J^*(\boldsymbol{x}, t) + \left[\frac{\partial J^*}{\partial t}\bigg|_{(\boldsymbol{x},t)} \right] \cdot \Delta t \\ &+ \left[\frac{\partial J^*}{\partial \boldsymbol{x}}\bigg|_{(\boldsymbol{x},t)} \right]^T \cdot \Big[\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \Big] \cdot \Delta t + \mathcal{O}(\Delta t^2) \Bigg\}. \end{aligned} \tag{4.18}$$

$J^*(\boldsymbol{x}, t)$ on the right hand side of Equation (4.18) does not depend on the controls $\boldsymbol{u}(\cdot)$, which allows for it to be pulled out of the minimization and cancelling with the identical term on the left hand side:

$$0 = \min_{\substack{\boldsymbol{u}(\tau) \\ t \le \tau \le t+\Delta t}} \Bigg\{ g(\boldsymbol{x}, \boldsymbol{u}, t) \cdot \Delta t + \left[\frac{\partial J^*}{\partial t}\bigg|_{(\boldsymbol{x},t)} \right] \cdot \Delta t \\ + \left[\frac{\partial J^*}{\partial \boldsymbol{x}}\bigg|_{(\boldsymbol{x},t)} \right]^T \cdot \Big[\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \Big] \cdot \Delta t + \mathcal{O}(\Delta t^2) \Bigg\}. \tag{4.19}$$

Dividing through by Δt and taking the limit as $\Delta t \to 0$ yields:

$$0 = \frac{\partial J^*}{\partial t}\bigg|_{(\boldsymbol{x},t)} + \min_{\boldsymbol{u}} \Bigg\{ g(\boldsymbol{x}, \boldsymbol{u}, t) + \left[\frac{\partial J^*}{\partial \boldsymbol{x}}\bigg|_{(\boldsymbol{x},t)} \right]^T \cdot \Big[\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \Big] \Bigg\}. \tag{4.20}$$

In Equation (4.20), note that in the limit we are left with minimizing over the instantaneous control $\boldsymbol{u}$ at time t. Dropping explicitly expressing the point of evaluation of the partial derivatives of the value function for brevity of notation, we can express Equation (4.20) compactly as:

$$\frac{\partial J^*}{\partial t} + \min_{\boldsymbol{u}} \Bigg\{ g(\boldsymbol{x}, \boldsymbol{u}, t) + \left[\frac{\partial J^*}{\partial \boldsymbol{x}} \right]^T \cdot \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \Bigg\} = 0. \tag{4.21}$$

Equation (4.21) is known as the Hamilton-Jacobi-Bellman (HJB) equation, and it is the central result that we have been looking for – a differential equation that the value function for our problem of interest must satisfy. In optimal control literature, the Hamiltonian H is defined to be (Evans, 2010)

$$H\left(\boldsymbol{x}, \frac{\partial J^*}{\partial \boldsymbol{x}}, t \right) = \min_{\boldsymbol{u}} \Bigg\{ g(\boldsymbol{x}, \boldsymbol{u}, t) + \left[\frac{\partial J^*}{\partial \boldsymbol{x}} \right]^T \cdot \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \Bigg\}, \tag{4.22}$$

in which case the Hamilton-Jacobi-Bellman equation is equivalently expressed as

$$\frac{\partial J^*}{\partial t} + H\left(\boldsymbol{x}, \frac{\partial J^*}{\partial \boldsymbol{x}}, t \right) = 0. \tag{4.23}$$

The boundary condition for the Hamilton-Jacobi-Bellman PDE can be found by considering the value function at the final time $t = t_f$:

$$J^*(\boldsymbol{x}, t = t_f) = h(\boldsymbol{x}, t_f). \tag{4.24}$$

That is, the value function at any state $\boldsymbol{x}$ at the final time is intuitively given by the terminal cost. This is due to the fact that at the final time, the minimum cost possible at a given state is simply the terminal cost since the running cost has no contribution. Equations (4.21) and (4.24) together form a final value problem which can be solved for the value function J^*.

Equation (4.21) furthermore encodes the optimal control policy for the system. Specifically, the optimal closed loop control at any state and time is given by the minimizing control in the Hamiltonian:

$$u^*(\boldsymbol{x}, t) = \arg\min_u \left\{ g(\boldsymbol{x}, \boldsymbol{u}, t) + \left[\frac{\partial J^*}{\partial \boldsymbol{x}}\right]^T \cdot \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \right\}. \tag{4.25}$$

From Equation (4.25) it can be seen that the optimal policy will depend in general on $\boldsymbol{x}$, t and $\frac{\partial J^*}{\partial \boldsymbol{x}}$.

Summary It is beneficial to take a step back and summarize the key results that have been obtained up until now. Recall that our primary goal has been to solve the optimization problem of determining the optimal control function $\underset{t_0 \leq \tau \leq t_f}{\boldsymbol{u}^*(\tau)}$ which causes the system governed by

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t), \quad \boldsymbol{x}(t_0) = \boldsymbol{x}_0, \quad t \in [t_0, t_f],$$

to follow a trajectory $\boldsymbol{x}^*(\cdot)$ that minimizes the cost function

$$J(\boldsymbol{x}_0, t_0, \underset{t_0 \leq \tau \leq t_f}{\boldsymbol{u}(\tau)}) = h(\boldsymbol{x}(t_f), t_f) + \int_{t_0}^{t_f} g(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), \tau) d\tau.$$

The method of dynamic programming provides an approach to do this by solving for the value function $J^*(\boldsymbol{x}, t)$. In particular, after setting the final condition $J^*(\boldsymbol{x}, t =$

$t_f) = h(\boldsymbol{x}, t_f)$, we solve the Hamilton-Jacobi-Bellman PDE (Equation (4.21)) for the value function $J^*(\boldsymbol{x}, t)$ at any arbitrary state and at any time $t \in [t_0, t_f]$. Moreover, at the initial state and time of interest, $(\boldsymbol{x}_0, t_0)$, the computed value function can be evaluated and physically corresponds to:

$$\begin{aligned} J^*(\boldsymbol{x}_0, t_0) &= \min_{\substack{\boldsymbol{u}(\tau) \\ t_0 \le \tau \le t_f}} \left\{ J(\boldsymbol{x}_0, t_0, \underset{t_0 \le \tau \le t_f}{\boldsymbol{u}(\tau)}) \right\}, \\ &= \min_{\substack{\boldsymbol{u}(\tau) \\ t_0 \le \tau \le t_f}} \left\{ h(\boldsymbol{x}(t_f), t_f) + \int_{t_0}^{t_f} g(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), \tau) d\tau \right\}. \end{aligned} \tag{4.26}$$

That is, the value function gives the minimum possible cost for the cost function for our optimization problem of interest. Starting from the state $\boldsymbol{x}_0$ at time t_0, the value function J^* can finally be used to find the optimal trajectory $\boldsymbol{x}^*(\cdot)$ through the state space using the optimal closed loop control law (Equation (4.25)).

4.2 Forward Reachability

Hamilton-Jacobi reachability analysis has proven to be quite useful in numerous applications. Primarily it has been employed as a formal method for guaranteeing the performance and safety properties of dynamical systems (Bansal et al., 2017). In recent years, however, reachability analysis has also proved to be a versatile method for optimal path planning in dynamic environments. Lolla et al. (2012, 2014b,a) first introduced approaches for time optimal path planning in continuous, strong and dynamic ocean environments built on Hamilton-Jacobi reachability. Our method for harvest-time optimal path planning, which builds upon (Lolla et al., 2014b), is also based on reachability. As a result, in this section, we summarize the fundamentals of Hamilton-Jacobi reachability as this will prove to be crucial going forward. The material in this section follows primarily (Kurzhanski and Varaiya, 2001) and we refer the reader to this reference for further details.

We now consider the same general dynamical system given by Equation (4.1), restated here for convenience:

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t), \quad \boldsymbol{x}(t_0) = \boldsymbol{x}_0, \quad t \in [t_0, t_f].$$

Furthermore, we follow the same convention as before of denoting a trajectory of the system from an initial state $\boldsymbol{x}_0$ at time t_0 using the control sequence $\boldsymbol{u}(\cdot)$ as $\boldsymbol{x}(t; t_0, \boldsymbol{x}_0, \boldsymbol{u}(\cdot))$.

Reachability analysis is concerned with the determination of the set of states that can be reached by a system with its available controls. Various sets of interest are typically computed based on the application at hand, most notably the backward reachable set, backward reachable tube, forward reachable set and forward reachable tube (Chen and Tomlin, 2018). Of particular relevance in this work is the forward reachable set. Consider a set of possible initial states of the system, $\mathcal{X}_0 \subset \mathbb{R}^n$, at the time t_0. The forward reachable set $\mathcal{X}(\hat{t}, t_0, \mathcal{X}_0)$ at a time $\hat{t} \in [t_0, t_f]$ can mathematically be expressed as:

$$\mathcal{X}(\hat{t}, t_0, \mathcal{X}_0) = \cup \left\{ \boldsymbol{x}(\hat{t}; t_0, \boldsymbol{x}_0, \boldsymbol{u}(\cdot)) \mid \boldsymbol{u}(\cdot) \in \mathbb{U}, \boldsymbol{x}_0 \in \mathcal{X}_0 \right\}. \tag{4.27}$$

In simple terms, the reachable set in Equation (4.27) contains all the states that the system could be in at time $\hat{t}$ if the initial state were drawn from the set $\mathcal{X}_0$. The boundary of this set, $\partial\mathcal{X}(\hat{t}, t_0, \mathcal{X}_0)$, is denoted as the reachability front. A schematic depicting the forward reachable set is shown in Figure 4-3.

The forward reachable set can be conveniently computed by solving an optimal control problem (Kurzhanski and Varaiya, 2001). To formulate the problem, we first start by defining the signed distance function. Given an initial set $\mathcal{X}_0$ and its boundary $\partial\mathcal{X}_0$, the signed distance function $d(\boldsymbol{x}, \partial\mathcal{X}_0)$ of the state $\boldsymbol{x}$ to the boundary is defined as

$$d(\boldsymbol{x}, \partial\mathcal{X}_0) = \begin{cases} -\min_{\boldsymbol{y} \in \partial\mathcal{X}_0} |\boldsymbol{x} - \boldsymbol{y}|, & \text{if } \boldsymbol{x} \in \mathcal{X}_0 \\ \min_{\boldsymbol{y} \in \partial\mathcal{X}_0} |\boldsymbol{x} - \boldsymbol{y}|, & \text{if } \boldsymbol{x} \notin \mathcal{X}_0 \end{cases}. \tag{4.28}$$

From Equation (4.28), it can be seen that any state $\boldsymbol{x}$ in the set $\mathcal{X}_0$ will have a

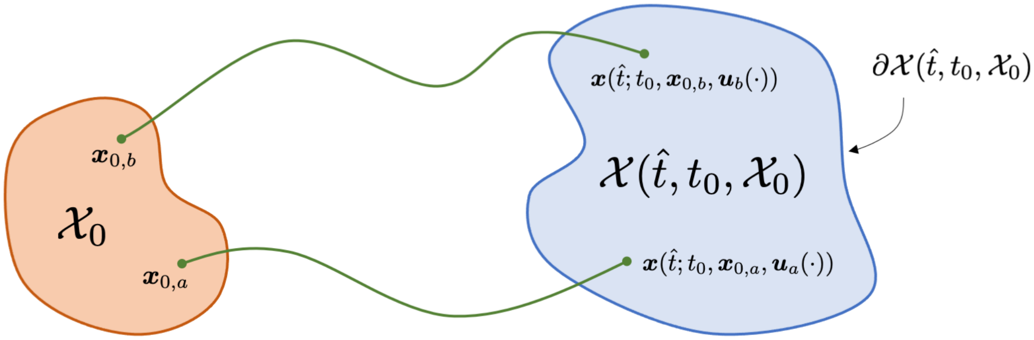


Figure 4-3: A schematic showing the forward reachable set. Starting from somewhere in the initial set $\mathcal{X}_0$ at the initial time t_0, the forward reachable set $\mathcal{X}(\hat{t}, t_0, \mathcal{X}_0)$ contains all the states that the system could be in (using an admissible control function) at time $\hat{t}$.

negative distance.

In what follows, we will compute the forward reachability set at a time $\hat{t}$ as follows. We will first consider the dynamics of the system *backwards* in time. This will involve a process of mapping the dynamics equations to a pseudo-time that allows to study the backward evolution of the system. We will then see that this will allow us to formulate an optimal control problem whose value function can then be used to extract the forward reachable set.

We begin by analyzing the evolution of the system backwards in time. Consider a prescribed control function $\boldsymbol{u}_s(t)$ defined on $t \in [t_0, t_f]$ resulting in the system defined by Equation (4.1) to follow a trajectory $\boldsymbol{x}(t) = \boldsymbol{x}(t; t_0, \boldsymbol{x}_0, \boldsymbol{u}_s(\cdot))$ from $\boldsymbol{x}_0 = \boldsymbol{x}(t_0; t_0, \boldsymbol{x}_0, \boldsymbol{u}_s(\cdot))$ at time $t = t_0$ to $\boldsymbol{x}_f = \boldsymbol{x}(t_f; t_0, \boldsymbol{x}_0, \boldsymbol{u}_s(\cdot))$ at time $t = t_f$. This trajectory can be identically traced out by considering the dynamics backwards in time. First, a pseudo-time variable τ can be defined in order to march backwards in time (but forward in pseudo-time) using the following invertible mapping:

$$\tau(t) = t_f - t, \quad t(\tau) = t_f - \tau. \tag{4.29}$$

The mapped pseudo-time domain correspondingly translates as follows:

$$t \in [t_0, t_f] \rightarrow \tau \in [\tau_0, \tau_f] = [0, t_f - t_0]. \tag{4.30}$$

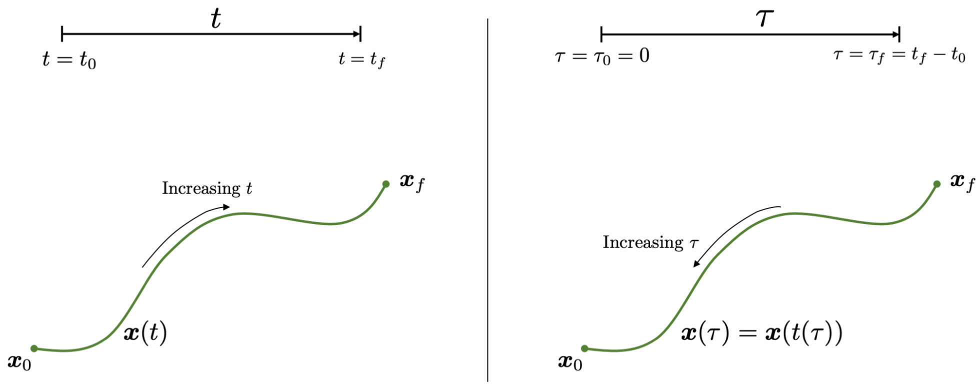


Figure 4-4: A schematic depicting the mapping of the dynamics into pseudo-time in order to equivalently analyze the system evolving backwards in time.

A state trajectory $\boldsymbol{x}(t)$ in the forward time domain is furthermore related to its counterpart $\boldsymbol{x}(\tau) = \boldsymbol{x}(t(\tau))$ in the backward time domain. Specifically, we have that the time derivative of the trajectory satisfies the following relation:

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}\tau} = \frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} \cdot \frac{\mathrm{d}t}{\mathrm{d}\tau} = -\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t}. \tag{4.31}$$

It follows then that the ODE

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}\tau} = -\boldsymbol{f}(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), t(\tau)), \quad \boldsymbol{x}(\tau = 0) = \tilde{\boldsymbol{x}}_0, \quad \tau \in [\tau_0, \tau_f] = [0, t_f - t_0], \tag{4.32}$$

governs the dynamics of the system backwards in time. Equation (4.32), solved using the control function $\boldsymbol{u}(\tau) = \boldsymbol{u}_s(t(\tau))$ and "initial condition" $\tilde{\boldsymbol{x}}_0 = \boldsymbol{x}_f$, will trace out the exact same trajectory as before, i.e. $\boldsymbol{x}(t; t_0, \boldsymbol{x}_0, \boldsymbol{u}_s(\cdot))$, but by marching backward in time (forward in pseudo-time) starting at the destination state $\boldsymbol{x}_f$ (Figure 4-4).

Now with the backwards dynamics defined, we return to the problem of computing the forward reachable set $\mathcal{X}(\hat{t}, t_0, \mathcal{X}_0)$ at a time $\hat{t} \in [t_0, t_f]$. Consider the backwards dynamical system, using the pseudo-time mapping $\tau(t) = \hat{t} - t$, given by

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}\tau} = -\boldsymbol{f}(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), t(\tau)), \quad \boldsymbol{x}(\tau = 0) = \tilde{\boldsymbol{x}}_0, \quad \tau \in [\tau_0, \tau_f] = [0, \hat{t} - t_0]. \tag{4.33}$$

Furthermore, consider the cost function, i.e. performance measure, composed of no running cost, i.e. $g(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), \tau) = 0$, and a terminal cost $h(\boldsymbol{x}(\tau_f), \tau_f) = d(\boldsymbol{x}(\tau_f), \partial\mathcal{X}_0)$. The corresponding value function for any arbitrary state $\boldsymbol{x}$ and pseudo-time τ is then given as:

$$J^*(\boldsymbol{x}, \tau) = \min_{\substack{\boldsymbol{u}(s) \\ \tau \leq s \leq \tau_f}} \Big\{ h(\boldsymbol{x}(\tau_f), \tau_f) \Big\} = \min_{\substack{\boldsymbol{u}(s) \\ \tau \leq s \leq \tau_f}} \Big\{ d(\boldsymbol{x}(\tau_f), \partial\mathcal{X}_0) \Big\}. \tag{4.34}$$

The value function further satisfies the Hamilton-Jacobi-Bellman equation:

$$\begin{aligned} \frac{\partial J^*}{\partial \tau} + \min_{\boldsymbol{u}} \left\{ \left[\frac{\partial J^*}{\partial \boldsymbol{x}} \right]^T \cdot \Big[-\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \tau) \Big] \right\} = 0 \\ J^*(\boldsymbol{x}, \tau = \tau_f) = d(\boldsymbol{x}, \partial\mathcal{X}_0). \end{aligned} \tag{4.35}$$

The solution of Equation (4.35) gives a value function in the $(\boldsymbol{x}, \tau)$ domain. Using the pseudo-time mapping, this can be mapped back to the $(\boldsymbol{x}, t)$ domain to give $J^*(\boldsymbol{x}, t) = J^*(\boldsymbol{x}, \tau(t))$ – the value function expressed in the standard/forward time domain.

Let us take now a step back to get an intuitive understanding of what this value function corresponds to. We solved for the value function $J^*(\boldsymbol{x}, \tau)$ in pseudo-time which can be equivalently expressed in the standard time domain as $J^*(\boldsymbol{x}, t)$ using the time mapping. Consider the value function at the initial pseudo-time $\tau = \tau_0$ and an arbitrary state $\boldsymbol{x}$ ($J^*(\boldsymbol{x}, \tau = \tau_0)$). This pseudo-time maps to a forward time of $t = \hat{t}$ and so is equivalently given by $J^*(\boldsymbol{x}, t = \hat{t})$ if we consider the value function mapped to the forward time domain. The value of $J^*(\boldsymbol{x}, \tau = \tau_0)$ – or equivalently $J^*(\boldsymbol{x}, t = \hat{t})$ – at a state $\boldsymbol{x}$ gives the *minimum signed distance* (relative to the boundary $\partial\mathcal{X}_0$) for an initial condition $\boldsymbol{x}_0$ at time $t = t_0$ such that a control function brings the system to $\boldsymbol{x}$ at time $t = \hat{t}$. Mathematically, this can be expressed as:

$$J^*(\boldsymbol{x}, \tau = \tau_0) = J^*(\boldsymbol{x}, t = \hat{t}) = \min_{\substack{\boldsymbol{u}(s) \\ t_0 \leq s \leq \hat{t}}} \Big\{ d(\boldsymbol{x}_0, \partial\mathcal{X}_0) \mid \boldsymbol{x}(\hat{t}; t_0, \boldsymbol{x}_0, \boldsymbol{u}(\cdot)) = \boldsymbol{x} \Big\}. \tag{4.36}$$

Therefore, the value function effectively encodes information about where a certain state came from at the initial time ($t = t_0$ or $\tau = \tau_f$ based on which direction of time is being considered). From this it is straightforward to see that the forward reachable set can be computed from the value function as follows

$$\mathcal{X}(\hat{t}, t_0, \mathcal{X}_0) = \{\boldsymbol{x} \mid J^*(\boldsymbol{x}, \tau(\hat{t})) \leq 0\}. \tag{4.37}$$

That is, all states reachable at time $t = \hat{t}$ by initial states in the set $\mathcal{X}_0$ will have a negative value for the value function. Furthermore, the forward reachability front is given as

$$\partial\mathcal{X}(\hat{t}, t_0, \mathcal{X}_0) = \{\boldsymbol{x} \mid J^*(\boldsymbol{x}, \tau(\hat{t})) = 0\}. \tag{4.38}$$

Although our work is technically complete, it is convenient to map Equation (4.35) back to the $(\boldsymbol{x}, t)$ domain so as not to have to explicitly work in pseudo-time τ. To do this, we first note that the minimization of the Hamiltonian in Equation (4.35) can be equivalently expressed as a maximization as follows:

$$\min_{\boldsymbol{u}} \left\{ \left[\frac{\partial J^*}{\partial \boldsymbol{x}}\right]^T \cdot \left[-\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \tau)\right] \right\} = -\max_{\boldsymbol{u}} \left\{ \left[\frac{\partial J^*}{\partial \boldsymbol{x}}\right]^T \cdot \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \tau) \right\}. \tag{4.39}$$

Mapping now from the $(\boldsymbol{x}, \tau)$ domain to the $(\boldsymbol{x}, t)$ domain we have the following relation for the value function's time derivative:

$$\frac{\partial J^*}{\partial t} = \frac{\partial J^*}{\partial \tau} \cdot \frac{\partial \tau}{\partial t} = -\frac{\partial J^*}{\partial \tau}. \tag{4.40}$$

The spatial derivatives, of course, are unchanged. Furthermore, the final-condition in Equation (4.35) maps as follows:

$$J^*(\boldsymbol{x}, \tau = \tau_f) = J^*(\boldsymbol{x}, t = t(\tau_f)) = J^*(\boldsymbol{x}, t = t_0) = d(\boldsymbol{x}, \partial\mathcal{X}_0). \tag{4.41}$$

Putting this all together, we have that Equation (4.35) is equivalently given in the

$(\boldsymbol{x}, t)$ domain as:

$$\begin{aligned} \frac{\partial J^*}{\partial t} + \max_{\boldsymbol{u}} \left\{ \left[\frac{\partial J^*}{\partial \boldsymbol{x}} \right]^T \cdot \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \right\} = 0 \\ J^*(\boldsymbol{x}, t = t_0) = d(\boldsymbol{x}, \partial \mathcal{X}_0) \end{aligned} \tag{4.42}$$

Equation (4.42) is the key result to be taken from this section. It corresponds to an initial value problem that can be solved starting from t_0 to any time t of interest. To recap, the resulting value function $J^*(\boldsymbol{x}, t)$ can then be used to get the forward reachable set for a system, starting from some initial set $\mathcal{X}_0$ at time t_0, at the time t as follows

$$\mathcal{X}(t, t_0, \mathcal{X}_0) = \{\boldsymbol{x} \mid J^*(\boldsymbol{x}, t) \leq 0\}, \tag{4.43}$$

with the forward reachable front given as

$$\partial \mathcal{X}(t, t_0, \mathcal{X}_0) = \{\boldsymbol{x} \mid J^*(\boldsymbol{x}, t) = 0\}. \tag{4.44}$$

Figure 4-5 provides a schematic to further enforce the physical interpretation of the resulting value function.

The optimal controls, moreover, are given by the maximizing controls for the Hamiltonian. Specifically, we have a control law of the following form:

$$\pi^*(\boldsymbol{x}, t) = \arg\max_{\boldsymbol{u}} \left\{ \left[\frac{\partial J^*}{\partial \boldsymbol{x}} \right]^T \cdot \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \right\}. \tag{4.45}$$

It is important to note that this forms a closed loop control for our *backwards* dynamical system, an unsurprising fact as our optimal control problem was formulated for the system evolving backwards in time. That is, this control law can be used to determine how to bring any arbitrary state $\boldsymbol{x}$ at some time $t > t_0$ back as close as possible (using the signed distance metric) to the initial boundary $\partial \mathcal{X}_0$ at time t_0.

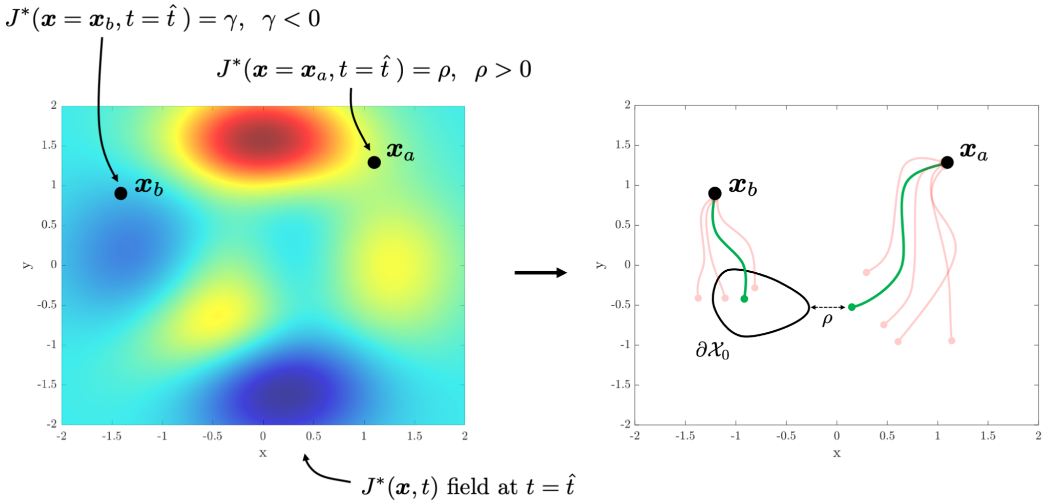


Figure 4-5: A schematic showing the physical interpretation of the value function solved from Equation (4.42). Shown on the left is a hypothetical value function field $J^*(\boldsymbol{x}, t=\hat{t})$ at the time $t=\hat{t}$. At any given state $\boldsymbol{x}$, the value function gives the minimum possible signed distance of an initial state $\boldsymbol{x}_0$ (at time $t=t_0$) relative to the boundary of the set $\partial\mathcal{X}_0$ such that a trajectory exists that takes the system to $\boldsymbol{x}$ at time $t=\hat{t}$. This is illustrated on the right, where the optimal trajectory that minimizes the cost is shown in green while other suboptimal candidate trajectories are shown in red.

4.3 Viscosity Solutions and Hamilton-Jacobi Equations

We now turn our attention to discussing some theoretical properties of the form of the PDEs that will be key to our path planning methodology. Specifically, we look to address the existence, uniqueness and various properties of solutions to initial value problems of the following form:

$$\begin{aligned} \frac{\partial w}{\partial t} + H\left(\frac{\partial w}{\partial \boldsymbol{x}}, \boldsymbol{x}\right) &= 0 \quad \text{in } \mathbb{R}^n \times (0,\infty), \\ w &= g \quad \text{on } \mathbb{R}^n \times \{t=0\}. \end{aligned} \tag{4.46}$$

Equation (4.46) is known as a Hamilton-Jacobi equation. H is the Hamiltonian of the form $H : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R}^n \to \mathbb{R}$ is the initial function. The unknown function to be determined is $w : \mathbb{R}^n \times [0,\infty) \to \mathbb{R}$, where $\frac{\partial w}{\partial \boldsymbol{x}} \in \mathbb{R}^n$ corresponds to the spatial gradient and $\frac{\partial w}{\partial t}$ its temporal derivative. We also note that the results that follow translate over Hamiltonians with a time dependence (details for this can be found in (Bardi and Capuzzo-Dolcetta, 2008)). The material in this section closely follows (Evans, 2010).

In proving properties on the existence and uniqueness of solutions to Equation (4.46), it turns out to be quite difficult to demand from the start a certain level of smoothness for the solution itself. Rather, what is much easier to do is for one to first *define* a class of *weak solutions* (or *generalized solutions*). This class of weak solutions is often defined such that not too much is asked for in terms of smoothness. Ultimately, it turns out to be much easier to prove existence and uniqueness in this chosen class of solutions (Evans, 2010).

Our approach to studying solutions to the Hamilton-Jacobi equation will be intimately connected to the following alternate problem

$$\begin{aligned} \frac{\partial w^\epsilon}{\partial t} + H\left(\frac{\partial w^\epsilon}{\partial \boldsymbol{x}}, \boldsymbol{x}\right) &= \epsilon \cdot \Delta w^\epsilon \quad \text{in } \mathbb{R}^n \times (0, \infty), \\ w^\epsilon &= g \quad \text{on } \mathbb{R}^n \times \{t = 0\}, \end{aligned} \tag{4.47}$$

where $\epsilon > 0$. Equation (4.47) consists of the original Hamilton-Jacobi equation with the addition of a viscosity term. The inclusion of this viscosity in Equation (4.47) renders it to be a quasilinear parabolic PDE which in turn has a unique smooth solution w^ϵ (Evans, 2010; Sethian, 1999). Of interest to us is the resulting solution to Equation (4.46) found in the limit of $\epsilon \to 0$ for solutions w^ϵ to Equation (4.47) (refer to Figure 4-6). This technique, known as the method of vanishing viscosity, we will see will be connected the weak solutions we consider to Equation (4.46) (Evans, 2010).

We now look to define the class of weak solutions – what we will refer to as the viscosity solution – to consider for Equation (4.46). This class of solution is characterized based on its behaviour at local extrema. We have the following definition from (Evans, 2010):

Definition 4.3.1. *A bounded, uniformly continuous function w is called a viscosity*

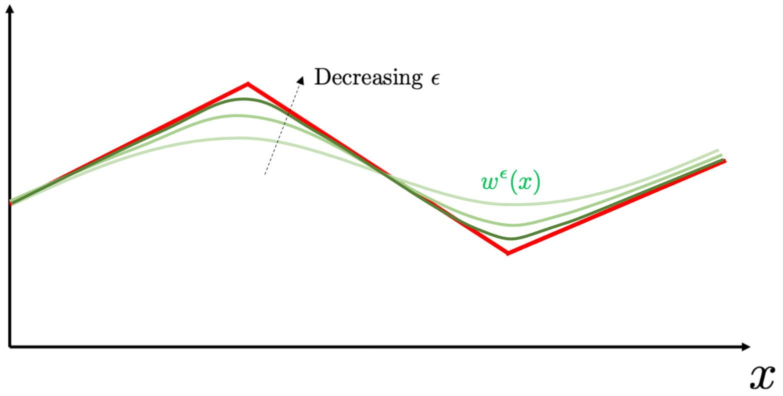


Figure 4-6: A schematic depicting hypothetical solutions to a Hamilton-Jacobi equation with the addition of a viscosity term (Equation (4.47)). Shown are different solutions w^ϵ for different values of ϵ (in green). Depicted is the solution found by the method of vanishing viscosity (in red) by considering the limit of $\epsilon \to 0$.

solution of the initial-value problem from Equation (4.46) *for the Hamilton-Jacobi equation provided:*

1. $w = g$ *on* $\mathbb{R}^n \times \{t = 0\}$

2. *For each smooth test function* $v \in \mathcal{C}^\infty(\mathbb{R}^n \times (0, \infty))$*:*

   (a) *If* $w - v$ *has a local maximum at a point* $(\boldsymbol{x}_0, t_0) \in \mathbb{R}^n \times (0, \infty)$*, then*

$$\left.\frac{\partial v}{\partial t}\right|_{(\boldsymbol{x}_0,t_0)} + H\left(\left.\frac{\partial v}{\partial \boldsymbol{x}}\right|_{(\boldsymbol{x}_0,t_0)}, \boldsymbol{x}_0\right) \leq 0. \tag{4.48}$$

   (b) *If* $w - v$ *has a local minimum at a point* $(\boldsymbol{x}_0, t_0) \in \mathbb{R}^n \times (0, \infty)$*, then*

$$\left.\frac{\partial v}{\partial t}\right|_{(\boldsymbol{x}_0,t_0)} + H\left(\left.\frac{\partial v}{\partial \boldsymbol{x}}\right|_{(\boldsymbol{x}_0,t_0)}, \boldsymbol{x}_0\right) \geq 0. \tag{4.49}$$

It is important to note that, in Definition 4.3.1, the solution w is never differentiated. Rather, all derivatives have been moved to the smooth test function v. It is also important to emphasize that for a candidate solution w to be a viscosity solution, it must satisfy the properties given by Equations (4.48) and (4.49) for *all* smooth functions v (Evans, 2010).

With the defined class of viscosity solutions (the weak solution), we have the following property on the consistency of a viscosity solution from (Evans, 2010):

Theorem 4.3.1. *Let w be a viscosity solution to the Hamilton-Jacobi equation given by Equation* (4.46)*, and suppose w is differentiable at some point $(\boldsymbol{x}_0, t_0)$ in $\mathbb{R}^n \times (0, \infty)$. Then*

$$\left.\frac{\partial w}{\partial t}\right|_{(\boldsymbol{x}_0,t_0)} + H\left(\left.\frac{\partial w}{\partial \boldsymbol{x}}\right|_{(\boldsymbol{x}_0,t_0)}, \boldsymbol{x}_0\right) = 0.$$

In other words, Theorem 4.3.1 states that if we are at a point where the viscosity solution is differentiable, then the solution satisfies the Hamilton-Jacobi equation at that given point. Additionally, we have the following key result on the uniqueness of a viscosity solution from (Evans, 2010):

Theorem 4.3.2. *Assume that the Hamiltonian H satisfies the following conditions of Lipschitz continuity:*

$$\begin{aligned} |H(\boldsymbol{p}, \boldsymbol{x}) - H(\boldsymbol{q}, \boldsymbol{x})| &\leq C\,|\boldsymbol{p} - \boldsymbol{q}| \\ |H(\boldsymbol{p}, \boldsymbol{x}) - H(\boldsymbol{p}, \boldsymbol{y})| &\leq C\,|\boldsymbol{x} - \boldsymbol{y}| \cdot (1 + |\boldsymbol{p}|), \end{aligned} \tag{4.50}$$

for $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{p}, \boldsymbol{q} \in \mathbb{R}^n$ and a constant $C \geq 0$. Then, there exists at most one viscosity solution to the Hamilton-Jacobi equation given by Equation (4.46)*.*

On top of this, several other properties can be shown to hold. The following additional claims are taken from (Sethian, 1999):

- If w is a smooth solution of the Hamilton-Jacobi equation, then it is a viscosity solution.
- The solution produced by taking the limit of the smooth solutions w^ϵ as ϵ goes to zero is the viscosity solution.

The last claim helps explain the reason for considering the alternate problem given by Equation (4.47). Even though no mention of the viscous counterpart to the Hamilton-Jacobi equation was made in Definition 4.3.1, it turns out that the unique viscosity solution is indeed equivalent to that found by the method of vanishing viscosity.

For many more details, as well as proofs verifying these properties, we refer the reader to (Evans, 2010; Sethian, 1999) and the references cited therein.

4.3.1 Details on Hamilton-Jacobi-Bellman Equations

Section 4.3 has so far addressed properties of general Hamilton-Jacobi equations. In this section, we consider some properties related to Hamiltonians that arise from solving optimal control problems by the method of dynamic programming. That is, we consider in this subsection some of the specific results on the properties of viscosity solutions to Hamilton-Jacobi-Bellman equations.

We consider the optimal control problem resulting in a value function w satisfying the Hamilton-Jacobi-Bellman equation given by

$$\begin{aligned} \frac{\partial w}{\partial t} + \min_{\boldsymbol{u}} \left\{ g(\boldsymbol{x}, \boldsymbol{u}) + \left[\frac{\partial w}{\partial \boldsymbol{x}} \right]^T \cdot \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) \right\} = 0, &\quad \text{in } \mathbb{R}^n \times [0, T), \\ w = g, &\quad \text{on } \mathbb{R}^n \times \{t = T\}. \end{aligned} \tag{4.51}$$

To be consistent with the material presented in (Evans, 2010) note that the PDE in Equation (4.51) is defined on $t \in [0, T]$ rather than $t \in [t_0, t_f]$ as we have seen earlier. Of course, it is a straightforward procedure to map a given equation from $t \in [t_0, t_f]$ to the $[0, T]$ domain to be consistent with the theorems presented in what follows. Furthermore, note that Equation (4.51) is for an autonomous system (i.e. dynamics and running costs that do not explicitly depend on time), and all results that follow are for this class of systems following the presentation in (Evans, 2010). These results however can be naturally extended to non-autonomous systems (i.e. dynamics and running costs that do explicitly depend on time) as well, and we refer the reader to (Bardi and Capuzzo-Dolcetta, 2008) for further details.

In addition to the smoothness assumptions on the dynamics, we further assume that the running cost and terminal cost satisfy

$$\begin{aligned}|g(\boldsymbol{x},\boldsymbol{u})| &\leq C,\\ |g(\boldsymbol{x},\boldsymbol{u})-g(\boldsymbol{y},\boldsymbol{u})| &\leq C\cdot|\boldsymbol{x}-\boldsymbol{y}|,\end{aligned}\tag{4.52}$$

and

$$\begin{aligned}|h(\boldsymbol{x})| &\leq C,\\ |h(\boldsymbol{x})-h(\boldsymbol{y})| &\leq C\cdot|\boldsymbol{x}-\boldsymbol{y}|,\end{aligned}\tag{4.53}$$

for states $\boldsymbol{x},\boldsymbol{y}\in\mathbb{R}^n$, control $\boldsymbol{u}\in\mathcal{U}$ and some constant $C>0$ (Evans, 2010). From these inequalities, it follows that the Hamiltonian

$$H\left(\frac{\partial w}{\partial \boldsymbol{x}},\boldsymbol{x}\right)=\min_{\boldsymbol{u}}\left\{g(\boldsymbol{x},\boldsymbol{u})+\left[\frac{\partial w}{\partial \boldsymbol{x}}\right]^T\cdot\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})\right\},$$

satisfies the smoothness assumptions given by Equation (4.50) in Theorem 4.3.2 (Evans, 2010).

It is important to note that the Hamilton-Jacobi-Bellman equation is a terminal-value problem whereas our preceeding analysis on Hamilton-Jacobi equations in the first part of Section 4.3 has studied initial value problems. The terminal value problem requires a new definition for the weak solution (viscosity solution). From (Evans, 2010) we have the following:

Definition 4.3.2. *A bounded, uniformly continuous function w is called a viscosity solution of the final-value problem for the Hamilton-Jacobi-Bellman equation given by Equation* (4.51) *provided:*

1. $w=g$ *on* $\mathbb{R}^n\times\{t=T\}$

2. *For each smooth test function* $v\in\mathcal{C}^\infty(\mathbb{R}^n\times(0,T))$*:*

(a) If $w - v$ has a local maximum at a point $(\boldsymbol{x}_0, t_0) \in \mathbb{R}^n \times (0, T)$, then

$$\left.\frac{\partial v}{\partial t}\right|_{(\boldsymbol{x}_0,t_0)} + H\left(\left.\frac{\partial v}{\partial \boldsymbol{x}}\right|_{(\boldsymbol{x}_0,t_0)}, \boldsymbol{x}_0\right) \geq 0. \tag{4.54}$$

(b) If $w - v$ has a local minimum at a point $(\boldsymbol{x}_0, t_0) \in \mathbb{R}^n \times (0, T)$, then

$$\left.\frac{\partial v}{\partial t}\right|_{(\boldsymbol{x}_0,t_0)} + H\left(\left.\frac{\partial v}{\partial \boldsymbol{x}}\right|_{(\boldsymbol{x}_0,t_0)}, \boldsymbol{x}_0\right) \leq 0. \tag{4.55}$$

Note that the key difference arising relative to the earlier definition for the initial-value problem counterpart is that the inequalities reverse their direction in Equations (4.54) and (4.55).

The smoothness requirements on the Hamiltonian, resulting from the assumptions on the dynamics and cost function components, and the updated definition of the viscosity solution result in the following key theorem on the uniqueness of solutions to the Hamilton-Jacobi-Bellman equation (Evans, 2010):

Theorem 4.3.3. *The value function w is the unique viscosity solution for the terminal-value problem for the Hamilton-Jacobi-Bellman equation given by Equation* (4.51)*.*

Theorem 4.3.3 therefore ensures for us that, provided that the dynamics and cost function components (i.e. running cost and terminal cost) are sufficiently smooth, the value function for the optimal control problem will be the unique solution to the corresponding Hamilton-Jacobi-Bellman PDE. Additionally, following from the smoothness assumptions on the augmented dynamics – which implies global Lipschitz continuity in the state variable, uniformly in the control variable – we have that $w(\boldsymbol{x}, t)$ is continuous as well (Bardi and Capuzzo-Dolcetta, 2008). This continuity property of the value function will be key for us going forward.

4.4 Harvest-Time Optimal Path Planning

In this section, we now present the main theoretical development for our new harvest-time optimal path planning methodology. It should be noted that, to remain consistent with earlier work presented in (Lolla et al., 2014b,a; Kulkarni and Lermusiaux, 2020; Subramani et al., 2018; Subramani and Lermusiaux, 2016, 2019), going forward we will denote the value function as $\phi(\cdot)$ rather than $J^*(\cdot)$.

4.4.1 Review: Time Optimal Path Planning with Forward Reachability

Before turning our attention to harvest-time optimal path planning, we first briefly review how forward reachability can be used in the context of time optimal path planning. For further details, we refer the reader to (Lolla et al., 2014b,a; Lermusiaux et al., 2017b).

For time optimal path planning, we consider the problem of navigating from a start point to a specified target through the dynamic flow field in minimal time. Consider a vehicle in a flow field $\boldsymbol{V}(\boldsymbol{x}, t)$. Starting from a point $\boldsymbol{x}_s$, the vehicle's motion through the physical space is governed by the ODE given in Equation (3.2a) which is restated here for convenience:

$$\frac{\mathrm{d}\boldsymbol{X}_p}{\mathrm{d}t} = \boldsymbol{V}\left(\boldsymbol{X}_p(t), t\right) + F(t)\,\hat{\boldsymbol{h}}(t), \quad \boldsymbol{X}_p(t=0) = \boldsymbol{x}_s.$$

The position of the vehicle at any time t, $\boldsymbol{X}_p(t)$, is ultimately governed by the control functions chosen for the speed $F(t)$ given by $F(t) : [0, \infty) \to [0, F_{max}],\ F_{max} \in \mathbb{R}_{>0}$ and heading:

$$\hat{\boldsymbol{h}}(t) : [0, \infty) \to \mathbb{R}^d \ \text{ such that } \left\|\hat{\boldsymbol{h}}(t)\right\| = 1,\ \forall t.$$

Considering an initial set $\mathcal{X}_0 = \{\boldsymbol{x}_s\}$ for the system (i.e. the vehicle) at time $t_0 = 0$, we aim to compute the forward reachable set using Equation (4.42). Substituting the dynamics, the evolution of the reachable set is given by

$$\frac{\partial \phi(\boldsymbol{x}, t)}{\partial t} + \max_{F, \hat{\boldsymbol{h}}} \left\{ F\,\hat{\boldsymbol{h}}^T \cdot \frac{\partial \phi}{\partial \boldsymbol{x}} \right\} + \boldsymbol{V}(\boldsymbol{x}, t) \cdot \frac{\partial \phi}{\partial \boldsymbol{x}} = 0,$$
$$\phi(\boldsymbol{x}, t=0) = \phi_0, \tag{4.57}$$

where ϕ_0 is the initial condition given by the signed distance function relative to the start point. The maximization in Equation (4.57) can be performed analytically. In particular, we have that the optimal speed and heading is given, at any point in space and time, as

$$F^*(\boldsymbol{x}, t) = F_{max}, \tag{4.58}$$

and

$$\hat{\boldsymbol{h}}^*(\boldsymbol{x}, t) = \frac{\left.\frac{\partial \phi}{\partial \boldsymbol{x}}\right|_{(\boldsymbol{x},t)}}{\left\| \left.\frac{\partial \phi}{\partial \boldsymbol{x}}\right|_{(\boldsymbol{x},t)} \right\|}, \tag{4.59}$$

resulting in the following PDE governing the evolution of the forward reachable set (Lolla et al., 2014b):

$$\begin{aligned} \frac{\partial \phi(\boldsymbol{x}, t)}{\partial t} + F_{max} \left\| \frac{\partial \phi}{\partial \boldsymbol{x}} \right\| + \boldsymbol{V}(\boldsymbol{x}, t) \cdot \frac{\partial \phi}{\partial \boldsymbol{x}} &= 0, \\ \phi(\boldsymbol{x}, t = 0) &= \phi_0. \end{aligned} \tag{4.60}$$

With the dynamics of the forward reachable set determined, time optimal paths can be computed by following the reachability front, i.e. $\{\boldsymbol{x} : \phi(\boldsymbol{x}, t) = 0\}$. Specifically, the PDE in Equation (4.60) is solved until the very first time t_f such that $\phi(\boldsymbol{x}_f, t_f) = 0$. In other words, the equation is solved until the reachability front reaches the target, as the very first time this happens corresponds to the minimum time of arrival.

Once the reachability front has reached the destination, it still remains to determine the optimal path to be followed by the vehicle. This information regarding the optimal controls is encoded in the value function $\phi(\boldsymbol{x}, t)$ and is given by Equation (4.45). Specifically, as outlined in Section 4.2, the forward reachability framework provides the optimal closed loop controls (or optimal policy) for the vehicle navigating *backwards* in time back to the start point. For this backwards system, the optimal closed loop controls from Equations (4.58) and (4.59) can be used to trace a time-optimal path from the destination $\boldsymbol{x}_f$ at $t = t_f$ to the start point $\boldsymbol{x}_s$. Specifically, the optimal path $\boldsymbol{X}_p^*(t)$ is computed by solving the backtracking ODE given by

$$\begin{aligned}\frac{\mathrm{d}\boldsymbol{X}_p^*}{\mathrm{d}t} &= -\boldsymbol{V}(\boldsymbol{X}_p^*(t),t) - F^*(\boldsymbol{X}_p^*(t),t)\cdot\hat{\boldsymbol{h}}^*(\boldsymbol{X}_p^*(t),t), \\ &= -\boldsymbol{V}(\boldsymbol{X}_p^*(t),t) - F_{max}\cdot\frac{\frac{\partial\phi}{\partial\boldsymbol{x}}\Big|_{(\boldsymbol{X}_p^*(t),t)}}{\left\|\frac{\partial\phi}{\partial\boldsymbol{x}}\Big|_{(\boldsymbol{X}_p^*(t),t)}\right\|},\end{aligned} \tag{4.61}$$

which is solved backward in time starting from the final condition $\boldsymbol{X}_p^*(t=t_f)=\boldsymbol{x}_f$ to the time $t=0$. Physically, it can be noted that the optimal path corresponds to that in which the vehicle always moves on the reachability front, at its maximum speed, and in a heading always normal to the front.

Furthermore, it should be noted that a history can be maintained of the controls from the optimal policy used in the backtracking process. This time history of controls forms an open-loop control law, which can be fed to the actual autonomous vehicle at the start state in order to follow the desired optimal path (forwards in time). Thus, while a closed-loop control law is used for the backwards dynamical system to determine the optimal path, this is used to define an optimal open-loop control law for the vehicle to follow forward in time.

4.4.2 Harvest-Time Optimal Path Planning with Forward Reachability

We now return to the harvesting system of interest in this work given by Equations (3.2a) and (3.2b), restated here for convenience

$$\begin{aligned}\frac{\mathrm{d}\boldsymbol{X}_p}{\mathrm{d}t} =& \boldsymbol{V}\left(\boldsymbol{X}_p(t), t\right) + F(t)\,\hat{\boldsymbol{h}}(t),\\ \frac{\mathrm{d}C_{p,i}}{\mathrm{d}t} =& \mathcal{H}_i(\boldsymbol{X}_p(t), \boldsymbol{C}_p(t), F(t), \hat{\boldsymbol{h}}(t), t), \quad i = 1, ..., d_h.\end{aligned}$$

Going forward, we will denote the joint harvesting dynamics from Equation (3.2b) equivalently in vector form as:

$$\frac{\mathrm{d}\boldsymbol{C}_p}{\mathrm{d}t} = \boldsymbol{\mathcal{H}}(\boldsymbol{X}_p(t), \boldsymbol{C}_p(t), F(t), \hat{\boldsymbol{h}}(t), t).$$

We furthermore consider the harvest-time optimal path planning problem given by Equation (3.5) and likewise restated as follows for convenience:

$$\begin{aligned}\underset{F(\cdot),\,\hat{\boldsymbol{h}}(\cdot)}{\text{minimize}} \quad & \Big[t_f, -C_{p,1}(t_f), -C_{p,2}(t_f), ..., -C_{p,d_h}(t_f)\Big]\\ \text{subject to} \quad & \boldsymbol{X}_p(0) = \boldsymbol{x}_s\\ & \boldsymbol{C}_p(0) = \boldsymbol{c}_s\\ & \boldsymbol{X}_p(t_f) = \boldsymbol{x}_f\\ & \boldsymbol{C}_p(t) > 0, \quad \forall t\\ & \frac{\mathrm{d}\boldsymbol{X}_p}{\mathrm{d}t} = \boldsymbol{V}\left(\boldsymbol{X}_p(t), t\right) + F(t)\,\hat{\boldsymbol{h}}(t)\\ & \frac{\mathrm{d}\boldsymbol{C}_p}{\mathrm{d}t} = \boldsymbol{\mathcal{H}}(\boldsymbol{X}_p(t), \boldsymbol{C}_p(t), F(t), \hat{\boldsymbol{h}}(t), t)\end{aligned} \quad .$$

We search for Pareto optimal solutions to this optimization problem of the form $(t_f^*, \boldsymbol{c}_f^*)$, where we have denoted by $\boldsymbol{c}_f^*$ the final harvested state upon reaching the destination. As we will now show, this multi-objective path planning problem can be readily solved using the concept of forward reachability in an augmented state space.

Consider the augmented state space of the dynamical system given by Equations

(3.2a) and (3.2b) consisting of the physical position of the vehicle and its harvest state: $\boldsymbol{x^a} = [\boldsymbol{x}, \boldsymbol{c}]^T \in \mathbb{R}^{d+d_h}$. Letting the initial set be $\mathcal{X}_0 = \{\boldsymbol{x}_s^{\boldsymbol{a}}\} = \{[\boldsymbol{x}_s, \boldsymbol{c}_s]^T\}$ at time $t_0 = 0$, our goal is to compute the forward reachable set of the system – the vehicle position and harvest state – using Equation (4.42). Substituting the dynamics, the reachable set's evolution is governed by:

$$\frac{\partial \phi(\boldsymbol{x}, \boldsymbol{c}, t)}{\partial t} + \max_{F, \hat{\boldsymbol{h}}} \left\{ F\, \hat{\boldsymbol{h}}^T \cdot \frac{\partial \phi}{\partial \boldsymbol{x}} + \boldsymbol{\mathcal{H}}(\boldsymbol{x}, \boldsymbol{c}, F, \hat{\boldsymbol{h}}, t)^T \frac{\partial \phi}{\partial \boldsymbol{c}} \right\} + \boldsymbol{V}(\boldsymbol{x}, t)^T \cdot \frac{\partial \phi}{\partial \boldsymbol{x}} = 0,$$
$$\phi(\boldsymbol{x}, \boldsymbol{c}, t = 0) = \phi_0, \tag{4.63}$$

where ϕ_0 corresponds to the initial condition given by the signed distance function relative to the start state $\boldsymbol{x}_s^{\boldsymbol{a}}$ in the augmented space. The maximization of the Hamiltonian in Equation (4.63), of course, will depend on the particular form of the harvesting dynamics.

With the forward reachable set evolution defined, a time optimal path to any state $\boldsymbol{x}_f^{\boldsymbol{a}} = [\boldsymbol{x}_f, \boldsymbol{c}_f]^T$ can be computed in a fashion exactly analogous to that described in Section 4.4.1. That is, we evolve the forward reachable front until the first time it touches the target state of interest in the augmented state space. The optimal controls can then be used to find the time optimal trajectory of the system through the state space. This ability to compute time optimal paths, using the forward reachable set, to any state can ultimately be extended to generate the Pareto front for the multi-objective path planning problem given by Equation (3.5). We delve deeper into how this can be done in the following section.

4.4.3 Computing the Pareto Front

Of central interest in this work is the determination of the family of Pareto optimal solutions to the multi-objective optimal path planning problem given by Equation (3.5) which can be found using the forward reachable set solved from Equation (4.63). Consider the value function $\phi(\boldsymbol{x}, \boldsymbol{c}, t)$ which encodes the reachable set's evolution. We further assume that the initial reachability front ϕ_0 is defined such that $\phi(\boldsymbol{x} =$

$\boldsymbol{x_f}, \boldsymbol{c}, t = 0) > 0,\ \forall \boldsymbol{c}$, that is, it is defined such that no harvesting state at the destination is initially reachable. This of course is not a restrictive assumption in path planning settings, where initially the start state is the only point in the augmented space that is reachable and ϕ_0 is defined to be the signed distance function relative to this state.

We have the following claim which will be central to the computation of the Pareto front:

Claim 4.4.1. *All Pareto optimal solutions* $(t^*, \boldsymbol{c}^*)$ *to the optimization problem given by Equation* (3.5) *lie in the set* $\{(t^*, \boldsymbol{c}^*) \mid \phi(\boldsymbol{x} = \boldsymbol{x}_f, \boldsymbol{c}^*, t^*) = 0\}$. *That is, all Pareto optimal solutions lie on the reachability front.*

To verify this claim, we center our attention at the target $\boldsymbol{x} = \boldsymbol{x}_f$ in the physical space. Furthermore, consider a harvesting state $\boldsymbol{c} = \hat{\boldsymbol{c}}$ and a time $t = \hat{t}$. We consider the following three cases: $\phi(\boldsymbol{x} = \boldsymbol{x}_f, \boldsymbol{c} = \hat{\boldsymbol{c}}, t = \hat{t}) > 0$, $\phi(\boldsymbol{x} = \boldsymbol{x}_f, \boldsymbol{c} = \hat{\boldsymbol{c}}, t = \hat{t}) = 0$ and $\phi(\boldsymbol{x} = \boldsymbol{x}_f, \boldsymbol{c} = \hat{\boldsymbol{c}}, t = \hat{t}) < 0$.

Case 1 ($\phi(\boldsymbol{x} = \boldsymbol{x}_f, \boldsymbol{c} = \hat{\boldsymbol{c}}, t = \hat{t}) > 0$): In this case, at the specified time $\hat{t}$, the state $[\boldsymbol{x}, \boldsymbol{c}] = [\boldsymbol{x}_f, \hat{\boldsymbol{c}}]$ is not reachable. Therefore, $(\hat{t}, \hat{\boldsymbol{c}})$ cannot be a solution to the optimization problem.

Case 2 ($\phi(\boldsymbol{x} = \boldsymbol{x}_f, \boldsymbol{c} = \hat{\boldsymbol{c}}, t = \hat{t}) = 0$): In this case, at the specified time $\hat{t}$, the state $[\boldsymbol{x}, \boldsymbol{c}] = [\boldsymbol{x}_f, \hat{\boldsymbol{c}}]$ is on the reachability front and so it is reachable. Therefore, $(\hat{t}, \hat{\boldsymbol{c}})$ can indeed be a candidate solution to the optimal path planning problem.

Case 3 ($\phi(\boldsymbol{x} = \boldsymbol{x}_f, \boldsymbol{c} = \hat{\boldsymbol{c}}, t = \hat{t}) < 0$): In this case, at the specified time $\hat{t}$, the state $[\boldsymbol{x}, \boldsymbol{c}] = [\boldsymbol{x}_f, \hat{\boldsymbol{c}}]$ is in the reachable set and so it is reachable. However, this state cannot be a Pareto optimal solution to the optimal path planning problem. To see why this is the case note that if $\phi(\boldsymbol{x} = \boldsymbol{x}_f, \boldsymbol{c} = \hat{\boldsymbol{c}}, t = \hat{t}) < 0$, then, following from the continuity of $\phi(\boldsymbol{x}, \boldsymbol{c}, t)$ (refer to Section 4.3.1) and the intermediate value theorem, there must exist some time $\tau < \hat{t}$ such that $\phi(\boldsymbol{x} = \boldsymbol{x}_f, \boldsymbol{c} = \hat{\boldsymbol{c}}, t = \tau) = 0$. This implies that a candidate solution to the multi-objective optimization problem exists

$(\tau, \hat{\boldsymbol{c}})$ that dominates $(\hat{t}, \hat{\boldsymbol{c}})$ (since such a path arrives at the destination state quicker). Thus, $(\hat{t}, \hat{\boldsymbol{c}})$ cannot be a candidate Pareto optimal solution.

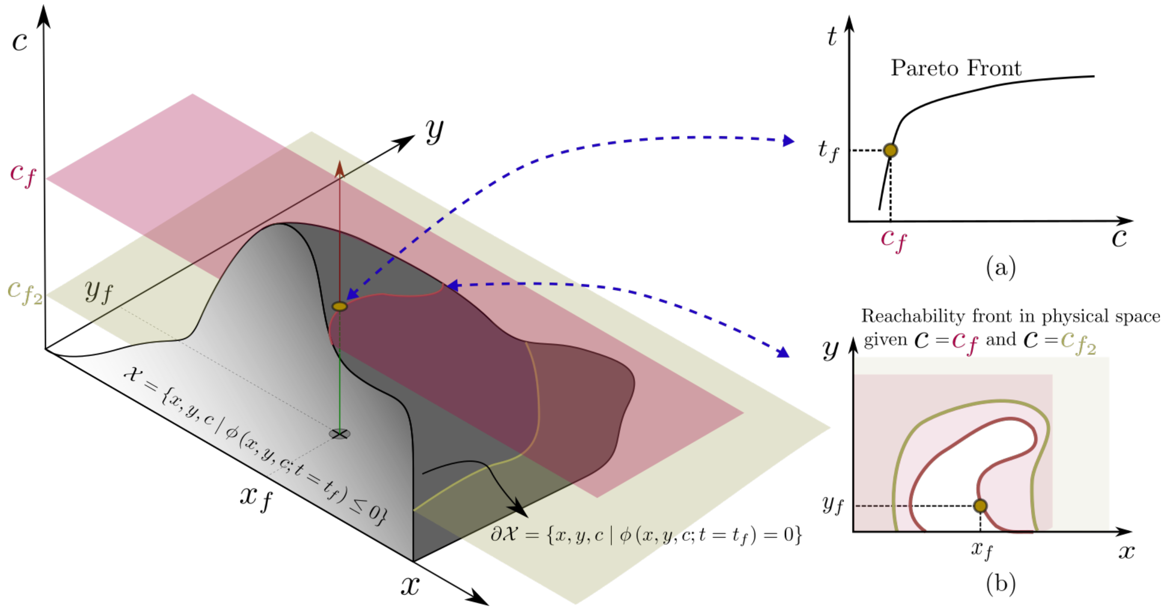
The property given by Claim 4.4.1 gives a straightforward way to now process and generate the Pareto front. Consider the time $t = \hat{t}$ and denote the set of Pareto optimal solutions found for times $t < \hat{t}$ of the form $(t^*, \boldsymbol{c}^*)$ as Θ_p. At the time $\hat{t}$, consider the set of harvest states on the reachability front $\mathcal{S}_c = \{\boldsymbol{c} \mid \phi(\boldsymbol{x} = \boldsymbol{x}_f, \boldsymbol{c}, t = \hat{t}) = 0\}$. Of this set, denote by $\mathcal{S}_{c,p}$, $\mathcal{S}_{c,p} \subseteq \mathcal{S}_c$, the set of candidate Pareto optimal harvest states. Specifically, this set holds all harvest states $\boldsymbol{c}$ that satisfy

$$\boldsymbol{c} \in \mathcal{S}_{c,p} \implies \nexists\, \tilde{\boldsymbol{c}} \in \mathcal{S}_c,\ \tilde{\boldsymbol{c}} \neq \boldsymbol{c},\ s.t.\ \tilde{\boldsymbol{c}} \succ_K \boldsymbol{c},\ K = \mathbb{R}_+^{d_h}, \tag{4.64}$$

where we have used the generalized inequality defined using the nonnegative orthant proper cone (Boyd et al., 2004). The subset of $\mathcal{S}_{c,p}$ to be added to the set of Pareto optimal solutions is then given as those states $\boldsymbol{c}$ that are not in the set Θ_p. This procedure is then repeated for all subsequent times to ultimately form the full Pareto front to the optimization problem.

Particular Case: One Harvesting Dimension

The procedure outlined above for computing the Pareto front has been presented in full generality, i.e. for an arbitrary number of harvesting dimensions. In reality, in most practical applications one, at most two, additional harvesting states can be considered. This follows directly from Bellman's curse of dimensionality, which results in the numerical computation of value functions for high dimensional systems to be prohibitively expensive (Kirk, 2004). Due to this, all cases computed in the present work consider only a single additional harvesting dimension for a vehicle navigating in a 2D physical domain (resulting in a 3D augmented state space). In this case, the procedure for computing the Pareto front simplifies as follows. Consider the time $t = \hat{t}$ and denote the set of Pareto optimal solutions computed for times $t < \hat{t}$ of the form (t^*, c^*) as Θ_p. At time $\hat{t}$, consider the set $\mathcal{S}_c = \{c \mid \phi(\boldsymbol{x} = \boldsymbol{x}_f, c, t = \hat{t}) = 0\}$. If this set is non-empty, the only candidate Pareto optimal harvest state is $\hat{c} = \max(\mathcal{S}_c)$.



(a)

(b)

Figure 4-7: A schematic demonstrating the growth of the reachability front in the augmented state space in the case of a 2D physical domain with a single harvesting state. Figure (a) shows the Pareto front which can be computed by tracking the reachability front at the destination as described in Section 4.4.3. Figure (b) shows the contours of the reachability front at different harvest state values projected onto the 2D physical space.

If $\hat{c} \notin \Theta_p$, the solution $(\hat{t}, \hat{c})$ is a Pareto optimal solution and so it can be added to Θ_p. This procedure is then repeated for all subsequent times in order to compute the full set of Pareto optimal solutions.

4.4.4 Backtracking: Computing the Optimal Path

Once the Pareto front has been generated, a Pareto optimal solution can be selected by a decision maker who can express preference for their objectives of interest, i.e. how much to tradeoff the final harvested amounts with the time of arrival. Consider a chosen Pareto optimal solution $[t^*_f, C^*_{p,1,f}, C^*_{p,2,f}, ...C^*_{p,m,f}]$ to the optimization problem given by Equation (3.5), where we have used $C^*_{p,i,f} \equiv C_{p,i}(t_f)$ to denote the final harvest state of the i^{th} field for the vehicle. Just as in the time-optimal path planning case (as described in Section 4.4.1), the optimal closed-loop controls to be followed by the system *backwards* in time are determined from Equation (4.45) and are given by the maximizers to the Hamiltonian in Equation (4.63). Specifically, using the optimal policies for the speed $F^*(\boldsymbol{x}, \boldsymbol{c}, t)$ and headings $\hat{\boldsymbol{h}}^*(\boldsymbol{x}, \boldsymbol{c}, t)$, the corresponding optimal path to be followed by the vehicle can be computed by solving the backtracking ODE

$$\frac{\mathrm{d}\boldsymbol{X}_p^a}{\mathrm{d}t} = -\,\boldsymbol{U}_*^a(\boldsymbol{X}_p^a(t), t) - \boldsymbol{V}^a\left(\boldsymbol{X}_p^a(t), t\right), \tag{4.65}$$

with the final condition $\boldsymbol{X}_p^a(t = t_f^*) = [\boldsymbol{x}_f, C_{p,1,f}^*, C_{p,2,f}^*, ...C_{p,m,f}^*]^T$ and where

$$\boldsymbol{X}_p^a(t) = \begin{bmatrix} \boldsymbol{X}_p(t) \\ \boldsymbol{C}_p(t) \end{bmatrix}, \tag{4.66a}$$

$$\boldsymbol{U}_*^a(\boldsymbol{X}_p^a(t), t) = \begin{bmatrix} F^*(\boldsymbol{X}_p^a(t), t) \cdot \hat{\boldsymbol{h}}^*(\boldsymbol{X}_p^a(t), t) \\ \boldsymbol{\mathcal{H}}(\boldsymbol{X}_p(t),\ \boldsymbol{C}_p(t),\ F^*(\boldsymbol{X}_p^a(t), t),\ \hat{\boldsymbol{h}}^*(\boldsymbol{X}_p^a(t), t),\ t) \end{bmatrix}, \tag{4.66b}$$

$$\boldsymbol{V}^a\left(\boldsymbol{X}_p^a(t), t\right) = \begin{bmatrix} \boldsymbol{V}\left(\boldsymbol{X}_p(t), t\right) \\ 0 \end{bmatrix}. \tag{4.66c}$$

Equation (4.65) is solved backward in time from $t = t_f$ to $t = 0$ to get the optimal trajectory of the system in the augmented state space. The optimal physical path (i.e. in the physical space) of the vehicle is simply found by considering the components of this trajectory corresponding to the physical space coordinates. Just as in the time-optimal case described in Section 4.4.1, the closed-loop control law for the backward dynamical system determines an optimal open-loop policy for the actual vehicle to follow (forward in time). Moreover, physically, in order to minimize the arrival time to the state of interest, these controls will always impose that the state remain on the reachability front throughout its optimal trajectory in the augmented state space (recall that all Pareto optimal solutions to the optimization problem lie on the reachability front as described in Section 4.4.3). As a result, it is sufficient to store the optimal control information, at any time t, at only those states in the augmented space that lie on the reachability front. This, in turn, results in significant memory savings.

Furthermore, we reiterate that each optimal path predicted using these equations and technique is *globally optimal*. For a given Pareto optimal solution $[t_f^*, C_{p,1,f}^*, C_{p,2,f}^*, ...C_{p,m,f}^*]$, no other path exists that attains a final harvest state of $\boldsymbol{c} = [C^*_{p,1,f}, C^*_{p,2,f}, ...C^*_{p,m,f}]$ in a lower amount of time.

4.5 Harvest-Time Optimal Path Planning: Particular Cases

In Section 4.4, the theory for harvest-time optimal path planning was presented in complete generality without considering a specific form of the harvesting dynamics. We now look to show how the presented equations simplify for some special forms of the dynamics and performance measure (cost function).

4.5.1 Energy Harvesting

In this case we consider energy-time optimal path planning. The presentation here follows closely (Doshi, 2021) and we refer the reader to this work for further details.

In the case of energy-time optimal path planning, the harvesting dynamics are given by Equation (3.3). Assuming a power law form for the energy usage function (Subramani and Lermusiaux, 2016), the energy dynamics is of the form

$$\frac{\mathrm{d}E_p}{\mathrm{d}t} = \dot{Q}(\boldsymbol{X}_p(t), t) - k \cdot F(t)^n, \tag{4.67}$$

where $k \in \mathbb{R}_{>0}$ is a proportionality constant and $n \in \mathbb{N}^+$ is a positive integer. Considering a vehicle with an initial state (initial position and energy) at $\boldsymbol{x}^a_s = [\boldsymbol{x}_s, e_s]^T$, the forward reachable set evolution is found by substituting the dynamics into Equation (4.63):

$$\begin{aligned}\frac{\partial \phi(\boldsymbol{x}, e, t)}{\partial t} + \max_{F, \hat{\boldsymbol{h}}} \left\{ F\, \hat{\boldsymbol{h}}^T \cdot \frac{\partial \phi}{\partial \boldsymbol{x}} - k \cdot F^n \frac{\partial \phi}{\partial e} \right\}& \\ + \boldsymbol{V}(\boldsymbol{x}, t)^T \cdot \frac{\partial \phi}{\partial \boldsymbol{x}} + \dot{Q}(\boldsymbol{x}, t) \cdot \frac{\partial \phi}{\partial e} &= 0, \\ \phi(\boldsymbol{x}, e, t = 0) &= \phi_0,\end{aligned} \tag{4.68}$$

where ϕ_0 is the initial condition in the augmented state space given by the signed distance function to the initial state $\boldsymbol{x}_s^a$. The maximization in Equation (4.68) can be done analytically. To do this, we first consider the optimal heading $\hat{\boldsymbol{h}}$. At any point in space and time, it is clear that, based on the properties of the Euclidean dot product, that the optimal heading is given by

$$\hat{\boldsymbol{h}}^*(\boldsymbol{x}, e, t) = \frac{\frac{\partial \phi}{\partial \boldsymbol{x}}\Big|_{(\boldsymbol{x},e,t)}}{\left\| \frac{\partial \phi}{\partial \boldsymbol{x}}\Big|_{(\boldsymbol{x},e,t)} \right\|}. \tag{4.69}$$

Substituting this heading, we are left with

$$\max_{F\in[0,F_{max}]} \underbrace{\left\{ F \left\| \frac{\partial \phi}{\partial \boldsymbol{x}} \right\| - k \cdot F^n \frac{\partial \phi}{\partial E} \right\}}_{g(F)}, \tag{4.70}$$

that is, a function only of F to be maximized. Note that the optimal F from Equation (4.70) will vary in state space, i.e. we will obtain $F^*(\boldsymbol{x}, e, t)$, due to the fact that the gradient of ϕ varies in space and time. In the following, we consider a fixed point in the state space and look to maximize the objective at that point. $g(F)$ moreover can be seen to be a polynomial, and it admits an analytical solution for the optimal speed based on different values of n considered. In the following, we consider the maxima of this expression for 2 cases: (i) $n \geq 2$ and (ii) $n = 1$.

(i) Case $n \geq 2$

For the case of $n \geq 2$, the maximization of $g(F)$ is given as follows:

$$g(F) = F \left\| \frac{\partial \phi}{\partial \boldsymbol{x}} \right\| - k \cdot F^n \cdot \frac{\partial \phi}{\partial E} \tag{4.71}$$

$$g'(F) = \left\| \frac{\partial \phi}{\partial \boldsymbol{x}} \right\| - k \cdot n \cdot \frac{\partial \phi}{\partial E} \cdot F^{n-1} \tag{4.72}$$

$$g''(F) = -k \cdot n \cdot (n-1) \cdot \frac{\partial \phi}{\partial E} \cdot F^{n-1} \tag{4.73}$$

$$g'(F) = 0 \implies \tilde{F} = \left(\frac{\left\| \frac{\partial \phi}{\partial \boldsymbol{x}} \right\|}{k \cdot n \frac{\partial \phi}{\partial E}} \right)^{\frac{1}{n-1}}. \tag{4.74}$$

Equation (4.73) implies that the polynomial $g(F)$ attains a maximum in the interval $F \in [0, F_{max}]$ only if $\frac{\partial \phi}{\partial E} > 0$. In this case, the maximum occurs at $F = \tilde{F}$ as given in Equation (4.74). In the case where this maximum is such that $\tilde{F} > F_{max}$, the monotonic nature of the polynomial $g(F)$ will result in the maximum occurring at $F = F_{max}$ when considering the extrema in the truncated domain $[0, F_{max}]$. In the case where $\frac{\partial \phi}{\partial E} \leq 0$, the maximum will be at the end points of the interval which, from Equation (4.71), can be concluded to occur at $F = F_{max}$. These conditions can all be compiled concisely to give the following for F^*, the optimal speed which maximizes $g(F)$:

$$\max g(F) = g(F^*) \tag{4.75}$$

$$\text{where } F^*(\boldsymbol{x}, e, t) = \begin{cases} F_{max} & \text{if } \frac{\partial \phi}{\partial E} \leq 0 \\ \min\left(F_{max}, \left(\frac{\left\|\frac{\partial \phi}{\partial \boldsymbol{x}}\right\|}{k \cdot n \frac{\partial \phi}{\partial E}}\right)^{\frac{1}{n-1}}\right) & \text{if } \frac{\partial \phi}{\partial E} > 0. \end{cases} \tag{4.76}$$

(ii) Case $n = 1$

In the linear energy usage model case we have that

$$g(F) = F\left(\left\|\frac{\partial \phi}{\partial \boldsymbol{x}}\right\| - k\,\frac{\partial \phi}{\partial E}\right). \tag{4.77}$$

Equation (4.77) is a linear function whose maximum can be trivially calculated for F^*, the optimal speed:

$$\max g(F) = g(F^*) \tag{4.78}$$

$$\text{where } F^*(\boldsymbol{x}, e, t) = \begin{cases} F_{max} & \text{if } \left\|\frac{\partial \phi}{\partial \boldsymbol{x}}\right\| \geq k \cdot \frac{\partial \phi}{\partial E} \\ 0 & \text{if } \left\|\frac{\partial \phi}{\partial \boldsymbol{x}}\right\| < k \cdot \frac{\partial \phi}{\partial E} \end{cases}. \tag{4.79}$$

Substituting the optimal speed $F^*(\boldsymbol{x}, e, t)$ – given by either Equation (4.79) or (4.76) based on the energy usage model – and the optimal heading, we have that the forward reachable set evolution is governed by the following PDE:

$$\begin{aligned}\frac{\partial \phi(\boldsymbol{x}, e, t)}{\partial t} + F^*(\boldsymbol{x}, e, t) \left\| \frac{\partial \phi}{\partial \boldsymbol{x}} \right\| + \boldsymbol{V}(\boldsymbol{x}, t)^T \cdot \frac{\partial \phi}{\partial \boldsymbol{x}} \\ + \left[\dot{Q}(\boldsymbol{x}, t) - k \cdot [F^*(\boldsymbol{x}, e, t)]^n \right] \cdot \frac{\partial \phi}{\partial e} &= 0 \end{aligned} \tag{4.80}$$

Once the forward solve is completed and Pareto front generated, the optimal path corresponding to a Pareto optimal solution of the form (t_f^*, e_f^*) can be found by solving the backtracking ODE. Substituting the dynamics, the backtracking ODE is of the form

$$\frac{\mathrm{d}\boldsymbol{X}_p^a}{\mathrm{d}t} = - \boldsymbol{U}_*^a(\boldsymbol{X}_p^a(t), t) - \boldsymbol{V}^a \left(\boldsymbol{X}_p^a(t), t \right), \tag{4.81}$$

with the final condition $\boldsymbol{X}_p^a(t = t_f^*) = [\boldsymbol{x}_f, e_f^*]^T$ and where

$$\boldsymbol{X}_p^a(t) = \begin{bmatrix} \boldsymbol{X}_p(t) \\ E_p(t) \end{bmatrix}, \tag{4.82a}$$

$$\boldsymbol{U}_*^a(\boldsymbol{X}_p^a(t), t) = \begin{bmatrix} F^*(\boldsymbol{X}_p^a(t), t) \cdot \hat{\boldsymbol{h}}^*(\boldsymbol{X}_p^a(t), t) \\ -k \cdot [F^*(\boldsymbol{X}_p^a(t), t)]^n \end{bmatrix}, \tag{4.82b}$$

$$\boldsymbol{V}^a \left(\boldsymbol{X}_p^a(t), t \right) = \begin{bmatrix} \boldsymbol{V} \left(\boldsymbol{X}_p(t), t \right) \\ \dot{Q}(\boldsymbol{X}_p(t), t) \end{bmatrix}. \tag{4.82c}$$

As before, Equation (4.81) is solved backward in time for any $0 \leq t \leq t_f$ to get the optimal trajectory of the system.

4.5.2 Collecting a Tracer Field

In the case of harvest-time optimal path planning when harvesting a single tracer field, the harvesting dynamics are given by Equation (3.4). Considering a vehicle with an initial state at $\boldsymbol{x}_s^a = [\boldsymbol{x}_s, c_s]^T$ (initial position and harvest state), the forward reachable set evolution is determined by substituting the dynamics into Equation (4.63):

$$\frac{\partial \phi(\boldsymbol{x}, c, t)}{\partial t} + \max_{F, \hat{\boldsymbol{h}}} \left\{ F \, \hat{\boldsymbol{h}}^T \cdot \frac{\partial \phi}{\partial \boldsymbol{x}} \right\} + \boldsymbol{V}(\boldsymbol{x}, t)^T \cdot \frac{\partial \phi}{\partial \boldsymbol{x}} + \alpha \cdot C_{field}(\boldsymbol{x}, t) \cdot \frac{\partial \phi}{\partial c} = 0,$$

$$\phi(\boldsymbol{x}, c, t = 0) = \phi_0, \quad (4.83)$$

where, as before, ϕ_0 is the initial condition in the augmented state space given by the signed distance function to the initial state $\boldsymbol{x}_s^a$. The maximization of the Hamiltonian in Equation (4.83) can be done analytically to produce the following maximizers:

$$\hat{\boldsymbol{h}}^*(\boldsymbol{x}, c, t) = \frac{\left.\frac{\partial \phi}{\partial \boldsymbol{x}}\right|_{(\boldsymbol{x},c,t)}}{\left\| \left.\frac{\partial \phi}{\partial \boldsymbol{x}}\right|_{(\boldsymbol{x},c,t)} \right\|}, \tag{4.84}$$

$$F^*(\boldsymbol{x}, c, t) = F_{max}. \tag{4.85}$$

Substituting the maximizers into Equation (4.83), we obtain the following PDE which governs the forward reachable set evolution:

$$\frac{\partial \phi(\boldsymbol{x}, c, t)}{\partial t} + F_{max} \left\| \frac{\partial \phi}{\partial \boldsymbol{x}} \right\| + \boldsymbol{V}(\boldsymbol{x}, t)^T \cdot \frac{\partial \phi}{\partial \boldsymbol{x}} + \alpha \cdot C_{field}(\boldsymbol{x}, t) \cdot \frac{\partial \phi}{\partial c} = 0. \tag{4.86}$$

Following the completion of the forward solve, the Pareto front can be generated and the optimal path corresponding to a Pareto optimal solution of the form (t_f^*, c_f^*) can be found by solving the backtracking ODE. Substituting the dynamics, the backtracking ODE is of the form

$$\frac{\mathrm{d}\boldsymbol{X}_p^a}{\mathrm{d}t} = -\,\boldsymbol{U}_*^a(\boldsymbol{X}_p^a(t),t) - \boldsymbol{V}^a\left(\boldsymbol{X}_p^a(t),t\right), \tag{4.87}$$

with the final condition $\boldsymbol{X}_p^a(t=t_f^*) = [\boldsymbol{x}_f, c_f^*]^T$ and where

$$\boldsymbol{X}_p^a(t) = \begin{bmatrix} \boldsymbol{X}_p(t) \\ C_p(t) \end{bmatrix}, \tag{4.88a}$$

$$\boldsymbol{U}_*^a(\boldsymbol{X}_p^a(t),t) = \begin{bmatrix} F^*(\boldsymbol{X}_p^a(t),t)\cdot\hat{\boldsymbol{h}}^*(\boldsymbol{X}_p^a(t),t) \\ 0 \end{bmatrix}, \tag{4.88b}$$

$$\boldsymbol{V}^a\left(\boldsymbol{X}_p^a(t),t\right) = \begin{bmatrix} V\left(\boldsymbol{X}_p(t),t\right) \\ \alpha\cdot C_{field}(\boldsymbol{X}_p(t),t) \end{bmatrix}. \tag{4.88c}$$

4.6 Summary

In this chapter, we have developed the theory for our optimal harvest-time path planning methodology. We started by first reviewing concepts from optimal control. Specifically, the standard optimal control problem was first introduced, following which we discussed how to solve this problem using the method of dynamic programming. In the case of continuous time systems, the resulting value function was shown to satisfy the Hamilton-Jacobi-Bellman PDE.

We then proceeded to the concept of forward reachability. We saw in particular that reachability ultimately boiled down to an optimal control problem but for the system of interest studied backwards in time. We then discussed how our multi-objective harvest-time optimal path planning problem could be solved with forward reachability. In particular, we discussed how the full set of Pareto optimal solutions could be determined using the reachability front, and how optimal paths could be computed by solving a backtracking ODE. Following this, we derived the governing equations for particular forms of harvesting dynamics.

Chapter 5

Numerical Implementation and Discussion

As outlined in Chapter 4, our optimal path governing equations and the corresponding methodology are composed of two primary parts. The first involves the evolution of the reachability front and corresponding Pareto front, which we now denote as the *forward solve.* The second step involves, once a Pareto optimal solution is chosen, computing the corresponding optimal path which we now denote as the *backward solve.* In this chapter, we dive into the details on how to numerically solve the corresponding partial differential equations and ordinary differential equations that are involved in these two steps. These numerical methods, together, form our path planning algorithm.

5.1 Forward Solve

The full PDE governing the evolution of the reachability front in the augmented state space $\boldsymbol{x}$, given in Equation (4.63), is of the form

$$\frac{\partial \phi}{\partial t} + H(\nabla \phi, \boldsymbol{x}, t) = 0, \qquad \phi(\boldsymbol{x}, t = 0) = \phi_0(\boldsymbol{x}). \tag{5.1}$$

As discussed in Chapter 4, Equation (5.1) falls into the class of Hamilton-Jacobi equations, and the existence, uniqueness and properties of viscosity solutions to this PDE have been extensively studied due to its broad applicability to many different fields (Osher and Fedkiw, 2005; Evans, 2010). Here, we consider a Hamiltonian function which, in general, takes as input the components of the gradient $\nabla\phi \in \mathbb{R}^d$ and may be a function of space $\boldsymbol{x} \in \mathbb{R}^d$ and time. Note that throughout this chapter, the gradient of ϕ will be represented compactly as $\nabla\phi$ rather than the notation used in Chapter 4.

Remark 2. *Our focus in this subsection (Section 5.1) will be in solving a general Hamilton-Jacobi equation given by Equation* (5.1) *in a d-dimensional space defined by a state variable* $\boldsymbol{x}$*. It should be understood that what we present is applicable to the Hamilton-Jacobi equations resulting from harvest-time optimal path planning if we consider* $\boldsymbol{x}$ *to be the augmented state (i.e. if we consider* $\boldsymbol{x}$ *to hold the combined the physical state and the harvest state). In this case, the dimension "d" we consider in this chapter would have to be equal to sum of the physical and harvesting dimensions (i.e. d in this chapter does not only correspond to the physical dimension as was done in Chapter 4). Moreover, due to this general treatment, any use of the word "space" or "spatial" (when considering* $\boldsymbol{x}$ *and its corresponding gradient components) should be naturally understood to correspond to the augmented space when applying these schemes to the harvest-time optimal path planning PDEs from Chapter 4.*

Several options exist for numerically discretizing and computing a viscosity solution to a given Hamilton-Jacobi equation, ranging from Finite Volume methods to even high-order discontinuous Galerkin methods (Lomax et al., 2013; Hesthaven and Warburton, 2007). In this work, high-order finite difference methods have been used to resolve solutions on structured, uniform, rectangular meshes. Our present code, in particular, has been built on top of an open source toolbox of level set methods (Mitchell, 2008). In what follows, we start by breaking down how to numerically solve a general Hamilton-Jacobi equation with high-order finite differences using the method of lines. In particular, we begin by outlining appropriate ways to perform the spatial discretization in forming what is known as the numerical Hamiltonian. Following this, we address options to use for the temporal discretization. The material presented in this section draws heavily from (Osher and Fedkiw, 2005; Shu, 2007) and we refer the reader to these references for further details.

5.1.1 Spatial Discretization: Monotone Schemes

Consider the general Hamilton-Jacobi equation given by (5.1). The semi-discretized version of this PDE (i.e. discretizing only space) is of the form

$$\frac{\partial \phi_{\boldsymbol{i}}}{\partial t} + \hat{H}(\nabla^{+}\phi_{\boldsymbol{i}}, \nabla^{-}\phi_{\boldsymbol{i}}, \boldsymbol{x}_{\boldsymbol{i}}, t) = 0, \tag{5.2}$$

where $\hat{H}$ is a numerical approximation of the Hamiltonian – called a numerical Hamiltonian. Here, $\phi_{\boldsymbol{i}}$ is the value of the discretized field $\phi(\boldsymbol{x}, t)$ at a given node $\boldsymbol{i}$ in the state space. Note that we denote by $\phi_{\boldsymbol{i}}$ the value of ϕ at the node i – it is not a vector. Furthermore, $\boldsymbol{i}$ corresponds to a d-dimensional multi-index of the form $\boldsymbol{i} = (i_1, i_2, ..., i_d)$, i.e. a d-tuple of non-negative integers, corresponding to a node in the discrete finite-difference mesh, while $\boldsymbol{x}_{\boldsymbol{i}}$ is the position of the node in this discrete state space. Finally, $\nabla^{+}\phi_{\boldsymbol{i}} \in \mathbb{R}^d$ and $\nabla^{-}\phi_{\boldsymbol{i}} \in \mathbb{R}^d$ are one-sided discrete approximations to the spatial derivatives at the corresponding node in the mesh. These space derivatives can be, moreover, discretized using either first-order one-sided differencing or higher-order accurate ENO or WENO schemes (Osher and Fedkiw, 2005). For instance, in the 2D case, the first-order one-sided discrete approximations of the gradient at the (i, j) node in the mesh, $\nabla^{+}\phi_{(i,j)} = [(\phi_x^{+})_{(i,j)}, (\phi_y^{+})_{(i,j)}]^T$ and $\nabla^{-}\phi_{(i,j)} = [(\phi_x^{-})_{(i,j)}, (\phi_y^{-})_{(i,j)}]^T$, are given as

$$(\phi_x^{+})_{(i,j)} = \frac{\phi_{(i+1,j)} - \phi_{(i,j)}}{\Delta x}, \quad (\phi_x^{-})_{(i,j)} = \frac{\phi_{(i,j)} - \phi_{(i-1,j)}}{\Delta x}, \tag{5.3}$$

and

$$(\phi_y^{+})_{(i,j)} = \frac{\phi_{(i,j+1)} - \phi_{(i,j)}}{\Delta y}, \quad (\phi_y^{-})_{(i,j)} = \frac{\phi_{(i,j)} - \phi_{(i,j-1)}}{\Delta y}, \tag{5.4}$$

where Δx and Δy correspond to the uniform spacing of the finite difference grid.

The numerical Hamiltonian, $\hat{H}$, is required to be consistent with the Hamiltonian H in the sense that

$$\hat{H}(\nabla\phi_{\boldsymbol{i}}, \nabla\phi_{\boldsymbol{i}}, \boldsymbol{x}_{\boldsymbol{i}}, t) = H(\nabla\phi_{\boldsymbol{i}}, \boldsymbol{x}_{\boldsymbol{i}}, t). \tag{5.5}$$

$\hat{H}$ is furthermore said to be monotone if it is monotonically non-decreasing in the $\nabla^{-}\phi_{\boldsymbol{i}}$ arguments and monotonically non-increasing in the $\nabla^{+}\phi_{\boldsymbol{i}}$ arguments (Shu, 2007). Equation (5.2), equipped with a monotone numerical Hamiltonian, is denoted as a monotone scheme (Shu, 2007).

Example: 2D Domain

To solidify the above concepts, consider a Hamilton-Jacobi equation to be solved on a 2D spatial domain $\boldsymbol{x} \in \mathbb{R}^2$:

$$\frac{\partial\phi}{\partial t} + H(\phi_x, \phi_y, x, y, t) = 0, \qquad \phi(x, y, t = 0) = \phi_0(x, y). \tag{5.6}$$

The numerical Hamiltonian, taking as input the one-sided approximations of the spatial derivatives, is of the form $\hat{H}(\phi_x^-, \phi_x^+, \phi_y^-, \phi_y^+, x, y, t)$ and is consistent provided that $\hat{H}(\phi_x, \phi_x, \phi_y, \phi_y, x, y, t) = H(\phi_x, \phi_y, x, y, t)$. Furthermore, the numerical Hamiltonian is monotone provided that it is monotonically non-decreasing in the first and third arguments, and monotonically non-increasing in the second and fourth arguments.

Monotone schemes are key to numerically solving Hamilton-Jacobi equations as a result of several beneficial properties. These properties include provable convergence of numerical solutions to the viscosity solution of the corresponding Hamilton-Jacobi equation (Shu, 2007). Furthermore, monotone schemes can be shown to be stable in the L^∞ norm (Shu, 2007). Several different monotone schemes have been rigorously studied and tested in various applications, and we outline some of these in what follows. For details, we refer to (Shu, 2007; Osher and Fedkiw, 2005).

Lax-Friedrichs Schemes

The Lax-Friedrichs scheme is a monotone scheme that approximates the numerical Hamiltonian as follows:

$$\hat{H}^{LF}(\nabla^{+}\phi_{\boldsymbol{i}}, \nabla^{-}\phi_{\boldsymbol{i}}, \boldsymbol{x}_{\boldsymbol{i}}, t) = H\left(\frac{\nabla^{+}\phi_{\boldsymbol{i}} + \nabla^{-}\phi_{\boldsymbol{i}}}{2}, \boldsymbol{x}_{\boldsymbol{i}}, t\right) - \sum_{k=1}^{d} \alpha_k \left(\frac{\nabla_k^{+}\phi_{\boldsymbol{i}} - \nabla_k^{-}\phi_{\boldsymbol{i}}}{2}\right). \tag{5.7}$$

In Equation (5.7), $\nabla_k^{+}\phi_{\boldsymbol{i}}$ and $\nabla_k^{-}\phi_{\boldsymbol{i}}$ correspond to the k^{th} component of the one-sided spatial gradient approximations at the grid point $\boldsymbol{i}$ in the mesh. The parameters α_k are dissipation coefficients that control the amount of numerical viscosity and are given as

$$\alpha_k = \max_{\boldsymbol{p} \in I} |H_k(\boldsymbol{p}, \boldsymbol{x}_{\boldsymbol{i}}, t)|, \tag{5.8}$$

where H_k is the partial derivative of the Hamiltonian with respect to the k^{th} component of the gradient $\nabla\phi$. The maximization in Equation (5.8) is performed over the gradient arguments in the Hamiltonian function. The domain, I, over which this maximization is performed is important. In the traditional Lax-Friedrichs scheme (also commonly referred to as the Global Lax-Friedrichs scheme), this domain is given by:

$$\begin{aligned} I = &\left[\min_{\boldsymbol{i}}(\nabla_1^{-}\phi_{\boldsymbol{i}}, \nabla_1^{+}\phi_{\boldsymbol{i}}), \max_{\boldsymbol{i}}(\nabla_1^{-}\phi_{\boldsymbol{i}}, \nabla_1^{+}\phi_{\boldsymbol{i}})\right] \times \\ &\qquad \left[\min_{\boldsymbol{i}}(\nabla_2^{-}\phi_{\boldsymbol{i}}, \nabla_2^{+}\phi_{\boldsymbol{i}}), \max_{\boldsymbol{i}}(\nabla_2^{-}\phi_{\boldsymbol{i}}, \nabla_2^{+}\phi_{\boldsymbol{i}})\right] ... \end{aligned} \tag{5.9}$$

In other words, I corresponds to a domain of the form $I = [l_1, r_1] \times [l_2, r_2] ... \times [l_d, r_d] \subset \mathbb{R}^d$, where the limits of this domain are dictated by the minimum and maximum values of the one sided derivatives over the *whole* discrete domain, i.e. by performing the minimization and maximization for the values of the one-sided derivatives over all the grid points in the mesh (Osher and Fedkiw, 2005).

The value of the dissipation coefficients is critical to the quality of the solution obtained. Increasing α_k will help with ensuring stability of the scheme, however it will also produce a more dissipative solution. It is therefore desirable to pick as small of a value for α_k as possible without inducing oscillations or other nonphysical phenomena into the numerical solution.

The decision to perform the maximization in Equation (5.9) over the whole domain for the traditional Lax-Friedrichs scheme typically produces large values for the dissipation coefficients and correspondingly more dissipative solutions. To address this, various variants of the Lax-Friedrichs scheme have been developed. Some of the key ones are given as follows:

Local Lax-Friedrichs (LLF) Scheme The LLF scheme performs the maximization for the dissipation coefficients at each grid point using local gradient information. Specifically, the domain I chosen for the maximization in Equation (5.8) is no longer globally specified but varies locally from grid point to grid point. Consider in particular the computation of the dissipation coefficient α_m at some grid point $\boldsymbol{j}$. $I_{m,\boldsymbol{j}}$, the domain over which to perform the maximization, is once again of the form

$$I_{m,\boldsymbol{j}} = [l_1, r_1] \times [l_2, r_2] ... \times [l_m, r_m] \times ... \times [l_d, r_d] \subset \mathbb{R}^d, \tag{5.10}$$

where now all limits are set identically to the Global Lax-Friedrichs scheme as shown in Equation (5.9) *except* for l_m and r_m which are determined locally as

$$l_m = \min\left(\nabla_m^- \phi_{\boldsymbol{j}}, \nabla_m^+ \phi_{\boldsymbol{j}}\right), \quad r_m = \max\left(\nabla_m^- \phi_{\boldsymbol{j}}, \nabla_m^+ \phi_{\boldsymbol{j}}\right). \tag{5.11}$$

The LLF scheme can additionally be shown to be a monotone scheme and it results in less dissipation than its global counterpart (Shu, 2007).

Local Local Lax-Friedrichs (LLLF) Scheme The LLLF Scheme extends the LLF scheme one step further by choosing the domain I for the maximization in Equation (5.8) using *solely* local information at a given grid point. Consider once again the computation of the dissipation coefficient α_m at a grid point $\boldsymbol{j}$. The domain $I_{m,\boldsymbol{j}}$ to perform the maximization is once again of the form $I_{m,\boldsymbol{j}} = [l_1, r_1] \times [l_2, r_2] ... \times$

$[l_d, r_d] \subset \mathbb{R}^d$, where now we have:

$$\begin{aligned} l_i &= \min\left(\nabla_i^- \phi_{\boldsymbol{j}}, \nabla_i^+ \phi_{\boldsymbol{j}}\right), \quad i = 1, 2, ...d, \\ r_i &= \max\left(\nabla_i^- \phi_{\boldsymbol{j}}, \nabla_i^+ \phi_{\boldsymbol{j}}\right), \quad i = 1, 2, ...d. \end{aligned} \tag{5.12}$$

While LLLF uses even less numerical dissipation, often times the dissipation values it produces is not adequate enough to avoid numerical instabilities (Osher and Fedkiw, 2005). Furthermore, LLLF is *not* a monotone scheme (Shu, 2007).

Additional Schemes

Several other popular schemes have also been thoroughly studied. The Roe-Fix Scheme and Godunov Scheme, in particular, are additional tried and tested approaches to solve general Hamilton-Jacobi equations (Osher and Fedkiw, 2005). Several monotone schemes have also been developed for Hamiltonians of specific forms. The Osher-Sethian scheme is one such option for cases where the Hamiltonians are of the form $H(\phi_x, \phi_y) = f(\phi_x^2, \phi_y^2)$ (shown here for the 2D case) where f is some monotone function of each argument (Shu, 2007).

Remark: Level Set Methods

Plenty of work has also been done to apply these techniques to solve more specific forms of Hamilton-Jacobi equations. One such class of problems falls in the field of dynamically evolving surfaces. Indeed, it should be noted that our problem of tracking the evolution of the reachability front comes under this class of problems. While several options exist for surface tracking, Eulerian methods for solving this problem have been shown to be quite robust. These methods, known as Level Set Methods, look to evolve a surface by representing it implicitly as the isocontour (or level set) of some function. The surface is then evolved by solving PDEs that govern the dynamics of this function. The PDEs governing the dynamics, moreover, are Hamilton-Jacobi equations. Several different types of dynamics have been studied including motions of surfaces under basic convection, motion by mean curvature and surfaces moving normal to themselves, to name a few. We refer interested readers to (Osher and Fedkiw, 2005; Sethian, 1999) for further details.

5.1.2 Temporal Discretization

The previous section outlined the spatial part of the discretization of Hamilton-Jacobi PDEs using monotone schemes. This semi-discretized system requires a time-marching scheme as do ODE systems. Here, we evolve each discrete solution value at all nodes in the mesh and the semi-discretized system is of the general form

$$\frac{\mathrm{d}\boldsymbol{\phi}}{\mathrm{d}t} = \boldsymbol{f}(t, \boldsymbol{\phi}), \tag{5.13}$$

where $\boldsymbol{\phi}$ is an augmented vector holding the discrete values of ϕ on the mesh. Equation (5.13) is an ODE system resulting from discretizing the spatial derivative operators and incorporating the boundary conditions.

Numerous options exist for performing the temporal marching (Lermusiaux, 2015). Higher-order in time temporal schemes are of course desirable in the case where accurate numerical solutions are needed. In solving Hamilton-Jacobi equations, total variation diminishing (TVD) Runge-Kutta (RK) methods in particular have shown to be quite promising. These TVD RK schemes effectively ensure that no spurious oscillations arise from higher-order accurate temporal discretizations provided oscillations are not produced in the basic forward Euler building block of each scheme (Osher and Fedkiw, 2005).

Prior to introducing these schemes, it is instructive to review TVD and total variation bounded (TVB) schemes. Consider a discrete scalar solution $\boldsymbol{u}$ in 1D. The total variation of this solution is defined as (Shu and Osher, 1988):

$$TV(\boldsymbol{u}) = \sum_j |u_{j+1} - u_j|. \tag{5.14}$$

A scheme is said to be TVD then provided that

$$TV(\boldsymbol{u}^{n+1}) \leq TV(\boldsymbol{u}^{n}), \tag{5.15}$$

where we have used the standard convention of superscripts denoting the value of the discrete solution at a given time step (Shu and Osher, 1988). Furthermore, a scheme is said to be TVB in a time interval $0 \leq t \leq T$ if

$$TV(\boldsymbol{u}^{n}) \leq B, \tag{5.16}$$

for all n and Δt satisfying $0 \leq n \cdot \Delta t \leq T$ and some B that depends only on $TV(\boldsymbol{u}^{0})$ (Shu and Osher, 1988). Natural extensions exist to additionally extend these definitions to higher dimensions, and for further information on this and other details we refer the reader to (Shu and Osher, 1988).

We now outline some standard explicit higher-order TVD RK schemes. A key assumption in these schemes is that the forward Euler discretization of Equation (5.13) is TVD. As we will see, higher-order schemes are defined by taking simple forward Euler steps and then taking a convex combination of the intermediary solutions. The resulting higher-order TVD RK scheme is then TVD provided the forward Euler step itself is TVD (Osher and Fedkiw, 2005).

First-Order TVD RK

The first-order TVD RK scheme is simply identical to a forward Euler discretization. In particular, considering $\boldsymbol{\phi}^{n}$ at time t^{n}, the solution $\boldsymbol{\phi}^{n+1}$ at the next time-step, $t^{n} + \Delta t$, is given as (Osher and Fedkiw, 2005):

$$\frac{\boldsymbol{\phi}^{n+1} - \boldsymbol{\phi}^{n}}{\Delta t} = \boldsymbol{f}(t^{n}, \boldsymbol{\phi}^{n}). \tag{5.17}$$

Second-Order TVD RK

The second-order accurate TVD RK scheme involves first taking a forward Euler step to time $t^n + \Delta t$:

$$\frac{\boldsymbol{\phi}^{n+1} - \boldsymbol{\phi}^n}{\Delta t} = \boldsymbol{f}(t^n, \boldsymbol{\phi}^n). \tag{5.18}$$

Then, a subsequent forward Euler step is completed to evolve the solution to $t^n + 2\Delta t$:

$$\frac{\boldsymbol{\phi}^{n+2} - \boldsymbol{\phi}^{n+1}}{\Delta t} = \boldsymbol{f}(t^{n+1}, \boldsymbol{\phi}^{n+1}). \tag{5.19}$$

This is finally followed by an averaging step given by:

$$\boldsymbol{\phi}^{n+1} = \frac{1}{2}\boldsymbol{\phi}^n + \frac{1}{2}\boldsymbol{\phi}^{n+2}, \tag{5.20}$$

which is a second-order accurate solution for ϕ at time $t^n + \Delta t$ (Osher and Fedkiw, 2005).

Third-Order TVD RK

The third-order accurate TVD RK scheme involves first taking a forward Euler step to time $t^n + \Delta t$:

$$\frac{\boldsymbol{\phi}^{n+1} - \boldsymbol{\phi}^n}{\Delta t} = \boldsymbol{f}(t^n, \boldsymbol{\phi}^n). \tag{5.21}$$

Then, a subsequent forward Euler step is completed to evolve the solution to $t^n + 2\Delta t$:

$$\frac{\boldsymbol{\phi}^{n+2} - \boldsymbol{\phi}^{n+1}}{\Delta t} = \boldsymbol{f}(t^{n+1}, \boldsymbol{\phi}^{n+1}). \tag{5.22}$$

An averaging step is then done to get an approximation of ϕ at time $t^n + \frac{1}{2}\Delta t$:

$$\boldsymbol{\phi}^{n+\frac{1}{2}} = \frac{3}{4}\boldsymbol{\phi}^n + \frac{1}{4}\boldsymbol{\phi}^{n+2}. \tag{5.23}$$

An Euler step is then taken again to yield

$$\frac{\boldsymbol{\phi}^{n+\frac{3}{2}} - \boldsymbol{\phi}^{n+\frac{1}{2}}}{\Delta t} = \boldsymbol{f}(t^{n+\frac{1}{2}}, \boldsymbol{\phi}^{n+\frac{1}{2}}). \tag{5.24}$$

which gives an approximate solution for time $t^n + \frac{3}{2}\Delta t$. This is followed by a final averaging step given as

$$\boldsymbol{\phi}^{n+1} = \frac{1}{3}\boldsymbol{\phi}^{n} + \frac{2}{3}\boldsymbol{\phi}^{n+\frac{2}{3}}, \tag{5.25}$$

which is a third-order accurate approximation for the solution ϕ at time $t^n + \Delta t$ (Osher and Fedkiw, 2005).

Higher-Order Schemes

Additional fourth-order, fifth-order (and even higher order) accuracy TVD schemes also exist, for which we refer the interested reader to (Osher and Fedkiw, 2005; Shu and Osher, 1988).

5.1.3 Discussion

We now outline details regarding how these schemes were used to solve some of the specific forms of the Hamilton-Jacobi equations encountered in this work:

Energy Harvesting The PDE governing the reachability front's evolution in the case of energy-time optimal path planning is given by Equation (4.80). To numerically solve this Hamilton-Jacobi PDE, we used the Local Lax-Friedrichs scheme. For computing the dissipation coefficients needed in the scheme, a closed form solution for the maximization of the partial derivatives of the Hamiltonian is unfortunately not available. To remain computationally efficient, in this work a simple grid search has been used to solve the maximization problem for obtaining an estimate of the dissipation parameter needed for the Local Lax-Friedrichs scheme. This has not caused any issues and has proved to be quite effective as we show in the following examples. Of course, we note that in the case of numerical instabilities, the artificial dissipation can be increased with the drawback of obtaining a correspondingly more dissipative, albeit stable, solution. We plan to investigate more robust approaches to compute the dissipation coefficients in future works.

Collecting a Tracer Field In the case of planning paths for harvesting a tracer field, the reachability front satisfies the Hamilton-Jacobi PDE given by Equation (4.86). While Lax-Friedrichs schemes could be used to solve this PDE, an alternate approach was followed. In particular, this PDE can be noted to be effectively of the same form as that presented in (Osher and Sethian, 1988), which describes an interface with motion in the normal direction along with an external velocity field (Osher and Fedkiw, 2005; Sethian, 1999). Using the normal direction front propagation PDE (Osher and Sethian, 1988) as reference, and following (Sethian, 1999), Equation (4.86) was discretized by treating the normal motion term $F_{max} \ \|\nabla \phi\|$ using a Godunov scheme, as outlined in (Osher and Fedkiw, 2005), while the passive advection term was discretized through upwinding.

5.1.4 Additional Numerical Details

Boundary Conditions

For boundaries of the physical domain ($x_i = x_{i,min}$ or $x_i = x_{i,max}$, $i = 1, ..., d_{physical}$) open to inlets/outlets or side walls, zero normal gradient (Neumann) boundary conditions were used as described in (Lolla et al., 2014b). Islands, obstacles and forbidden regions were furthermore treated by following the masking procedure outlined in (Lolla et al., 2014b, 2015). Specifically, at all grid points in the physical space lying in islands, obstacles or forbidden regions, the dynamics of the value function at those nodes was set to

$$\frac{\mathrm{d}\phi_i}{\mathrm{d}t} = 0, \tag{5.26}$$

for the semi-discretized system. For boundaries in the harvesting domain, zero Neumann boundary conditions were used at the minimum harvest value ($c = c_{i,min}$, $i = 1, ..., d_h$) since, by setting $c_{i,min} = 0, \ i = 1, ..., d_h$, this allowed the ability to enforce the non negativity condition $\boldsymbol{C_p}(t) > 0, \ \forall t$ in Equation (3.5). At the max harvesting state boundaries in the state space ($c = c_{i,max}$, $i = 1, ..., d_h$) care was needed based on the harvesting dynamics considered. For the cases considered in this particular work, we have the following:

Energy Harvesting In the case of the harvesting dynamics given by Equation (3.3) for energy harvesting, the vehicle's "filled battery" state must be modelled for the max harvest state boundary in the state space. This was enforced by masking along the energy domain, analogously to the time-domain procedure described in (Lolla et al., 2014b, 2015) for the case of islands, obstacles and forbidden regions). That is, energy states *above* the filled battery state were considered to be a forbidden region in the augmented state space. As a result of this masking procedure, any arbitrary Dirichlet or Neumann boundary condition could be used at the $c = c_{max}$ boundary in the energy harvest dimension since the mask resulted in it having no effect on the overall solution.

Collecting a Tracer Field For collecting a tracer field, the harvesting dynamics were given by Equation (3.4). Here, since the vehicle could only increase its harvesting state by collecting until its max capacity (i.e. the harvesting state could not decrease), it was sufficient to use a zero Neumann boundary condition at the boundary of the max harvest value in the state space.

Reinitialization

As outlined in Chapter 4, our path planning methodology is built around tracking the evolution of the reachability front. While the exact value of the value function ϕ used to encode the reachable set has physical significance, ultimately the zero level set of this value function is all that is needed to solve our multi-objective path planning problem of interest. Due to this, it is often convenient to employ numerical "remedies" that help with numerically evolving this zero level set. One of these techniques is reinitialization (Lolla et al., 2012, 2014b,c).

As discussed in Chapter 4, the value function ϕ is initialized always as a signed distance function from an initial set of interest. Throughout the forward evolution, however, this function will naturally drift away from this signed distance property. Specifically, ϕ can develop noisy features and steep gradients that are challenging to handle using finite-difference schemes. As a result of this, it is recommended to use reinitialization, which involves occasionally resetting ϕ to be a signed distance function from the $\phi = 0$ isocontour throughout a solve. A reinitialization procedure aims to effectively keep, at the time it is applied, the $\phi = 0$ isocontour fixed while adjusting all the $\phi < 0$ and $\phi > 0$ regions so that the updated values there form a signed distance from the $\phi = 0$ level set. While numerous approaches for this have been proposed (Osher and Fedkiw, 2005), one of the more effective ones employs solving what is known as the reinitialization equation, given as

$$\frac{\partial \phi}{\partial t} + S(\phi_0) \cdot \big(|\nabla \phi| - 1 \big) = 0, \tag{5.27}$$

to steady state. In Equation (5.27), $S(\phi_0)$ is a sign function taken as 1 wherever $\phi_0 > 0$, -1 where $\phi_0 < 0$ and 0 on the interface ($\phi = 0$). Furthermore, ϕ_0 corresponds to the initial function to be reinitialized. At steady state ϕ no longer changes, thus implying $\frac{\partial \phi}{\partial t} = 0$. This, in turn, reduces Equation (5.27) to $S(\phi_0) \cdot \big(|\nabla \phi| - 1 \big) = 0$ from which it follows that ϕ is indeed a signed distance function with the same isocontour $\phi = 0$ as the original function $\phi_0 = 0$ as desired.

In a full forward solve, a user can decided to perform the reinitialization procedure at a certain frequency of time steps of their choice. In discretizing and solving Equation (5.27), Godunov schemes are recommended to discretize the normal motion terms. Additionally, low accuracy discretizations of derivatives in space and time are also often sufficient in solving the given equation, thus making the procedure numerically tractable.

For many more details on reinitialization, its benefits and the nuances regarding its implementation, we refer the reader to (Osher and Fedkiw, 2005).

5.2 Backward Solve

Once the forward solve has been used to generate the Pareto front, the backtracking equation, Equation (4.65), is solved to determine the final optimal path for a given Pareto optimal solution. This is a system of ODEs which can be discretized and solved using standard explicit or implicit schemes, as outlined in (Lolla et al., 2014b; Kulkarni and Lermusiaux, 2020).

In this work, an explicit first order time integration scheme has been used to backtrack all optimal paths. Denoting the augmented state of the vehicle at time t as $(\boldsymbol{X}_p^a)_t$, the discretized form of Equation (4.65) is given as

$$\frac{(\boldsymbol{X}_p^a)_{t+\Delta t} - (\boldsymbol{X}_p^a)_t}{\Delta t} = -\,\boldsymbol{U}_*^a(t+\Delta t) - \boldsymbol{V}^a\left((\boldsymbol{X}_p^a)_{t+\Delta t}, t+\Delta t\right). \qquad (5.28)$$

It is important to note that Equation (5.28) is indeed explicit, as the backtracking equation is solved *backward* in time starting from the final state $(\boldsymbol{X}_p^a)_{t_f}$ at the final time t_f. Therefore, at each time step, $(\boldsymbol{X}_p^a)_{t+\Delta t}$ is known whereas $(\boldsymbol{X}_p^a)_t$ is the unknown to be computed. Throughout the backtrack, the state must remain on the reachability front at all times, however numerical errors have a tendency to pull it off the front. This is an issue which can be tackled by additionally projecting the state onto the reachability front at each timestep (Lolla et al., 2014b) or ideally by using dual schemes (Kulkarni and Lermusiaux, 2020).

5.3 Summary

In this chapter, we have presented the numerical algorithms needed to numerically solve the PDEs and ODEs that arise in our path planning method. Specifically, we addressed how monotone schemes can be used to find viscosity solutions to Hamilton-Jacobi equations. The Lax-Friedrichs scheme, a particular monotone scheme, was reviewed in detail. We then addressed relevant numerical details to solve specific forms of Hamilton-Jacobi equations arising for different harvesting applications. Finally, we showed how optimal paths for chosen Pareto optimal solutions can be computed by solving the backtracking ODE using explicit time-stepping schemes.

Chapter 6

Applications

In this chapter, we illustrate our theory and schemes on numerical cases and simulations. In the first example, we consider a canonical case of a vehicle having to harvest a field in a halfspace region prior to reaching the destination. This case admits a semi-analytical solution which therefore makes it useful to validate our methodology. In the second case, we consider energy-time optimal path planning for a vehicle navigating in a double-gyre flow field representative of near-surface ocean circulation at mid-latitude regions. We follow this by considering an application to path planning for optimal harvesting of harmful algae blooms. Finally, we look at applying our approach to a next generation approach for optimal farm fishing using autonomously moving aquaculture farms.

In all applications that follow, when not specified, all values should be assumed to be dimensionless.

6.1 Harvest-Time Optimal Path Planning: Halfspace Harvest Region Benchmark Case

In this section, we consider the case of harvest-time optimal path planning in a simple idealized case of a vehicle navigating in a steady 2D physical space. In this idealized case, all values provided are dimensionless. We consider a simple harvesting dynamics

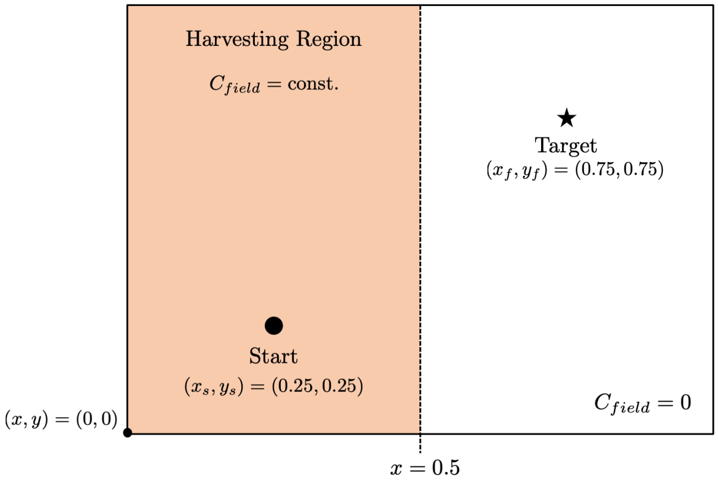


Figure 6-1: A schematic showing the parameters for the benchmark case of interest. This case, in particular, consists of a vehicle navigating through a domain with a steady harvest field covering a halfspace.

equation of the form of Equation (3.4), restated here for convenience,

$$\frac{\mathrm{d}C_p}{\mathrm{d}t} = \alpha \cdot C_{field}(\boldsymbol{X_p}(t), t), \tag{6.1}$$

where a vehicle is tasked with harvesting some tracer field. The harvest region covers the halfspace defined by $\{(x, y) \mid x \leq 0.5\}$ and is steady. The vehicle is furthermore tasked with navigating from a start point at position $(x, y) = (0.25, 0.25)$ to the fixed target at $(x, y) = (0.75, 0.75)$ while travelling at a speed less than or equal to its max of $F_{max} = 0.5$. In the harvest region, the harvest field is given by a constant value of $C_{field}(\boldsymbol{x}, t) = c_{Harvest\ Region} = 0.5$, and the collection rate for the vehicle is set to a constant value of $\alpha = 1.0$ (Figure 6-1 summarizes all these case parameters). The vehicle starts with an initial harvest state of 0. Furthermore, we reiterate that we aim to solve the multi-objective path planning problem, given by Equation (3.5), of both maximizing the total amount of the field harvested while also minimizing the time of arrival.

Semi-analytical solution The halfspace case provides a good benchmark as it admits a semi-analytical solution that can be used to test the validity of our methodology. We thus start by outlining how to determine this semi-analytical solution. For this problem of interest, the augmented state space is given by (x, y, c), where c corresponds to the harvested state of the vehicle (i.e. the amount of material harvested). Consider an initial spherical reachability front centered at (x_c, y_c, c_c) in the augmented state space with initial radius r_0 (the initial reachable set is given by the interior of this sphere). In this case of interest, we have that $(x_c, y_c, c_c) = (0.25, 0.25, 0)$ so that the initial reachable set is given by a hemisphere (since in the augmented state space, we consider only harvest states satisfying $c \geq 0$). Our aim, recall, is to determine Pareto optimal paths originating from anywhere in this reachable set to the destination that minimize travel time and maximizes the final harvested state of the vehicle. Intuitively, it is clear that any such Pareto optimal path must originate from a state on the boundary of the reachability set – the reachability front. The unknown start point of the path from the spherical front can be parametrized using two degrees of freedom as:

$$\begin{aligned}
x_i(\alpha, \beta) &= x_c + r_0 \, sin(\beta) \, cos(\alpha) \\
y_i(\alpha, \beta) &= y_c + r_0 \, sin(\beta) \, sin(\alpha) \\
c_i(\alpha, \beta) &= c_c + r_0 \, cos(\beta). \qquad (6.2)
\end{aligned}$$

The variables α and β correspond to the angular positions of the start point on the front, where α denotes the polar angle and β the azimuthal angle as defined in the standard spherical coordinate system.

We look to build now the set of Pareto optimal solutions Θ_p of the form (t_f^*, c_f^*) – where t_f^* is the arrival time of the vehicle and c_f^* is the final harvest state of the vehicle – as follows. We will consider a fixed $c_f^* = \gamma_c$ value for the final harvest state and then attempt to find a time-optimal path for the vehicle that attains this state. By fixing the final harvest state, we are able to reduce the number of objectives to formulate a single objective optimization problem involving just minimizing the travel time. If a feasible path is found, corresponding to an arrival time of $t_f^* = \gamma_t$, then $(t_f^*, c_f^*) = (\gamma_t, \gamma_c)$ is a Pareto optimal solution belonging to the set Θ_p. This process is then repeated for larger and larger γ_c values in order to build the full Pareto front for the multi-objective optimization problem.

Consider now the determination of the time-optimal path for a fixed final harvest state value of $c_f^* = \gamma_c$. The optimal path of the vehicle can be found by considering its journey through the two main segments of the domain – in the harvesting region and after exiting the region. Throughout its journey, as the vehicle aims to reach in a time-optimal manner, it will travel at its max speed. The path through the harvesting region is dictated principally by the specified final harvest state value γ_c. In particular, the total length of the path through this region must be given by

$$\begin{aligned} c_{Harvest\ Region} \cdot (L_{Harvest\ Region}/F_{max}) &= \gamma_c, \\ L_{Harvest\ Region} &= \frac{\gamma_c \cdot F_{max}}{c_{Harvest\ Region}}. \end{aligned} \tag{6.3}$$

As we will see, this will be a straight line path in the harvest region. After exiting the harvesting region at $x = 0.5$, the vehicle no longer can harvest any material and so it proceeds to take a time-optimal path to the destination. This will be given by a straight line path from the point of exit to the destination. Thus, the final optimal path is composed of two straight line paths in each region. To determine this optimal path, denote the y-coordinate for the point of exit from the harvesting region as y_m. If the start point on the reachability front is given by Equation (6.2), we have that the total time to reach the destination is given as

$$t_f(\alpha, \beta, y_m) = \frac{\left\|[0.5, y_m]^T - [x_i(\alpha, \beta), y_i(\alpha, \beta)]^T\right\| + \left\|[x_f, y_f] - [0.5, y_m]^T\right\|}{F_{max}}. \tag{6.4}$$

The harvested state at the destination is further given as

$$c_f(\alpha, \beta, y_m) = \frac{\left\|[0.5, y_m]^T - [x_i(\alpha, \beta), y_i(\alpha, \beta)]^T\right\|}{F_{max}} \cdot c_{Harvest\ Region}. \tag{6.5}$$

Given Equations (6.4) and (6.5), the final optimization problem to determine the

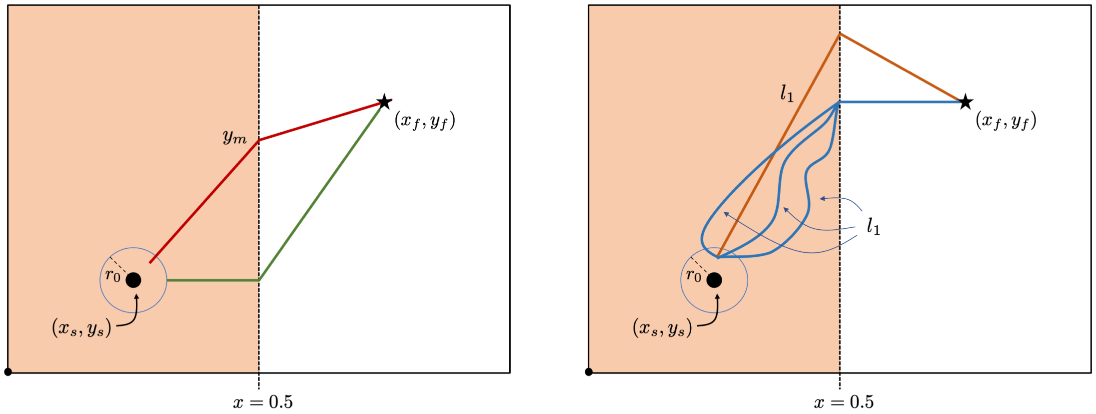


Figure 6-2: (Left) Candidate optimal paths for the halfspace region. Shown in red is a representative optimal path of the vehicle through the space. Shown in green is the form of the path that will yield the minimum possible final harvest state c_f^* in the set of Pareto optimal solutions. (Right) The non-uniqueness of solutions for higher values of final harvest states. After a solution from the class of "straight line" paths (from Equation (6.6)) returns a $y_{m_opt} > y_f$ (such as the orange path shown), the time optimal path is no longer unique. The true optimal path will belong to the class of solutions that all exit at $y_{m,opt} = y_f$, but which may not have straight line paths in the harvesting region. Shown in blue in the sub-figure are just a few of these non-unique optimal solutions for curved paths where the vehicle travels still at its max speed of F_{max}.

optimal path is as follows:

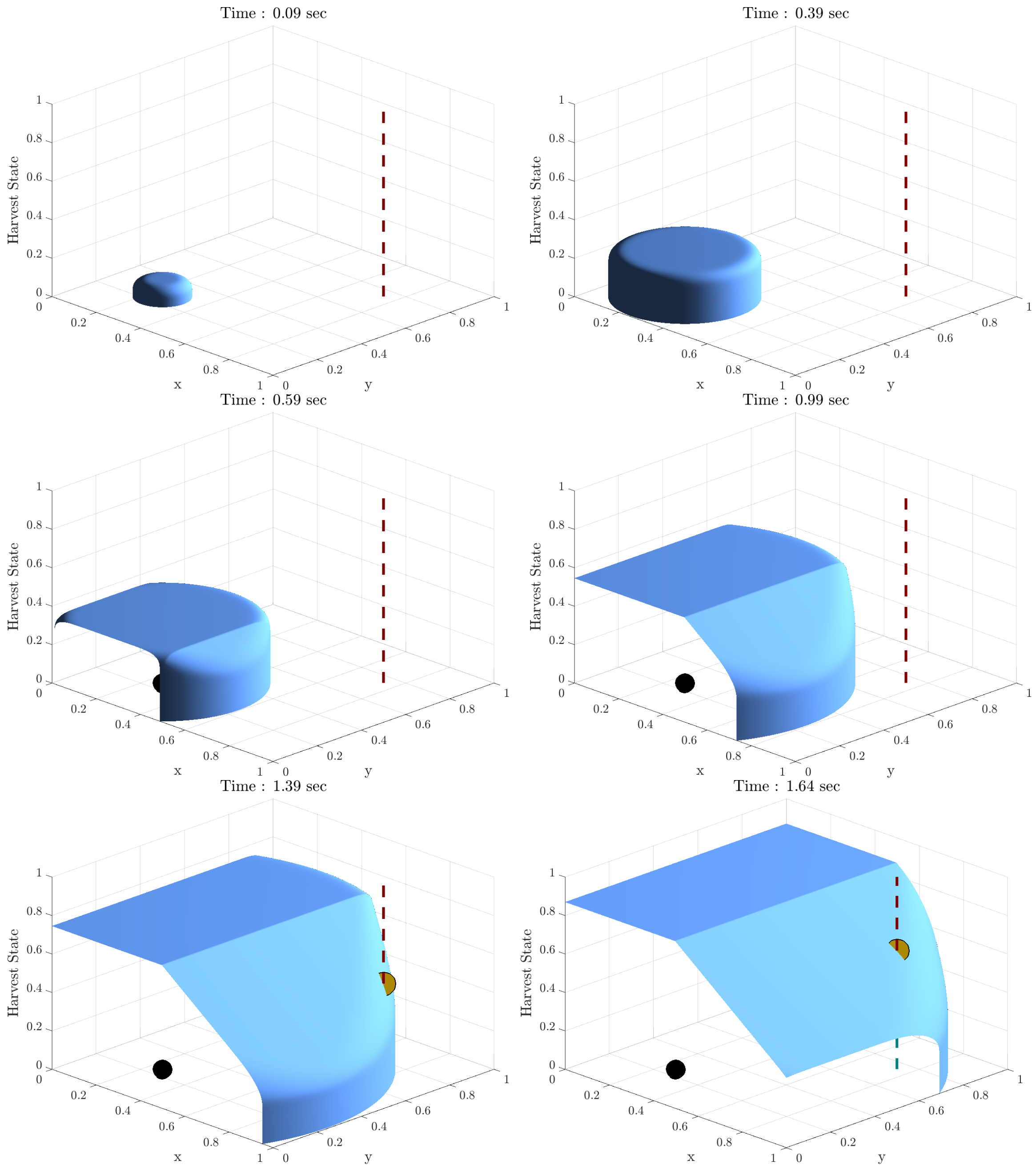
$$\begin{aligned} \underset{\alpha,\, \beta,\, y_m}{\text{minimize}} \quad & t_f(\alpha, \beta, y_m) \\ \text{subject to} \quad & c_f(\alpha, \beta, y_m) = \gamma_c \end{aligned} . \tag{6.6}$$

As outlined earlier, the full Pareto front for the multi-objective problem may be constructed by solving the optimization problem given by Equation (6.6) for different values of γ_c.

A few subtleties must be addressed about the semi-analytical solution. Firstly, the minimum possible final harvested state c_f is given when $y_m = y_s$, $\alpha = 0$, $\beta = \frac{\pi}{2}$. This corresponds to the minimum possible final harvest state c_f^* in the set of Pareto optimal solutions. Secondly, some attention must be paid to the case when the output to the optimization problem in Equation (6.6) returns a solution $y_{m,opt} > y_f$. In this case, the optimal solution would seem to indicate that the vehicle would need to exit above the line given by $y = y_f$ in order to harvest the requisite amount to attain the specified final harvest state of γ_c. In fact, the optimal solution is not in the class of "straight path" solutions, and the optimal path is not unique for this case. Let the length of the first leg of the journey (in the harvesting region) of the numerically computed path with $y_{m,opt} > y_f$ (and start point $(x_i(\alpha_{opt}, \beta_{opt}), y_i(\alpha_{opt}, \beta_{opt}))$) be given by l_1. Then, the class of optimal solutions contains the set of all paths, originating from $(x_i(\alpha_{opt}, \beta_{opt}), y_i(\alpha_{opt}, \beta_{opt}))$, that exit perpendicularly to the halfspace region but have a length of l_1 in the harvesting region (the path in the harvesting region need not be straight). The set of optimal paths further contains all other paths which originate from the within the reachable set that allow the vehicle to attain a harvest state value of γ_c when exiting at $(x, y) = (0.5, y_f)$, that is, perpendicularly to the halfspace region (these paths may not require the vehicle to travel at a constant maximum speed of F_{max}). Thus, the true minimum time to reach, for cases when the optimization problem given by Equation (6.6) is solved to yield $y_{m,opt} > y_f$, is determined as follows. First, one uses the returned incorrect "straight path" solution to find the required time to be spent in the harvest region. The travel time for this leg of the journey is then added to the travel time of the straight line path outside of the harvesting region for the vehicle traveling from $(x, y) = (0.5, y_f)$ to $(x, y) = (x_f, y_f)$ at its max speed F_{max}. This is the true minimum time to reach with the required final harvest state value. These details are all outlined in the schematic shown in Figure 6-2.

Numerical solution To numerically solve this benchmark path planning problem, the augmented state space was first discretized using 200 nodes along all three coordinate axes (x, y and c). For the forward solve, a second order ENO scheme was used to compute all spatial derivatives and a second order TVD Runge-Kutta scheme was used to integrate in time using a fixed time step size of $\Delta t = 5 \times 10^{-4}$. An initial reachability front radius of $r_0 = 0.05$ was used. Furthermore, a first order explicit scheme was used to backtrack the optimal paths corresponding to different Pareto optimal solutions chosen.

Figures 6-3 and 6-4 show the time evolution of the reachability front from the


Time : 0.09 sec

Time : 0.39 sec

Time : 0.59 sec

Time : 0.99 sec

Time : 1.39 sec

Time : 1.64 sec

Figure 6-3: Snapshots of the reachability front evolution in the augmented state state space for the benchmark halfspace case.

forward solve. Figure 6-3 shows a 3D isometric view of the front in the augmented state space, while Figure 6-4 depicts the contours of this front (along the harvest axis) when projected onto the physical space. The front can be seen to grow along the harvesting dimension in the harvest region as expected. This is representative of the vehicle being capable of harvesting material in this region. Upon leaving the

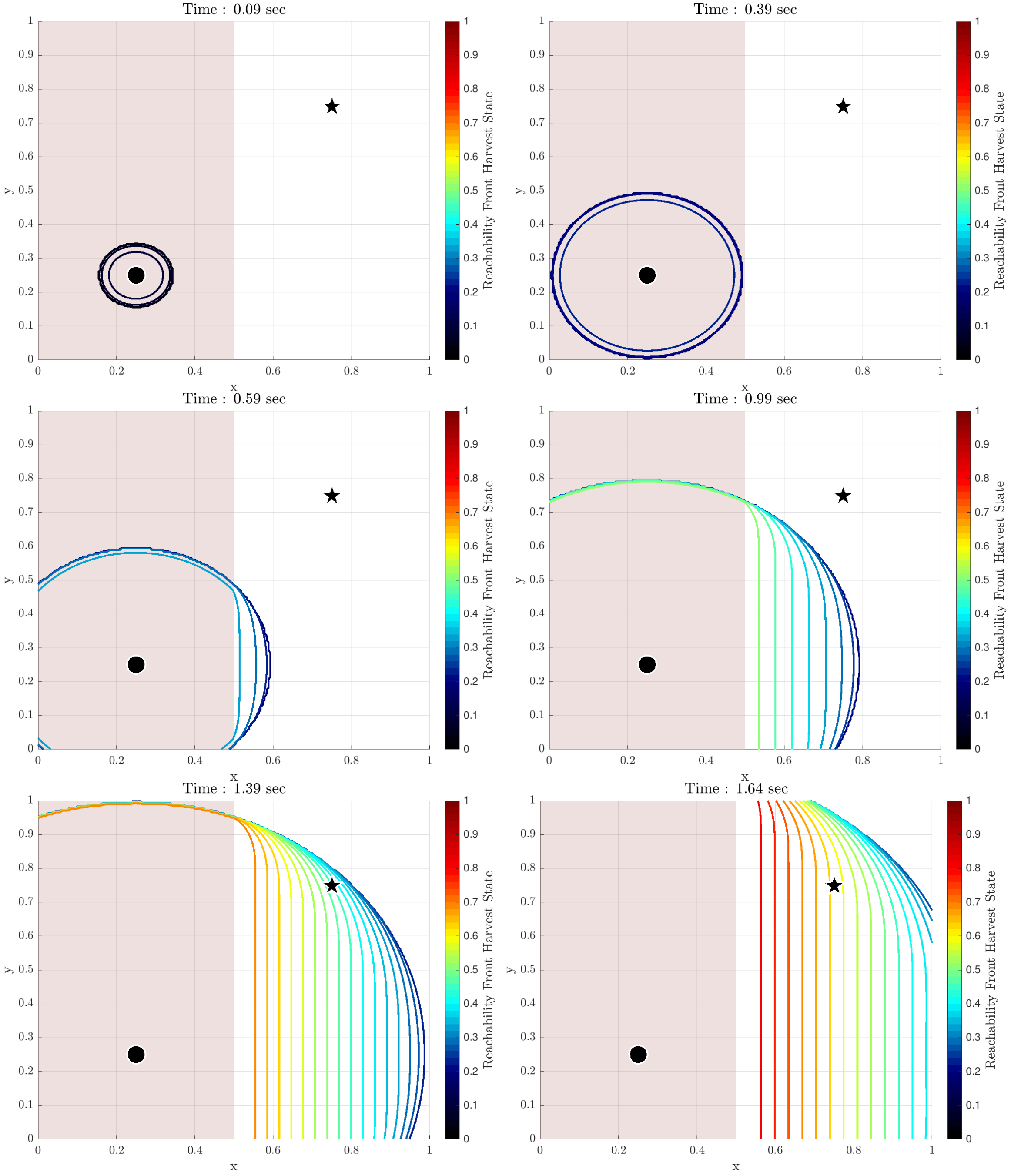
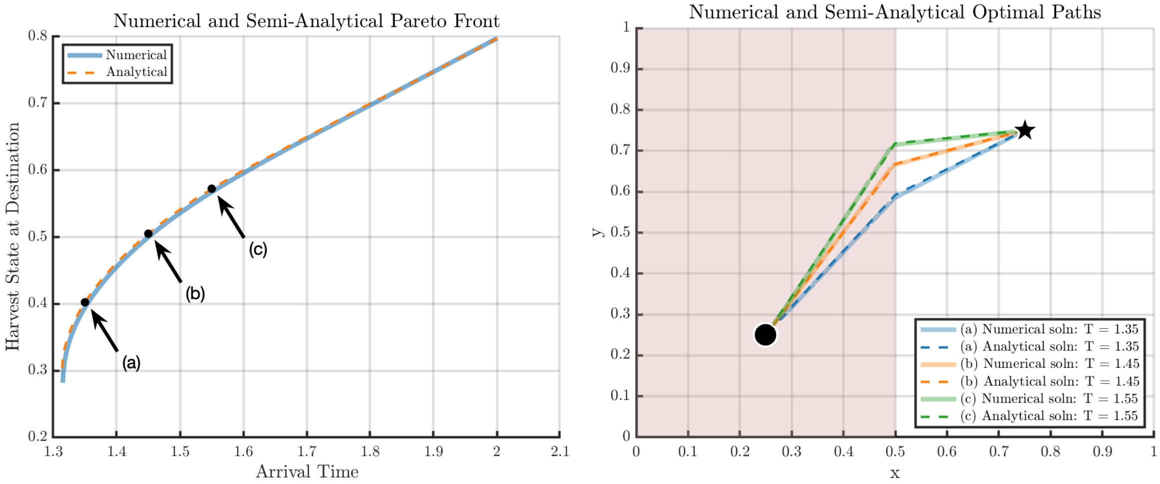


Figure 6-4: Snapshots of the reachability front evolution for the benchmark halfspace case, visualized using projections of its contours onto the physical space. Specifically, shown are projections of contours of the reachability front at different harvest state values.

halfspace harvest region, a "waterfall" type feature develops as the front enters the region where nothing is to be harvested.

Figure 6-5a depicts the Pareto front for this case. As can be seen, the numeri-



(a) The Pareto front.

(b) Pareto optimal paths.

Figure 6-5: Numerical and semi-analytical results for the benchmark halfspace case. These can be seen to be almost identical for both the computed Pareto front as well the different backtracked optimal paths (for three different solutions from the Pareto front).

cally computed front, determined using our methodology outlined in Chapter 4, is effectively identical to the semi-analytical solution, which was determined by solving the optimization problem given by Equation (6.6) numerically using SciPy (Virtanen et al., 2020). Three different Pareto optimal solutions from the Pareto front were further selected and the corresponding optimal paths were backtracked. These solutions were chosen for cases where the optimal path was unique (i.e. when $y_{m,opt} \leq y_f$). The resulting optimal paths, shown in Figure 6-5b, can be seen to be nearly identical to those obtained semi-analytically.

6.2 Energy-Time Optimal Path Planning: Double-Gyre Ocean Circulation

In this case we consider energy-time optimal path planning in a highly dynamic flow field. The time-dependent environment of interest consists of an idealized double-gyre flow. This flow simulates near-surface ocean circulation at mid-latitude regions, where easterlies and trade winds in the northern hemisphere drive a cyclonic and an anticyclonic gyre with the zonal jet in between, e.g. an idealized version of the Gulf Stream (Lolla et al., 2014b). The fluid flow field is furthermore governed by the non-dimensional PDEs

$$
\begin{aligned}
\frac{\partial u}{\partial t} &= \frac{\partial p}{\partial x} + \frac{1}{\mathrm{Re}}\Delta u - \frac{\partial (u^2)}{\partial x} - \frac{\partial (uv)}{\partial y} + fv + a\tau_x, \\
\frac{\partial v}{\partial t} &= \frac{\partial p}{\partial y} + \frac{1}{\mathrm{Re}}\Delta v - \frac{\partial (uv)}{\partial x} - \frac{\partial (v^2)}{\partial y} - fu + a\tau_y, \\
0 &= \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y},
\end{aligned}
\tag{6.7}
$$

which are solved numerically using our modular finite volume framework (Ueckermann and Lermusiaux, 2012). For the case shown, a flow Reynolds number of 150 was used with $f = \tilde{f} + \beta y$, the non-dimensional Coriolis coefficient, and $a = 10^3$, the strength of the wind stress. In non-dimensional terms, we use $\tilde{f} = 0$, $\beta = 10^3$. The flow in the basin is forced by an idealized steady zonal wind stress, $\tau_x = -\frac{1}{2\pi} cos\ 2\pi y$ and $\tau_y = 0$. Snapshots of the flow field at three non-dimensional times are shown in Figure 6-6. In addition, the environment is assumed to consist of a non-dimensional, steady, input energy flux field, $\dot{Q}(\boldsymbol{x}, t)$, shown in Figure 6-7.

The autonomous surface vehicle was tasked with navigating from the start point shown at $(x_s, y_s) = (0.2, 0.2)$ to a fixed target point $(x_f, y_f) = (0.8, 0.8)$. We furthermore considered in this problem a single additional harvesting state: the vehicle's energy. Specifically, the vehicle was equipped with energy harvesting capabilities with its energy dynamics governed by Equation (3.3). A cubic energy usage model of the form $\dot{W}(F) = 0.1 \cdot F^3$ was used for Equation (3.3). The vehicle could additionally

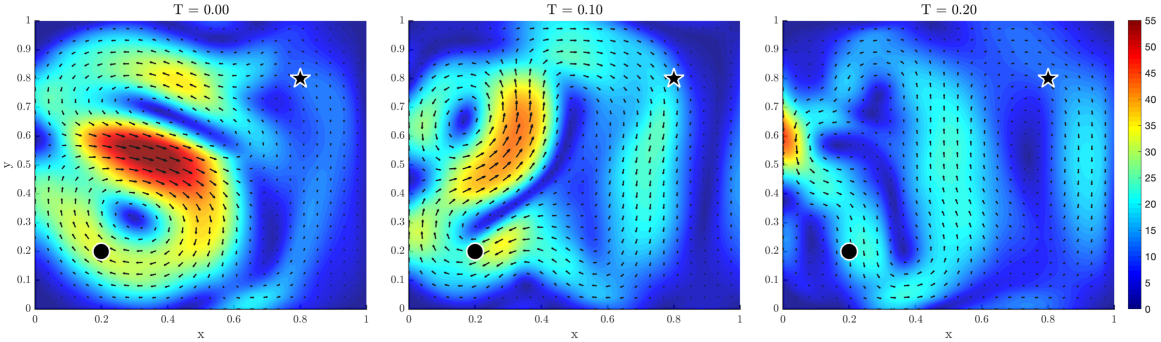


Figure 6-6: Snapshots of the flow field for the double-gyre case at three different non-dimensional times. Shown are the start point (circle) and end point (star) as well as the non-dimensional velocity magnitude plotted in the background.

control its heading $\hat{\boldsymbol{h}}$ and non-dimensional speed F, with the max possible value of the speed set at $F_{max} = 4$. The path planning problem then, we reiterate, was for the vehicle to simultaneously minimize the travel time and maximize the energy it had remaining upon reaching the destination, see Equation (3.5). This problem could be representative of a long-endurance autonomous surface vehicle, equipped with solar harvesting capabilities, navigating in a highly dynamic ocean environment with a solar flux input field (Figure 6-7).

To numerically solve the path planning problem, the augmented (x, y, e) state space was first discretized using 75 nodes along all three coordinate axes. To evolve the reachability front in the forward solve, a second order ENO scheme was used to compute all spatial derivatives and a second order TVD Runge-Kutta scheme was used to integrate in time using a fixed time step size of $\Delta t = 5 \times 10^{-5}$. Furthermore, as in the first case, a first order explicit scheme was used to backtrack the optimal paths corresponding to different Pareto optimal solutions.

Depicted in Figure 6-8 is the time evolution of the reachability front in the augmented (x, y, e) state-space. To visualize the progression of the 3D reachability front, shown in the figure are the contours of the front at various energy values projected onto the physical domain. It can be noted that the strong flow field has a significant effect on the propagation of the reachability front, as evidenced by its motion at earlier times where it can be seen to flow in a circular pattern as a result of the gyre (refer to the snapshots from non-dimensional times 0.01 to 0.05). The front furthermore

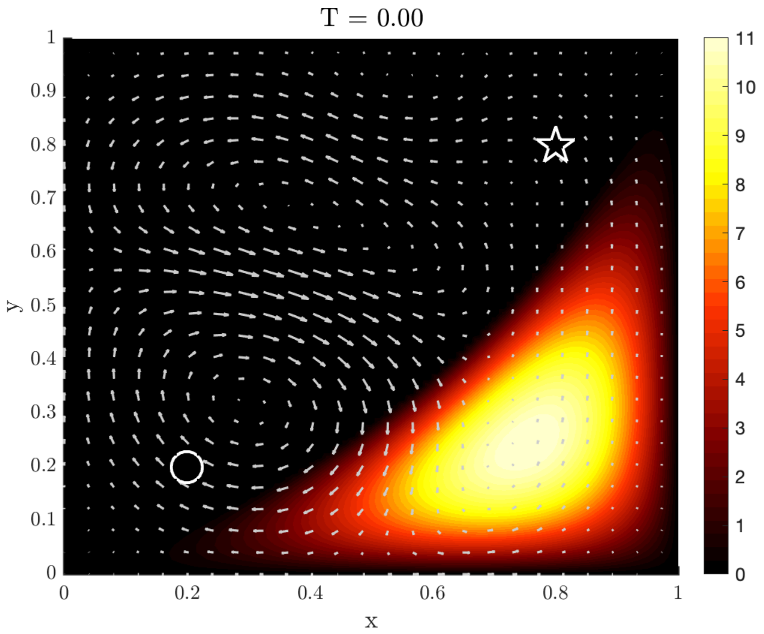


Figure 6-7: The velocity vector field at the initial time $t = 0$ with the non-dimensional, steady, input energy flux field $\dot{Q}(\boldsymbol{x}, t)$ shown in the background.

can be seen to expand fastest in directions in which it is assisted by the flow field (refer to the snapshot at non-dimensional time 0.08). When the reachability front enters the high energy input region at the bottom right corner of the domain (refer to the snapshots from non-dimensional times 0.12 to 0.20), the front expands along the energy dimension, which is representative of the vehicle being able to absorb energy.

Figure 6-9a displays the set of Pareto optimal solutions of the form (t_f^*, e_f^*) for this multi-objective energy-time path planning problem. The dotted red curve in the figure was first obtained by continuously measuring the maximum energy value of the reachability front at the destination as a function of time. This information could then be processed using the method outlined in Section 4.4.3 to compute the Pareto front. From this Pareto front, three different Pareto optimal solutions with increasing energy at destination values were considered, with the corresponding optimal path and optimal vehicle control sequence obtained by solving the backtracking Equation (4.81). Figure 6-9b shows the optimal path through the domain obtained for these chosen Pareto optimal solutions. Furthermore, the optimal speed function for each path is shown in Figure 6-9c, while the time evolution of the energy state of the vehicle

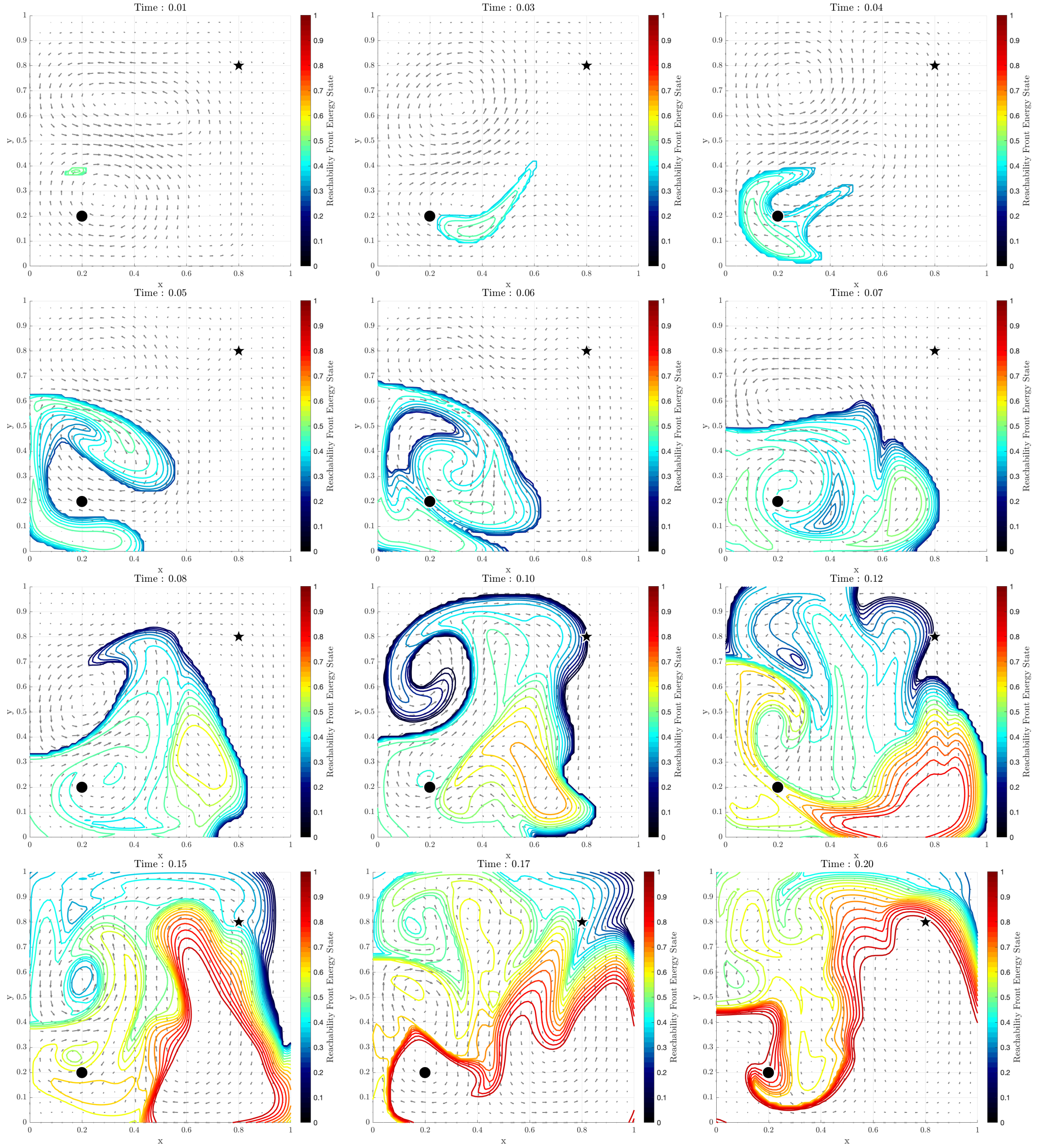
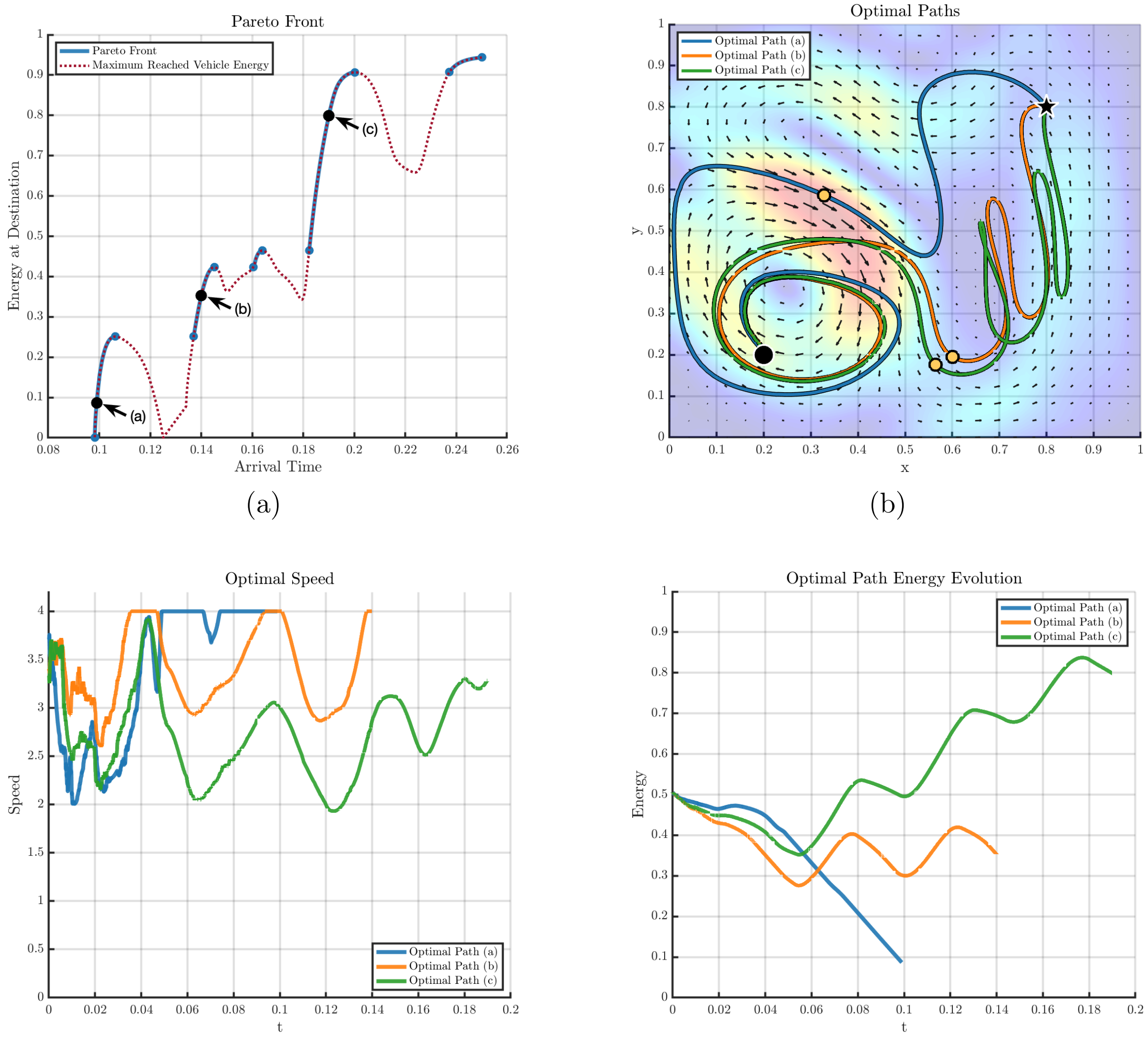


Figure 6-8: Snapshots at different times depicting the reachability front evolution in the augmented (x, y, e) state space for the double-gyre case. Shown is the projection of the reachability front at different energy values.

for each path is shown in Figure 6-9d. Each solution can be seen to have optimal paths that are quite similar near the start point, as they all perform a loop due to the influence of the gyre. Following this, the optimal solution with the lower final energy – optimal path (a) – uses a higher vehicle speed which allows for a more direct



(a) (b)

(c) (d)

Figure 6-9: Numerical results obtained for the multi-objective energy-time path planning problem. Shown in (6-9a) is the Pareto front. Additionally, for three different solutions from this Pareto front, we show in (6-9b) the corresponding optimal path through the physical space (velocity magnitude is shown in the background at the time each vehicle is at the the yellow points shown along their respective paths), in (6-9c) the optimal vehicle speed function $F^*(t)$ and in (6-9d) the energy evolution of the vehicle for the given optimal path.

path to the target. For the other two solutions, their higher energy at destination values require the vehicles to use a lower speed, in order to conserve energy, as well as navigate into the high energy input region in order to harvest energy from the environment. Specifically, optimal path (b) can be seen to make two trips into the energy input region, while optimal path (c) takes three. We emphasize, furthermore, that these computed optimal paths are globally optimal – for each (t_f^*, e_f^*) pair, no other path exists that allows the vehicle to reach the destination faster while reaching with the same energy.

6.3 Optimal Harmful Algae Bloom Collection

The material that follows in this section has been adapted from (Bhabra et al., 2020).

We now present cases to demonstrate how our methodology can be applied to a highly relevant issue: the control of harmful algae. Harmful algae blooms (HABs) occur due to a range of reasons including rising temperatures and nutrient-rich wastewater runoff (Schmale et al., 2019), leading to large colonies of micro or macro algae growing at rapid rates. HABs have been observed in all 50 states in the U.S., in large freshwater lakes, small inland lakes, rivers, reservoirs, and marine coastal areas and estuaries. These HABs can create low-oxygen dead zones (Griffith and Gobler, 2020; Joyce S, 2000), release toxins that are harmful to humans and animals, displace indigenous species, alter habitats and ecosystems (Anderson, 2009; Flynn and McGillicuddy, 2018), and lead to millions of dollars of economic losses each year through public health effects, commercial fishery impacts, medical expenses, and lost work days (Stauffer et al., 2019). Our optimal path planning methodology can help in not only monitoring, but also controlling HABs by actively collecting/harvesting these fields.

The dynamics of HAB growth is not fully understood. To enhance incomplete models, common forecasts utilize satellite and in-situ data (Stauffer et al., 2019). Traditional in-situ sampling methods are reactive, slow to respond, and operate with fixed time and spatial sampling grids, leading to calls to use AUVs for improved data collection (Robbins et al., 2006; Schofield et al., 2007; Denniston et al., 2019). These AUVs would greatly benefit from our optimal theory.

In addition to modeling and monitoring, many control strategies for HABs have been proposed. These include flocculation, sediment resuspension, burial, harvesting, water column mixing, biological control, and genetic engineering as well as the targeted release of allelochemicals, biosurfactants, hydrogen peroxide, copper sulfate, and silica (Kidwell, 2015; Anderson, 2009; Pal et al., 2020). Once areas where HABs are likely or have begun growing are known, our optimal path planning would deliver control strategies in an efficient and cost-effective manner to at-risk areas and so reduce ecological and economic damage.

In the following, we showcase our optimal harvesting as a control strategy to tackle a simplified simulated form of HABs, using autonomous surface vehicles to collect such dynamic algae fields time-optimally. In all cases, the algae growth and decay is modeled along with its advection due to the environmental flow field. We start by outlining some details regarding the methodology for this application, including relevant information about the ocean dynamics model used as well as the algae dynamics model. We then showcase the results obtained when applying this approach to two cases: (i) collection of algae in an idealized double-gyre flow and (ii) collection of fields biologically modeled in the Massachusetts (Mass.) Bay. This work and results have also been published within the joint manuscript (Bhabra et al., 2020), as part of a collaborative MSEAS group contribution.

6.3.1 Case Methodology Details

Dynamic Ocean Environment Modeling

Idealized Double-Gyre Simulation:

For the idealized flow case considered, the same double-gyre flow field from Section 6.2 was used. We refer the reader to this section for additional details.

Massachusetts Bay Simulation:

For the realistic simulations of Mass. Bay off the northeast US coast used in our second case, we employ our relocatable modeling system. Specifically, our group utilized our MIT-MSEAS modeling system (Haley and Lermusiaux, 2010; MSEAS Group, 2013; Lermusiaux et al., 2011; Haley et al., 2015), including our hydrostatic PE code with a nonlinear free surface, based on second-order structured finite volumes, and a generalized biogeochemical model for lower trophic levels (Lermusiaux et al., 2004). This MSEAS software is used for fundamental research and for realistic simulations and predictions in varied regions of the world's ocean (Leslie et al., 2008; Onken et al., 2008; Haley et al., 2009; Gangopadhyay et al., 2011; Ramp et al., 2011; Colin et al., 2013; Lermusiaux et al., 2017a; Subramani et al., 2017; Kulkarni et al., 2018; Gupta et al., 2019; Lermusiaux et al., 2019), including monitoring (Lermusiaux et al., 2007), ecosystem prediction and environmental management (Beşiktepe et al., 2003; Cossarini et al., 2009), and multi-disciplinary predictions and data assimilation (Lermusiaux, 2002; Haley et al., 2020).

The Mass. Bay set-up (Haley et al., 2020) has a 333 m horizontal resolution and 100 vertical levels with optimized level depths (e.g., in deeper water, higher resolution near the surface or large vertical derivatives, while at coasts, evenly spaced to minimize vertical CFL restrictions). The bathymetry was obtained from the 3 arc second USGS Gulf of Maine digital elevation model (Twomey and Signell, 2013). The sub-tidal initial and boundary conditions were downscaled from 1/12-degree Hybrid Coordinate Ocean Model (HYCOM) analyses (Cummings and Smedstad, 2013), using our optimization for our higher resolution coastlines and bathymetry (Haley et al., 2015). Local corrections were made using feature models and synoptic CTDs of opportunity. Tidal forcing was computed from the high resolution TPXO8-Atlas from OSU (Egbert and Erofeeva, 2002, 2013), by reprocessing for our higher resolution bathymetry/coastline and quadratic bottom drag. The atmospheric forcing consisted of hourly analyses/forecasts of wind stresses, net heat flux, and surface freshwater flux from the 3 km North American Mesoscale Forecast System (NAM) (National Centers for Environmental Prediction (NCEP), 2019).

Algae Dynamics Modeling

The algae growth model we used to simulate the fields for optimal collection is a modified version of the macroalgae growth model of Ren et al. (2014). The general applicability of this model and its inclusion of a wide array of environmental parameters and internal algae biological mechanisms allowed us to explore different environmental growth conditions and simulation regimes. The Ren model provides algae concentrations as a function of three state variables: Carbon, Nitrogen, and Phosphorous. The concentration of algae is then directly proportional to that of the carbon. Algae growth is linearly correlated to carbon growth, which can only occur under certain temperature, light, fluid flow, and internal state conditions, as modeled by:

$$U_C = G_{max} * C * f(T) * f(L) * f(Q_N) * f(Q_P) * f(V)\,. \tag{6.8}$$

In Equation (6.8), U_C is the carbon uptake rate, or scaled algae mass uptake rate; G_{max} the maximum carbon uptake rate; C the carbon concentration; and $f(T)$, $f(L)$, $f(Q_N)$, $f(Q_P)$, and $f(V)$ the effects on growth rate from temperature, light, Nitrogen quota, Phosphorous quota, and fluid velocity, respectively. The temperature and light functions $f(T)$ and $f(I)$ are defined respectively as an Arrhenius relationship comparing actual temperature to a set of biological references and a Lambert-Beer law to describe the behavior of light-attenuating materials in the water. The Nitrogen and Phosphorous quota functions $f(Q_N)$ and $f(Q_P)$ model the effect of internal nutrient ratios on the growth rate of the algae. The velocity function $f(V)$ is a Monond-type equation that models the dependence of algae growth on the flow field surrounding it. Formulations for these intermediate functions are in Ren et al. (2014).

Nitrogen and phosphorous uptake rates, which dictate algae growth through $f(Q_N)$ and $f(Q_P)$ in Equation (6.8), are:

$$U_N = \frac{f(T) * C * (U_{NH} + U_{NO})}{1 + exp[(Q_N - Q_{Nmax})/Q_{Noff}]}, \tag{6.9}$$

$$U_P = \frac{f(T) * C * U_{PO}}{1 + exp[(Q_P - Q_{Pmax})/Q_{Poff}]}, \tag{6.10}$$

where U_N and U_P are the uptake rates of nitrogen and phosphorous, respectively; U_{NH}, U_{NO}, and U_{PO} the ammonium, nitrate, and phosphorus uptake rates defined by a type-II function response to concentrations in the surrounding water; and Q_N, Q_P, Q_{Nmax}, Q_{Pmax}, Q_{Noff}, and Q_{Poff} the nitrogen and phosphorous current quota values N/C and P/C, where N and P are the stored nitrogen and phosphorus values; nitrogen and phosphorus maximum quota values; and nitrogen and phosphorus storage switch transition width values, respectively.

These three uptake rate functions are used in a system of ODEs to calculate the algae growth over time:

$$\frac{dC}{dt} = U_C - \Phi_C, \tag{6.11}$$

$$\frac{dN}{dt} = U_N - \Phi_N, \tag{6.12}$$

$$\frac{dP}{dt} = U_P - \Phi_P. \tag{6.13}$$

Here, $\Phi_{C,N,P}$ represent the death rate of each state variable:

$$\Phi_C = (k_r * f(T) + \omega_i + \omega_s) * C, \tag{6.14}$$

$$\Phi_N = Q_N * \Phi_C - \epsilon_{up} * (U_{NO} + UNH), \tag{6.15}$$

$$\Phi_P = Q_P * \Phi_C - \epsilon_{up} * (U_P), \tag{6.16}$$

where k_r is the carbon respiration coefficient; ω_i the algae intrinsic morality (an exponential function of the difference between the environmental temperature and a trigger temperature for intrinsic morality); ω_s the oxygen stress morality (a Monond-type equation of the oxygen concentration below a threshold level); and ϵ_{up} the nutrient uptake associated excretion coefficient.

Environmental input fields to this point ODE system is taken from our MSEAS high resolution coupled physical-biogeochemical simulations. The fields not available from these simulations include dissolved oxygen levels and environmental phosphorous availability. Both of these terms are assumed here to be uniformly replete throughout the domain. This assumption eliminates the dissolved oxygen growth death term ω_s in the carbon death rate term and sets U_{PO} to be uniformly equal to its maximum value U_{POmax}. Certain algae parameters are slightly modified to better fit algae growth to the northeast climates, such as the Arrhenius temperature constants used in $f(T)$. Detritus and zooplankton fields are used as particulate inorganic and particulate organic light-attenuating materials for the Lambert-Beer light function $f(I)$. For increased accuracy, the Ren model was also modified to use our instantaneous local flux of solar intensity accounting for cloud cover, as opposed to only seasonal fluctuations. Finally, a Monond-type scaling factor (Bendoricchio et al., 1994) modifies the carbon intrinsic morality term ω_i for more accurate results in our simulations.

The ODE system given by Equations (6.11), (6.12) and (6.13) describes the evolution of algae at a given point. To further account for the dynamics of the algae field in ocean contexts (from advection) this ODE can be used with the material derivative of each nutrient field resulting in a set of advection-reaction PDEs. This is what has been solved numerically, using the QUICK scheme in our modular finite volume framework (Ueckermann and Lermusiaux, 2012), in the results that follow.

Optimal Algae Collection Path Planning

The ocean dynamics model and algae model can be first simulated to yield $\mathcal{A}(\boldsymbol{x}, t)$ – the dynamic field describing the evolution of the algae concentration in the environment. The harvesting dynamics of algae for the autonomous surface vehicle then follows from Chapter 3 for the model of collecting an arbitrary tracer field. That is, we can model the amount of algae collected by the vehicle at a given point to be proportional to the amount of algae present at that location. Specifically, consider the augmented state space given by (x, y, a), where a corresponds to the amount of algae harvested. If we considered a given path $\boldsymbol{X}_p(t)$ through the physical space, resulting in a time history $A_p(t)$ for the vehicle's amount of algae collected, we have that the $A_p(t)$ satisfies

$$\frac{\mathrm{d}A_p}{\mathrm{d}t} = \alpha \cdot \mathcal{A}(\boldsymbol{X}_p(t), t), \tag{6.17}$$

where $\alpha \in \mathbb{R}_{>0}$ is a constant of proportionality. Given a specified start and end point in the domain, a Pareto front can be found for the multi-objective optimization problem (outlined in Equation (3.5)) for the joint objectives of minimizing the time of arrival and maximizing the amount of algae harvested.

In the following, a simpler set of results are presented. In particular, we do not, as before, form the full Pareto front holding solutions of the form (t_f^*, a_f^*) – arrival time and amount of algae harvested upon reaching the destination. Rather, we present results for time optimal paths where the vehicle is constrained to collect a specified amount of algae, $a_{constraint}$, prior to reaching the destination. This ultimately corresponds to a given point in the set of Pareto optimal solutions (it is the point on the Pareto front with a final algae amount harvested upon reaching the destination value of $a_f^* = a_{constraint}$). Of course, if desired, the full Pareto front can be trivially computed as well using the methodology outlined in Chapter 4 as was done in the cases presented in Sections 6.1 and 6.2.

We additionally reiterate (following from the discussion in Chapter 3) that in the proceeding results one key assumption is that the vehicle has a negligible influence on the algae field it is collecting. This must be assumed as the dynamics of the vehicle path cannot be incorporated into the external field during the solve. Therefore, the effect of the reduction in algae from the surrounding field cannot be imposed until the final path is computed. Fortunately, the length scales of the algal fields are typically orders of magnitude larger than the widths of the paths traced by the autonomous vessels, and so this assumption is then not too restrictive. Similarly, it is also common for the duration of the vehicle's mission to be limited (e.g. power constraints) such that the time required by the vehicle to complete its harvesting is typically short compared to the time-scales of the field to be harvested. In such cases, once a mission is completed, the harvested quantities can be removed from the field to be harvested at the right times along the vehicle path. The next path planning mission can then use this reduced field to be harvested.

6.3.2 Results

Double-Gyre Flow Advecting a Dynamic Algae Field

In this first sub-case, we consider optimal algae collection in an idealized double-gyre flow field (Figure 6-10). We employ our modified algae growth model (Section 6.3.1). All relevant environmental parameters are assumed constant except for the three nutrient fields (nitrate, ammonium, phosphorus) and the light field. The spatially varying light field is lower intensity toward the left hand side of the domain ($x < 0.5$)

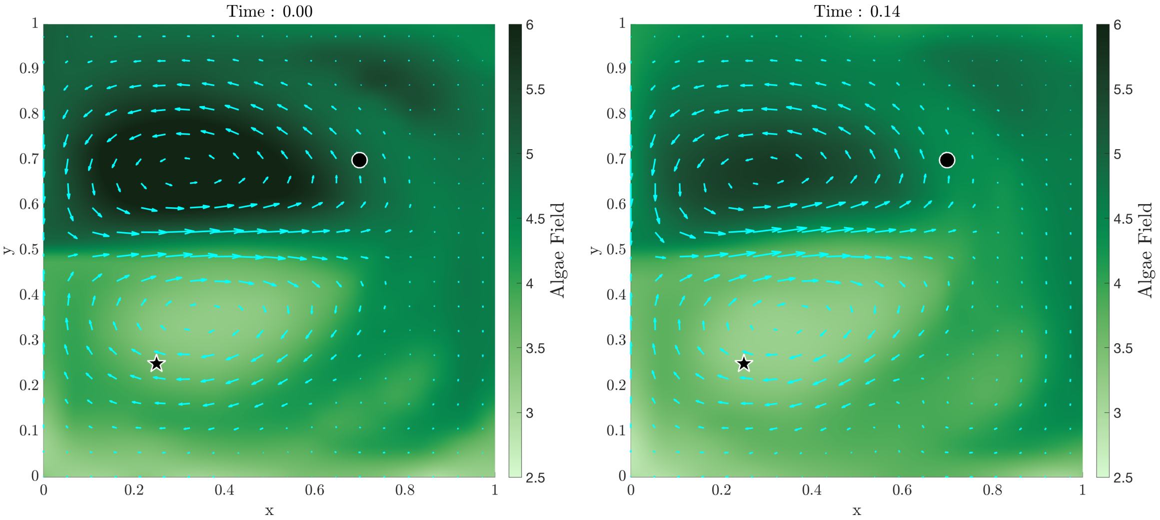


Figure 6-10: Snapshots of the environment at non-dimensional times 0 and 0.14 with the unsteady velocity fields overlaid on an algae collection rate field (in units of $\frac{\text{[units of algae]}}{\text{[time]}}$) – high values correspond to regions where the algae concentration is higher resulting in a higher collection rate. Start (full circle) and end (star) points of the vehicle are shown.

and higher intensity toward the right hand side of the domain ($x > 0.5$) to simulate cloud coverage impairing the growth of the initial high-concentration area (see upper left of the two snapshots in Figure 6-10). In addition, the initial values for the three nutrient fields are assigned high and low concentration areas throughout the domain to simulate spatial variability in the algae growth rate.

In this sub-case, we demonstrate an optimal collection with the constraint of having to collect at least 0.7 units of algae before reaching the destination. Figure 6-10 shows the dynamic environment at two times, with two gyres, the jet, and the algae field dynamically evolving and transported by the flow.

Figure 6-11 shows the evolution of the 3D reachability front. As required, these 2D contours (slices of the 3D front) evolve until the level set value of 0.7 units is reached at the target.

The final computed optimal path is in Figure 6-12. The path clearly evolves to remain in the areas of high concentration, while simultaneously using the external currents to its advantage in order to reach the target as fast as possible. It is important to reiterate that this computed path is *globally optimal*; no other path from the start

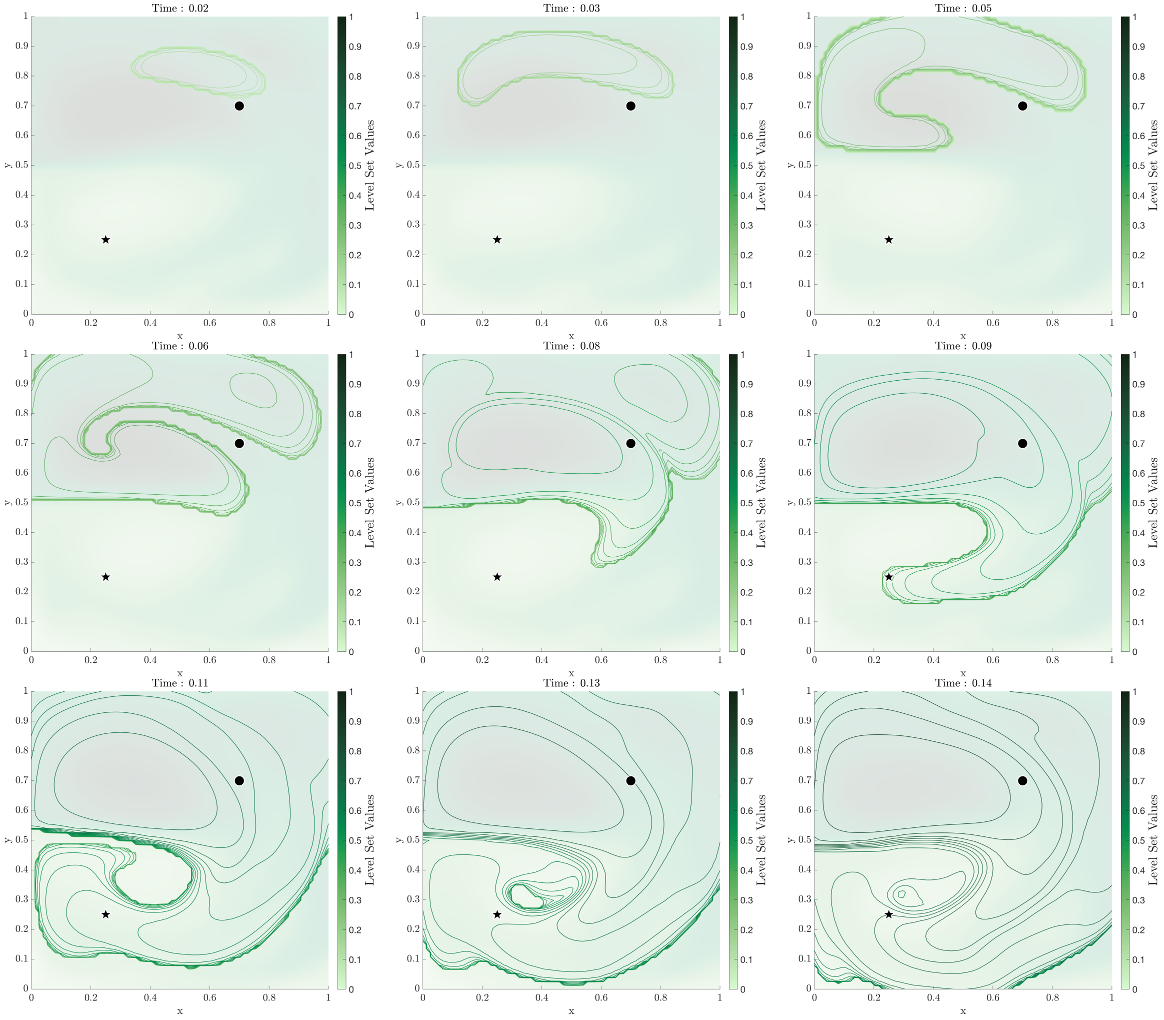


Figure 6-11: Snapshots of the level set at various points in time. Since the level set is a surface in an augmented space (2D physical space and an additional dimension for the collected algae amount), the constant algae collection amount contours of this surface are shown. The level set evolves until it attains a value of 0.7 units at the target.

to the end point exists in which the vehicle can reach faster while collecting 0.7 units of the field.

Massachusetts Bay - Harvesting Dynamic Chlorophyll Fields

For the second sub-case, we consider the Mass. Bay region. We showcase an alternate control strategy to monitor harmful algal blooms: the study of chlorophyll fields. Various studies have shown that the concentration of chlorophyll (particularly chlorophyll-a) provides a direct measurement of algae growth in aquatic environments

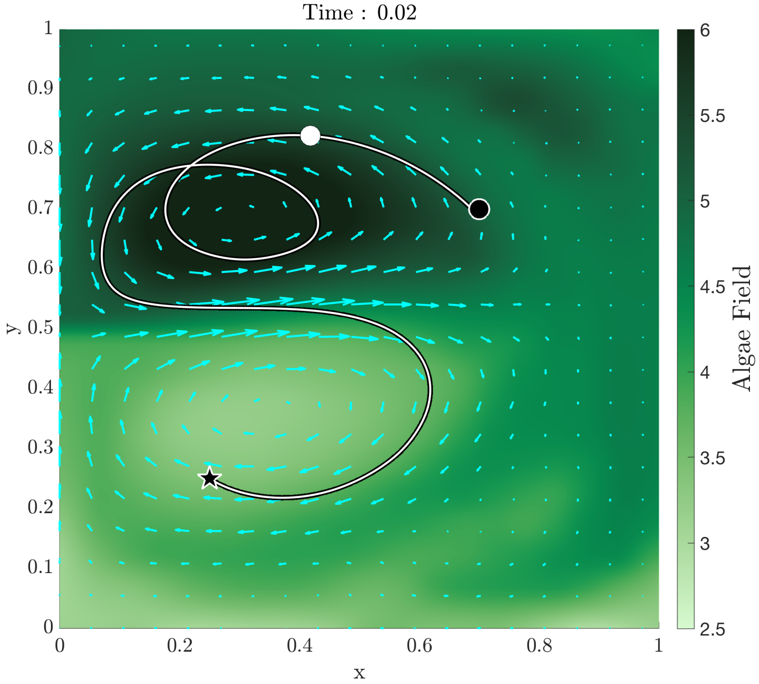


Figure 6-12: The final computed globally-optimal path for the algae collection problem with the vehicle (white circle) shown traversing it at some intermediate time. Drawn in the background is the algae collection rate field at an intermediary time (the time corresponding to when the vehicle is at the white point shown).

(Karki et al., 2018). This close connection between chlorophyll and algae has been exploited in several research avenues for predicting harmful algal blooms (Baban, 1996; Schofield et al., 2007; Stauffer et al., 2019) and here motivates our collection of dynamic chlorophyll fields.

Our high-resolution coupled physical-biogeochemical simulations are used to model the ocean currents (Figure 6-13) as well as the chlorophyll field.

Using our theory and schemes for optimal harvesting, we plan the path of a vehicle that travels from a start point to a destination while optimally collecting chlorophyll along the way. Figure 6-14 shows the evolution of the level sets from the forward solve. Here, as before, the reachability front evolves at almost a constant collected chlorophyll value until it reaches offshore from Scituate. In this region, chlorophyll has bloomed which results in a corresponding increase in the reachability front value (at 3:40 GMT for instance). The front then, as before, expands until it reaches the target point with the desired final chlorophyll field amount of 0.4 units.

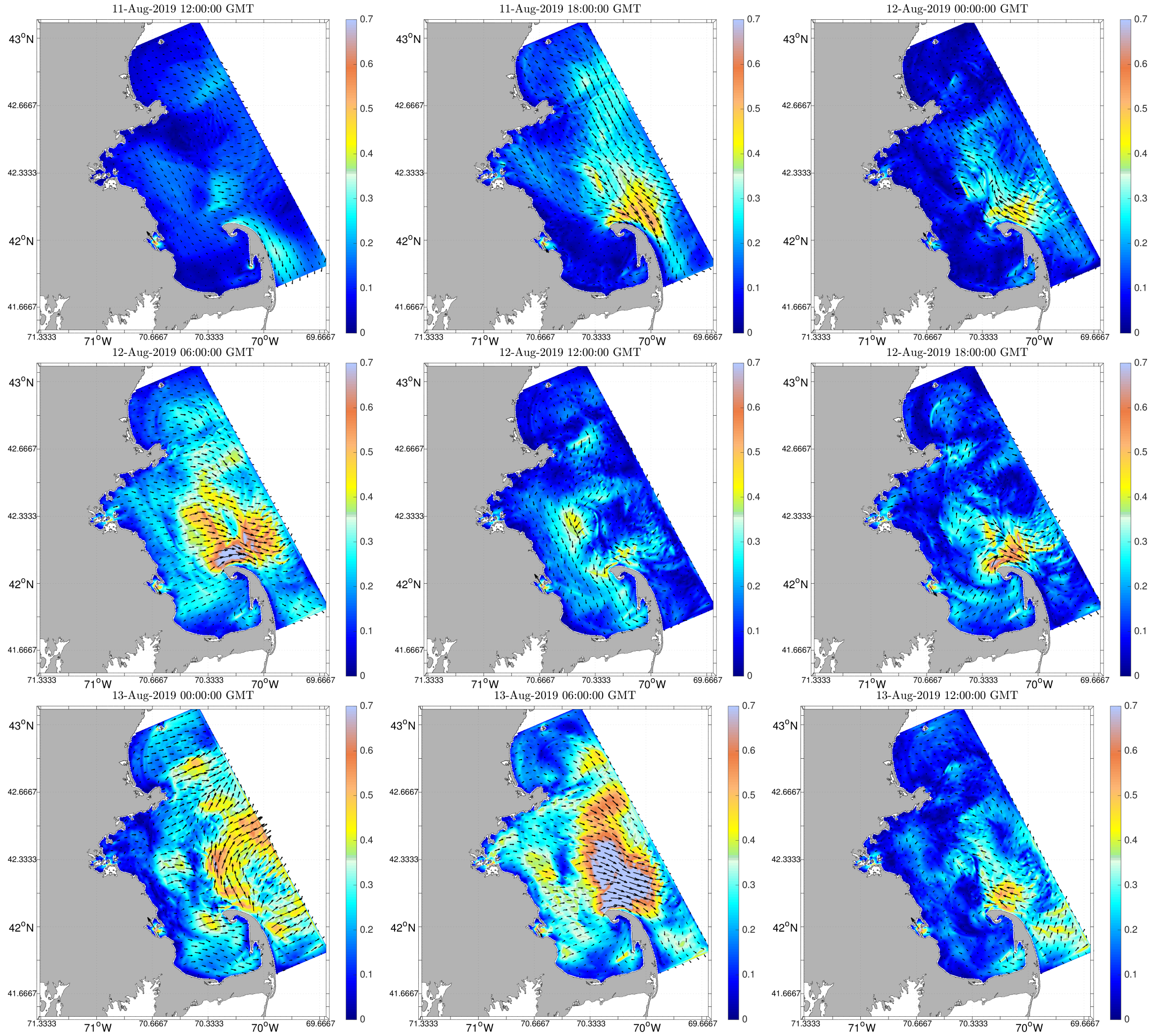


Figure 6-13: Dynamic velocity field in Mass. Bay simulated using our data-assimilative MSEAS-PE modeling system and plotted between 11 and 13 Aug. 2019. Colored in the background is the velocity magnitude.

Figure 6-15 shows the final optimal path. As expected, the autonomous vessel's optimal path takes it to high chlorophyll concentration regions offshore from Scituate to harvest just enough of the field before starting its return journey to the target. The vehicle reaches the end point with the desired chlorophyll amount of 0.4 units at 12 Z 12 Aug 2019.

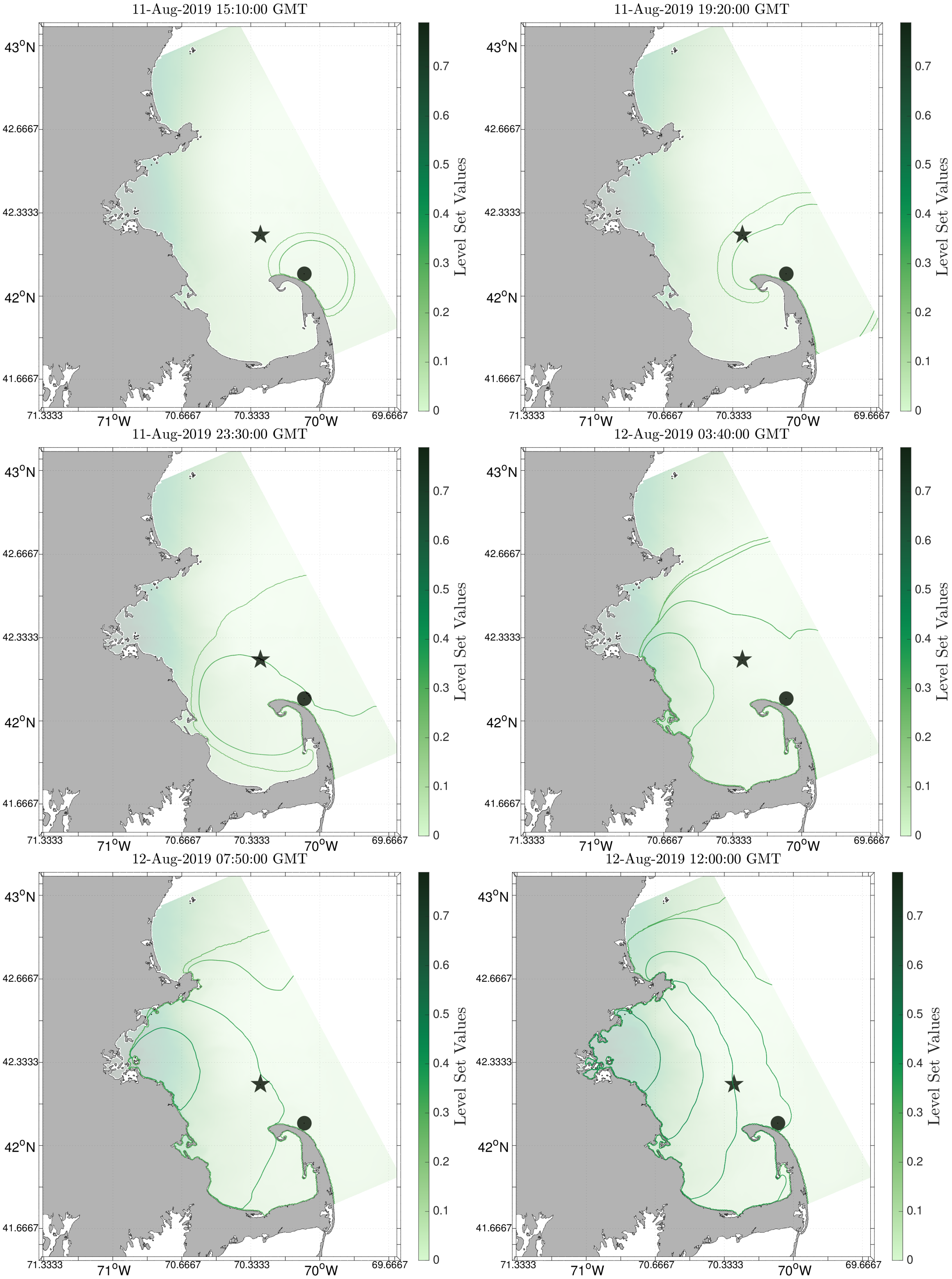


Figure 6-14: Evolution of the forward level set. The background represents the concentration of Chlorophyll from our MSEAS PE-bio simulation at that time. The contours shown represent the reachability front at various collection levels.

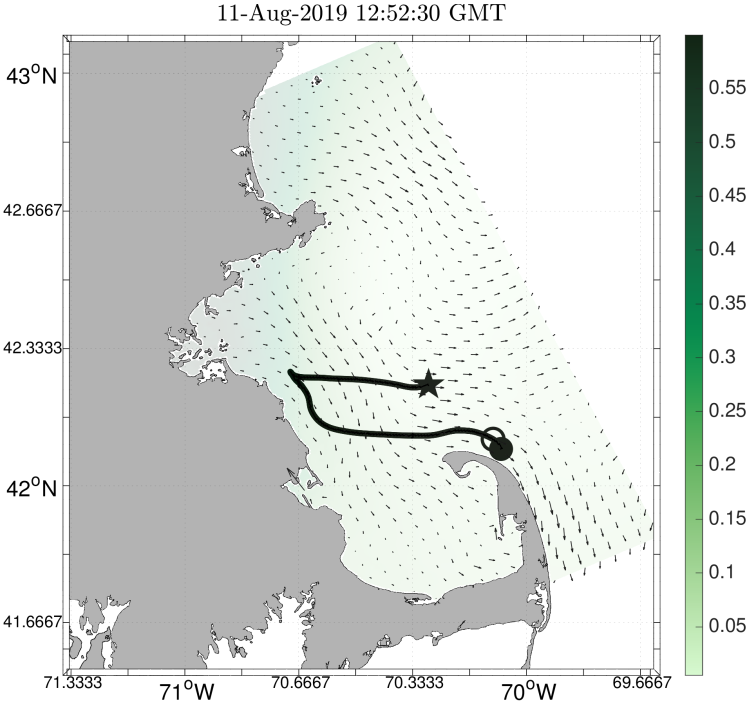


Figure 6-15: The final globally-optimal path from start (full circle) to finish (star) while collecting the required amount of Chlorophyll. A snapshot is shown at an intermediate time (vehicle shown as the circle). Shown in the background is the spatially varying Chlorophyll collection rate field (in units of $\frac{\text{[units of Chl]}}{\text{[time]}}$) at the time the vehicle is at the point shown in the circle along the optimal path – high values correspond to regions where the concentration of Chlorophyll is higher resulting in a higher collection rate.

6.4 Optimal Fish Farming

In this final case, we consider an application of our methodology to offshore aquaculture. As discussed in Chapter 2, recent years have seen active development in the research community regarding revolutionary new methods for offshore fish farming. One such next generation technology consists of large, mobile, fully autonomous cages for farming fish. These cages would be equipped with thrusters to allow it to control itself while navigating. The key idea would then be that, after being launched with lab-bred baby fish, these moving farms would cruise along ocean currents and ultimately arrive at a specified destination with mature animals ready for market (Mann, 2004; Cohen, 2008; Goudey, 1998).

Our novel approach to harvest-time optimal path planning is directly applicable to this application. Our presented methodology, in particular, can be used by these autonomous moving farms to determine what paths to take in order to reach destinations time-optimally while also maximizing the amount of fish growth in the farm itself, e.g., by navigating in regions rich in nutrients to be eaten.

Key to the analysis of these moving fish farms is a model representative of the growth dynamics of fish. Due to the complexities of marine ecosystems, various models have been proposed for different families of fish. For the results presented in this section, we follow the model used in Gupta et al. (2019). This model couples biological models for two main parts of the ocean ecosystem: Lower Trophic Levels (LTL) and Higher Trophic Levels (HTL). A three-component nutrients-phytoplankton-zooplankton (NPZ) model is used for the lower trophic levels. The NPZ model output is then coupled with the outputs of the physical model for the ocean dynamics – the nonhydrostatic Navier-Stokes equations with a variable-density Boussinesq approximation – to obtain a spatial field referred to as the habitat index, $H_{hab}(\boldsymbol{x}, t)$. This habitat index encodes the favorability of a location for the fish, which in the case of Gupta et al. (2019) was tuna. As a simple first design pass, we also consider the mass growth of fish in the farm to be directly proportional to the habitat index. Consider the augmented state space given by (x, y, m_{tuna}), where m_{tuna} corresponds to the mass of tuna in the fish farm. If we consider a farm path $\boldsymbol{X}_p(t)$ through the physical space and the farm's tuna mass evolution is denoted $M_{tuna,p}(t)$, we have that the evolution of tuna satisfies

$$\frac{\mathrm{d}M_{tuna,p}}{\mathrm{d}t} = \alpha \cdot H_{hab}(\boldsymbol{X}_p(t), t), \tag{6.18}$$

where $\alpha \in \mathbb{R}_{>0}$ is a constant of proportionality. We once again note that we do not model the effect of the fish farm on the habitat index field, i.e. a reduction in the habitat index values in regions where the fish farm has been due to the consumption of nutrients. This assumption should not result in large inaccuracies, as in most applications the length scale of the farm is orders of magnitude smaller than that of the habitat index field. Furthermore, strong wave mixing around the farm structure should allow for rapid nutrient replenishment.

For this case, we additionally consider a slightly different autonomous vehicle. While until now autonomous surface vehicles have been considered, autonomously moving fish farms will require the ability to be submersible. As such, we look at an autonomous underwater vehicle for this case. Furthermore, we consider a 2D domain, given by a vertical slice of the ocean (in what follows, x will denote the range position of the vehicle while y corresponds to the vertical distance from the seafloor). Most autonomous underwater vehicles, such as gliders, have different actuators for moving in range and depth (Griffiths, 2002). To account for this, we use the following simple model for the dynamics of our submersible moving fish farm in the vertical 2D domain

$$\frac{\mathrm{d}\boldsymbol{X}_p}{\mathrm{d}t} = \boldsymbol{V}(\boldsymbol{X}_p(t), t) + F_x(t) \cdot \boldsymbol{e}_x + F_y(t) \cdot \boldsymbol{e}_y, \tag{6.19}$$

where $\boldsymbol{V} = [V_x, V_y]$ is the external velocity field, $\boldsymbol{e}_x = [1, 0]^T$ and $\boldsymbol{e}_y = [0, 1]^T$. The different actuators used to navigate along each coordinate direction result in different relative speeds, which we denote by the constraints $|F_x| \leq F_{x,max}$ and $|F_y| \leq F_{y,max}$.

Given the harvesting dynamics in Equation (6.18) and vehicle dynamics in Equation (6.19), the Hamilton-Jacobi equation governing the evolution of the reachability front can be derived analogous to what was shown in Chapter 4. Substituting the dynamics into the Equation (4.63) we obtain:

$$\frac{\partial \phi}{\partial t} + \max_{F_x, F_y} \left\{ \frac{\partial \phi}{\partial \boldsymbol{x}}^T \cdot \Big(\boldsymbol{V}(\boldsymbol{x}, t) + F_x \cdot \boldsymbol{e}_x + F_y \cdot \boldsymbol{e}_y \Big) + \frac{\partial \phi}{\partial m_{tuna}} \cdot \Big(\alpha \cdot H_{hab}(\boldsymbol{x}, t) \Big) \right\} = 0, \tag{6.20}$$

where the time dependent value function is defined on the full state and is of the form $\phi = \phi(\boldsymbol{x}, m_{tuna}, t)$. Pulling out all terms that do not depend on the controls from the maximization, this simplifies to

$$\begin{aligned} \frac{\partial \phi}{\partial t} + \max_{F_x, F_y} \left\{ F_x \cdot \frac{\partial \phi}{\partial x} + F_y \cdot \frac{\partial \phi}{\partial y} \right\} + \frac{\partial \phi}{\partial \boldsymbol{x}}^T \cdot \boldsymbol{V}(\boldsymbol{x}, t) \\ + \frac{\partial \phi}{\partial m_{tuna}} \cdot \Big(\alpha \cdot H_{hab}(\boldsymbol{x}, t) \Big) = 0. \end{aligned} \tag{6.21}$$

The optimal speeds F_x and F_y along each coordinate direction are now determined based on the sign of the components of the spatial gradient. In the case where $\frac{\partial \phi}{\partial x} \le 0$, we have that $F_x^* = -F_{x,max}$, whereas when $\frac{\partial \phi}{\partial x} > 0$ we have $F_x^* = F_{x,max}$. This holds similarly for the speed F_y. Substituting the optimal speeds, we obtain the following

$$\begin{aligned} \frac{\partial \phi}{\partial t} + F_x \left| \frac{\partial \phi}{\partial x} \right| + F_y \cdot \left| \frac{\partial \phi}{\partial y} \right| + \frac{\partial \phi}{\partial \boldsymbol{x}}^T \cdot \boldsymbol{V}(\boldsymbol{x}, t) \\ + \frac{\partial \phi}{\partial m_{tuna}} \cdot \Big(\alpha \cdot H_{hab}(\boldsymbol{x}, t) \Big) = 0, \end{aligned} \tag{6.22}$$

which is the Hamilton-Jacobi equation governing the evolution of the reachability front in the augmented state space. Just as described in Chapter 4, the maximizing speeds determine an optimal closed-loop control law for the backwards dynamical system, which can be used to trace out the optimal path (for a chosen point on the Pareto front) by solving the backtracking ODE.

As in Gupta et al. (2019), in the following case we consider a 2D domain with a seamount representing an idealized sill or strait that can create an upwelling of the nutrients and thus phytoplankton blooms. The seamount also forces the advection of cold water upward which has a negative effect on the habitat index, thus limiting the favorable habitat for tuna to very specific depths and locations. In part

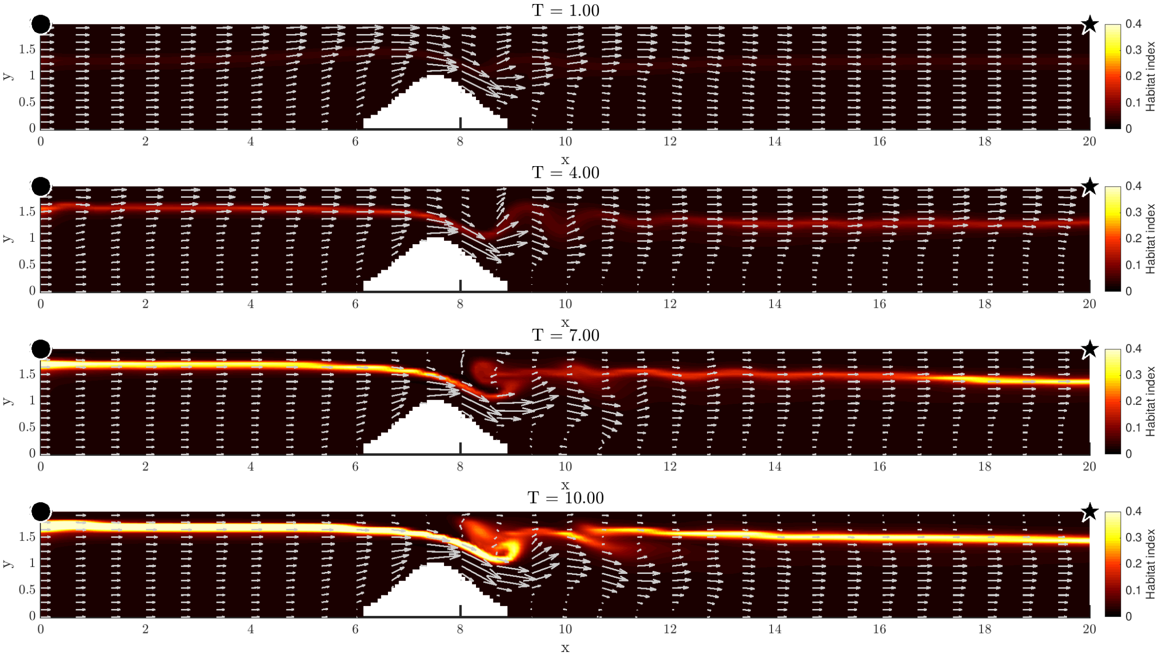


Figure 6-16: Snapshots of the environment at four different non-dimensional times. Shown is the velocity vector field along with the habitat index in the background. The habitat index can be seen to have high values in only a very narrow region.

due to the nonhydrostatic dynamics, we additionally have that internal waves (e.g., lee waves), recirculations, and other instabilities can be created downstream of the seamount, leading to additional biogeochemical-fish responses. This domain is inspired by Stellwagen Bank off of Massachusetts (Lermusiaux, 2001; Beşiktepe et al., 2003; Ueckermann and Lermusiaux, 2010). Here, flow occurs from left to right in the positive x-direction over the seamount. For velocity, we apply a Dirichlet boundary condition for the inlet, no-slip for the bump, free-slip at the top and bottom of the domain, and an open boundary condition at the outlet. For the tracer fields, we use zero-Neumann on all the boundaries. We additionally numerically solve for the habitat index using an in-house Finite Volume solver (Ueckermann and Lermusiaux, 2012). The evolution of the habitat index and the corresponding flow is given in Figure 6-16.

Here, the full Pareto front provides solutions of the form $(t_f^*, m_{tuna,f}^*)$ – arrival time and mass of tuna in the farm upon reaching the destination. As in the previous case, we do not showcase here the full Pareto front but rather search for a time-optimal

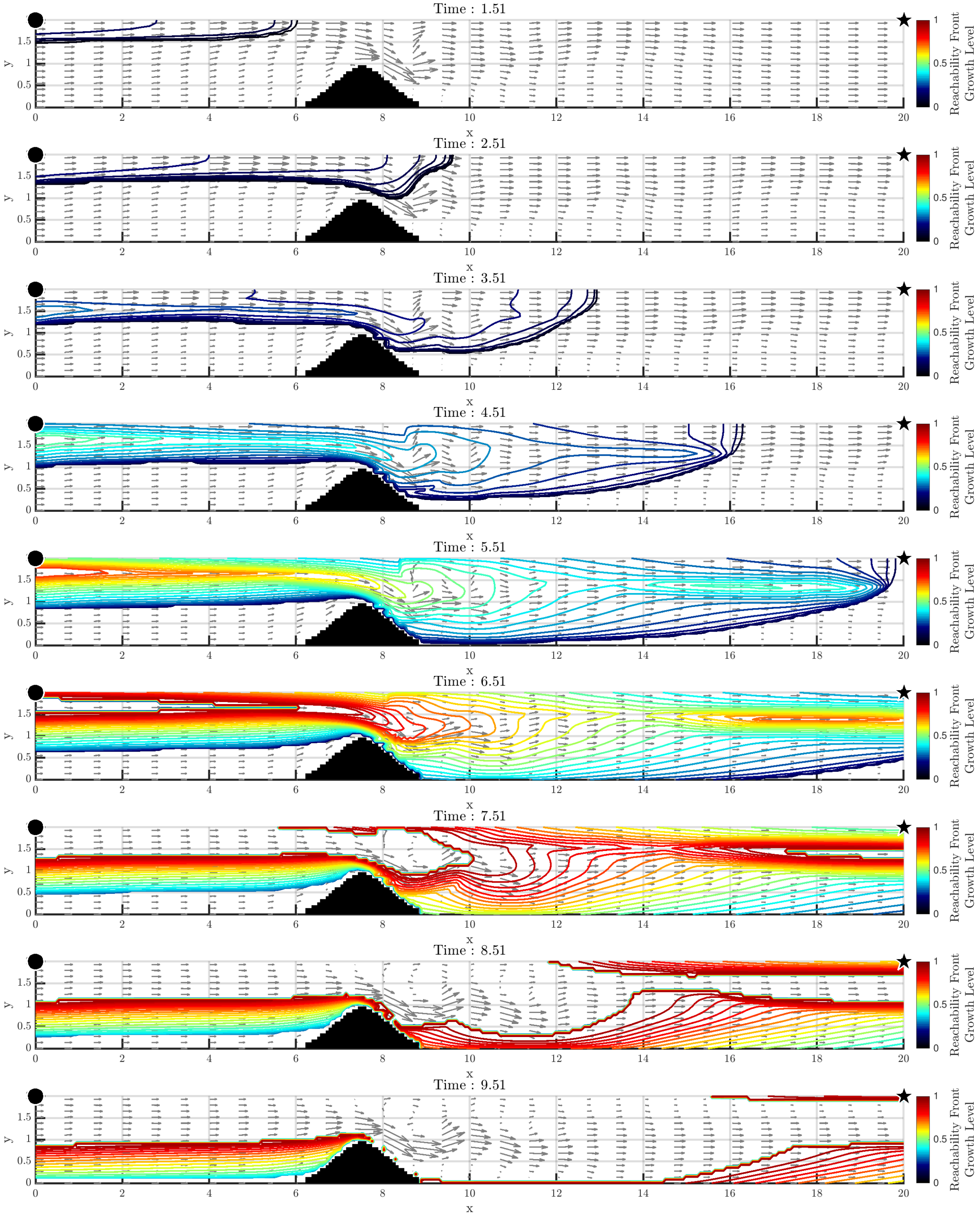


Figure 6-17: Snapshots of the reachability front evolution at various non-dimensional times in the augmented (x, y, m_{tuna}) state space. Shown are projections onto the physical domain of contours of the reachability front at constant m_{tuna} values.

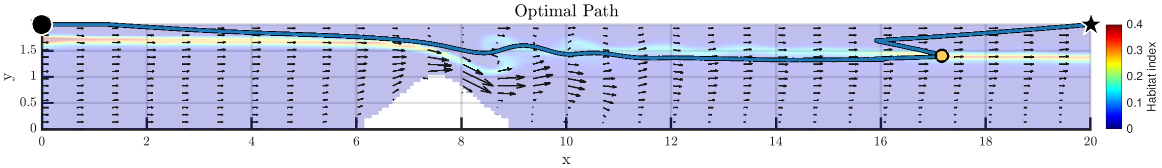


Figure 6-18: The final time optimal path for the fish farm to reach the destination with a non-dimensional mass of fish of $m_{tuna} = 0.9$. Shown in the background is the habitat index field at an intermediary non-dimensional time when the vehicle is at the yellow point shown along the path.

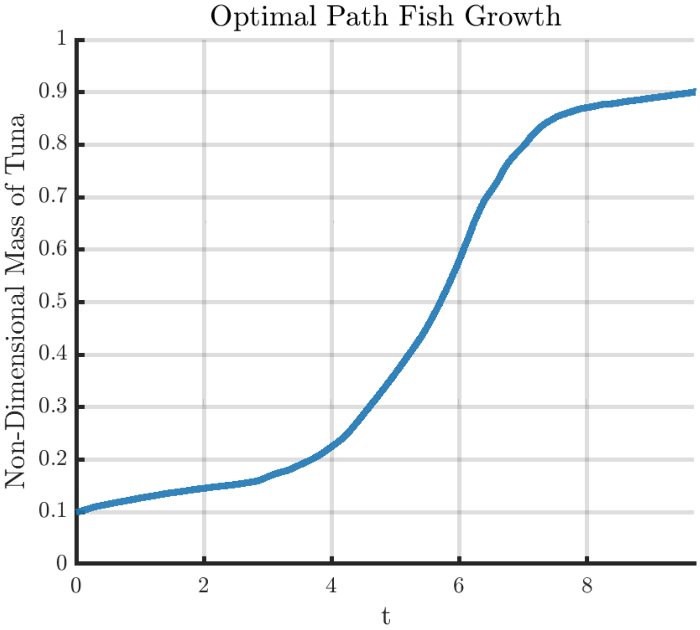


Figure 6-19: The time evolution of the non-dimensional mass of tuna in the fish farm when proceeding along the optimal path.

path for the farm to reach the target while having grown a non-dimensional tuna mass of 0.9 when starting from a initial tuna mass of 0.1. This corresponds to a specific point on the Pareto front (the point corresponding to the non-dimensional mass of $m^*_{tuna,f} = 0.9$).

For this simulation, the augmented state space was discretized using 300 nodes in range, 30 nodes in depth, and 50 nodes along the m_{tuna} axis. For the forward solve, a second order ENO scheme was used to compute all spatial derivatives and a second order TVD Runge-Kutta scheme was used to integrate in time using a fixed time step size of $\Delta t = 1 \times 10^{-2}$. The maximum speed of the vehicle was set to non-dimensional values of $F_{x,max} = 2.0$ and $F_{y,max} = 0.2$ along each coordinate direction.

The vehicle was further tasked with navigating from a start point $(x, y) = (0, 2)$ to a target $(x, y) = (20, 2)$.

Shown in Figure 6-17 is the evolution of the reachability front. The reachability front can be seen to expand along the harvest dimension (i.e. mass of tuna dimension in the state space) as it reaches the relatively thin zone with high values of the habitat index. The final optimal path, computed via backtracking, is shown in Figure 6-18. The farm can be seen, as expected, to navigate near areas of higher habitat index in order to promote fish growth. The resulting time evolution of the non-dimensional mass of fish in the farm, for this optimal path, is shown in Figure 6-19. We emphasize, again, that this is the globally time optimal path for a fish farm to reach the destination with the required 0.9 non-dimensional mass of tuna; no other path exists that will reach the destination faster.

6.5 Summary

In this chapter, we have presented various applications of our harvest-time path planning methodology. We first considered a benchmark case involving a steady halfspace harvest region. This case allowed for a semi-analytical solution to be easily computed, which therefore allowed us to validate our approach. We then considered energy-time optimal path planning in a double-gyre flow field. Here, we looked at a vehicle, equipped with energy harvesting capabilities, to navigate in a domain with the objectives of minimizing arrival time and maximizing the amount of energy remaining at the destination. Next, we studied how our methodology could be applied to the problem of harmful algae bloom collection. Finally, we considered optimal path planning for the case of an offshore moving fish farm – a next generation offshore aquaculture concept.

Chapter 7

Harvest-Time Optimal Path Planning Under Uncertainty: Preliminary Results

Our principle focus throughout this work has been on path planning in highly dynamic but deterministic environments. That is, the velocity fields and harvesting fields have been assumed to be exactly known, resulting in deterministic equations governing the dynamics of the system in the augmented state space. In reality, rarely is the ocean environment known perfectly. Instead, at best, probabilistic forecasts for the environment are known and, as a result, a truly robust path planning scheme must be able to account for this uncertainty. In this chapter, we address how our multi-objective harvest-time optimal path planning problem can be solved in the case of uncertain dynamic environments. As the work presented in this chapter is in progress, we present preliminary results and discuss avenues for future developments.

7.1 Problem Statement

We start by first defining the optimal path planning problem we plan to solve in the case of uncertain environments.

Consider a vehicle tasked with navigating in a physical domain $\mathcal{D} \subseteq \mathbb{R}^d$ from a start point $\boldsymbol{x}_s$ to a specified target point $\boldsymbol{x}_f$. Furthermore, consider a probability space $(\Omega, \mathcal{B}, \mathbb{P})$, where Ω is the sample space, $\mathcal{B}$ corresponds to the σ-algebra associated with Ω and $\mathbb{P}$ is the probability measure. The vehicle travels in an environment with a background dynamic flow that is uncertain, $\boldsymbol{V}(\boldsymbol{x}, t; \omega)$, where $\omega \in \Omega$ is a random event. As it traverses to the target, the vehicle has two controls at its disposal. The first is the time-dependent relative speed $F(t)$ given by $F(t) : [0, \infty) \to [0, F_{max}],\ F_{max} \in \mathbb{R}_{>0}$. The second is the heading $\hat{\boldsymbol{h}}(t)$ that it can choose to move in, which also may be time-dependent and can be represented with a unit vector:

$$\hat{\boldsymbol{h}}(t) : [0, \infty) \to \mathbb{R}^d \text{ such that } \left\|\hat{\boldsymbol{h}}(t)\right\| = 1,\ \forall t. \tag{7.1}$$

Using only these controls, the goal of the vehicle is to optimally solve the given path planning problem.

As before, in addition to its physical position in space, the vehicle has additional state variables of interest related to harvesting (denoted by $\boldsymbol{c} \in \mathbb{R}^{d_h}$). Altogether, we therefore have an augmented vehicle state space $\boldsymbol{x}^a = [\boldsymbol{x}, \boldsymbol{c}]^T$, a subset of $\mathbb{R}^{d+d_h}$, consisting of the physical position $\boldsymbol{x}$ and harvest state $\boldsymbol{c}$. For the uncertain realization ω, we denote the position and harvesting states of the vehicle at time t as $\boldsymbol{X}_p(t; \omega)$ and $\boldsymbol{C}_p(t; \omega)$ for an arbitrary continuous trajectory through the state space. The full, now stochastic, dynamics of the vehicle's state is of the following form:

$$\frac{\mathrm{d}\boldsymbol{X}_p(t;\omega)}{\mathrm{d}t} = \boldsymbol{V}\left(\boldsymbol{X}_p(t;\omega), t; \omega\right) + F(t)\, \hat{\boldsymbol{h}}(t), \tag{7.2a}$$

$$\frac{\mathrm{d}C_{p,i}(t;\omega)}{\mathrm{d}t} = \mathcal{H}_i(\boldsymbol{X}_p(t;\omega), \boldsymbol{C}_p(t;\omega), F(t), \hat{\boldsymbol{h}}(t), t; \omega), \quad i = 1, ..., d_h. \tag{7.2b}$$

Equation (7.2a) governs the motion of the vehicle through the uncertain physical domain. Equation (7.2b) corresponds to the general form of the dynamics of the harvesting states, which also in general will be uncertain due to the uncertainties in the environment and possibly within the harvesting fields themselves.

In this chapter, we consider only a single additional harvesting state, i.e. $d_h = 1$.

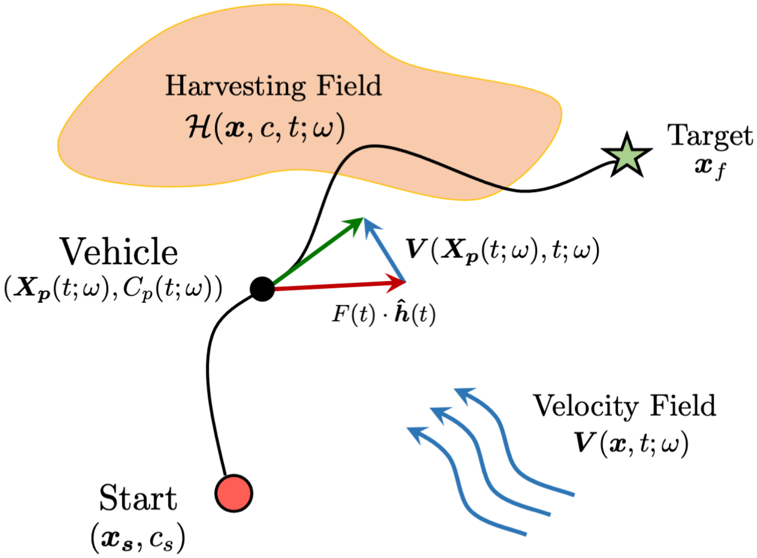


Figure 7-1: The stochastic harvest-time path planning problem of interest in this chapter.

Furthermore, we look at specialized harvesting dynamics of the form:

$$\frac{\mathrm{d}C_p(t;\omega)}{\mathrm{d}t} = \mathcal{H}(\boldsymbol{X}_p(t;\omega), C_p(t;\omega), t; \omega). \tag{7.3}$$

That is, we consider harvesting dynamics that do not depend on the vehicle controls. It can be noted that our special case of collecting a tracer field discussed in Chapter 3 is in this class of harvesting dynamics. Specifically, in the uncertain case, Equation (3.4) translates as follows:

$$\frac{\mathrm{d}C_p(t;\omega)}{\mathrm{d}t} = \alpha \cdot C_{field}(\boldsymbol{X}_p(t;\omega), t; \omega), \tag{7.4}$$

which indeed is of the form of Equation (7.3).

7.1.1 The Stochastic Harvest-Time Path Planning Problem

In this chapter, we consider the identical multi-objective optimization problem given by Equation (3.5) but now look to account for uncertainty. Specifically, our central aim is to determine the Pareto front for the corresponding multi-objective optimization problem in the uncertain domain. Let Θ_p denote the set of Pareto optimal solutions of the form (t_f^*, c_f^*) to the optimization problem given by Equation (3.5) for the case of a single harvesting state governed by the simplified harvesting dynamics that have no dependence on the vehicle controls. In the case of uncertainty in the environment, each uncertain realization ω would yield a different Pareto front for that given "universe". Altogether, these would then form the stochastic Pareto front $\Theta_p(\omega)$ to the uncertain multi-objective optimization problem. In this chapter, our primary aim is to efficiently compute this stochastic Pareto front to the path planning problem which, as we will show, can be completed using Hamilton-Jacobi reachability. In future work we plan to address how, after computing the Pareto front, one can compute the corresponding distribution of optimal paths via backtracking (for a chosen solution on the Pareto front) for this stochastic path planning problem.

7.2 Multi-Objective Harvest-Time Optimal Path Planning Under Uncertainty: Theory

Our approach to computing the stochastic Pareto front $\Theta_p(\omega)$ is a natural extension of the work presented in Chapter 4. Consider first the deterministic version of our problem at hand. For the case of harvesting dynamics of the form

$$\frac{\mathrm{d}C_p}{\mathrm{d}t} = \mathcal{H}(\boldsymbol{X}_p(t), C_p(t), t), \tag{7.5}$$

the corresponding Hamilton-Jacobi equation describing the evolution of the forward reachable set can be found using Equation (4.63) to yield:

$$\begin{aligned} \frac{\partial \phi(\boldsymbol{x}, c, t)}{\partial t} + F_{max} \left\| \frac{\partial \phi(\boldsymbol{x}, c, t)}{\partial \boldsymbol{x}} \right\| + \mathcal{H}(\boldsymbol{x}, c, t) \frac{\partial \phi(\boldsymbol{x}, c, t)}{\partial c} + \\ \boldsymbol{V}(\boldsymbol{x}, t)^T \cdot \frac{\partial \phi(\boldsymbol{x}, c, t)}{\partial \boldsymbol{x}} = 0, \\ \phi(\boldsymbol{x}, c, t = 0) = \phi_0(\boldsymbol{x}, c). \end{aligned} \tag{7.6}$$

The Pareto front in the deterministic case can then be simply computed using the approach outlined in Section 4.4.3.

In the case of uncertainty in the environment, we have a corresponding uncertain value function of the form $\phi(\boldsymbol{x}, c, t; \omega)$. This value function further satisfies the following stochastic Hamilton-Jacobi PDE:

$$\frac{\partial \phi(\boldsymbol{x}, c, t; \omega)}{\partial t} + F_{max} \left\| \frac{\partial \phi(\boldsymbol{x}, t, c; \omega)}{\partial \boldsymbol{x}} \right\| + \mathcal{H}(\boldsymbol{x}, c, t; \omega) \, \frac{\partial \phi(\boldsymbol{x}, t, c; \omega)}{\partial c} \\ + \boldsymbol{V}(\boldsymbol{x}, t; \omega)^T \cdot \frac{\partial \phi(\boldsymbol{x}, t, c; \omega)}{\partial \boldsymbol{x}} = 0, \tag{7.7}$$

with the initial condition

$$\phi(\boldsymbol{x}, c, t = 0; \omega) = \phi_0(\boldsymbol{x}, c; \omega). \tag{7.8}$$

Once a solution to Equation (7.7) is found, the stochastic Pareto front $\Theta_p(\omega)$ can then be determined by applying the approach outlined in Section 4.4.3 to each stochastic realization ω. The task of solving the uncertain multi-objective path planning problem therefore boils down to solving the above stochastic PDE (SPDE), which we now look to address.

7.3 Numerical Methods for Stochastic PDEs

Numerically computing solutions to SPDEs has been an active area of research for decades with several methods having been developed and refined (Xiu, 2010). Of these, one of the most widely used technique is Monte Carlo sampling. Here, the key principle is to generate independent realizations of the random inputs in the SPDE according to their corresponding probability distribution. For each draw, the resulting SPDE then turns into a deterministic PDE which can be solved using standard techniques. One can then collect an ensemble of solutions to these deterministic PDEs for each random input realization, and use this ensemble to compute relevant statistical information (Xiu, 2010).

In recent years, spectral methods have also gained traction in the uncertainty quantification community for solving SPDEs. The stochastic Galerkin and stochastic collocation methods, both built upon generalized polynomial chaos (gPC), are commonly used methods from this class of schemes (Xiu, 2010). gPC has several shortcomings, however, which are addressed and improved upon in a recently developed approach known as the Dynamically Orthogonal (DO) method (Sapsis and Lermusiaux, 2009; Ueckermann et al., 2013). In this chapter, we look at using the DO methodology to solve the given stochastic harvest-time optimal PDE in Equation (7.7), and we outline how this is done in the following section.

7.4 The Dynamically Orthogonal Equations

In this section we present the DO method for solving SPDEs. The presentation here follows closely (Sapsis and Lermusiaux, 2009; Subramani et al., 2018), and we refer the reader to these references for further details.

The DO method tackles the problem of solving a given SPDE by first considering a specific form of the stochastic solution. Consider the uncertain value function $\phi(\boldsymbol{x}, c, t; \omega)$ in Equation (7.7). This stochastic field can be represented using the following decomposition:

$$\phi(\boldsymbol{x}, c, t; \omega) = \bar{\phi}(\boldsymbol{x}, c, t) + \sum_{i=1}^{s} Y_i(t; \omega) \cdot \phi_i(\boldsymbol{x}, c, t). \tag{7.9}$$

Equation (7.9) is a generalized dynamic Karhunen-Loeve (KL) decomposition of the stochastic field known as a DO decomposition. Here, $\bar{\phi}$ is the mean of the stochastic field, given as $\bar{\phi}(\boldsymbol{x}, c, t) = \mathbb{E}^{\omega}[\phi(\boldsymbol{x}, c, t; \omega)]$, where the expectation operator is given as:

$$\mathbb{E}^{\omega}[\phi(\boldsymbol{x}, c, t; \omega)] = \int_{\Omega} \phi(\boldsymbol{x}, c, t; \omega) \cdot d\mathbb{P}(\omega). \tag{7.10}$$

Moreover, $\phi_i(\boldsymbol{x}, c, t)$ are mutually orthonormal fields referred to as DO modes. The modes are orthonormal relative to the spatial inner product defined as (for two arbitrary fields):

$$\Big\langle \psi_1(\boldsymbol{x}, c, t), \psi_2(\boldsymbol{x}, c, t) \Big\rangle = \int \psi_1(\boldsymbol{x}, c, t)\ \psi_2(\boldsymbol{x}, c, t)\ d\boldsymbol{x}\ dc. \tag{7.11}$$

The time-dependent random variables $Y_i(t;\omega)$ are stochastic processes, known as DO stochastic coefficients, which all have zero-mean: $\mathbb{E}^{\omega}[Y_i(t;\omega)] = 0$. The parameter s specifies the number of modes to be retained in the decomposition. While s can be increased for a better representation of the original stochastic field, this comes at the expense of increased computational cost.

The DO method looks to solve for the solution to the SPDE by deriving evolution equations for $\bar{\phi}(\boldsymbol{x}, c, t)$, $Y_i(t;\omega)$ and $\phi_i(\boldsymbol{x}, c, t)$, and evolving the components of the DO decomposition instead. These evolution equations are known as the DO equations, and they are derived directly from the original SPDE. For conciseness, we present the final DO equations corresponding to our stochastic Hamilton-Jacobi equation of interest without derivation (these can be readily derived by applying the procedures presented in Sapsis and Lermusiaux (2009) and in Subramani and Lermusiaux (2016); Subramani et al. (2018)).

7.4.1 Mean Equation

The evolution of the mean field, $\bar{\phi}$, is given by the following deterministic PDE:

$$\begin{aligned}
\frac{\partial \bar{\phi}}{\partial t} = &- F_{max} \cdot \mathbb{E}^{\omega}\left[\left\| \frac{\partial \phi}{\partial \boldsymbol{x}} \right\| \right] - \mathbb{E}^{\omega}[\boldsymbol{V}(\boldsymbol{x}, t;\omega)]^T \cdot \frac{\partial \bar{\phi}}{\partial \boldsymbol{x}} \\
&- \sum_{i=1}^{s} \left[\mathbb{E}^{\omega}[Y_i \cdot \boldsymbol{V}(\boldsymbol{x}, t;\omega)]^T \cdot \frac{\partial \phi_i}{\partial \boldsymbol{x}} \right] - \mathbb{E}^{\omega}[\mathcal{H}(\boldsymbol{x}, c, t;\omega)] \cdot \frac{\partial \bar{\phi}}{\partial c} \\
&- \sum_{i=1}^{s} \left[\mathbb{E}^{\omega}[Y_i \cdot \mathcal{H}(\boldsymbol{x}, c, t;\omega)] \cdot \frac{\partial \phi_i}{\partial c} \right].
\end{aligned} \tag{7.12}$$

In Equation (7.12), we have not explicitly listed the spatial, temporal and stochastic arguments for the mean, modes and stochastic coefficients for brevity of notation.

7.4.2 Modes Equations

The evolution of the modes ϕ_i is given by a set of deterministic PDEs known as the mode equations. They are given as follows:

$$\frac{\partial \phi_i}{\partial t} = C^{-1}_{Y_i(t)Y_k(t)} \cdot \left[\mathbb{E}^{\omega}[Y_k \cdot Q(\boldsymbol{x}, c, t; \omega)] - \left\langle \mathbb{E}^{\omega}[Y_k \cdot Q(\boldsymbol{x}, c, t; \omega)], \phi_n \right\rangle \phi_n \right], \tag{7.13}$$

where $Q(\boldsymbol{x}, c, t; \omega)$ is given as

$$\begin{aligned} Q(\boldsymbol{x}, c, t; \omega) = & - F_{max} \left\| \frac{\partial \phi}{\partial \boldsymbol{x}} \right\| - \boldsymbol{V}(\boldsymbol{x}, t; \omega)^T \cdot \frac{\partial \bar{\phi}}{\partial \boldsymbol{x}} \\ & - \sum_{i=1}^{s} \left[Y_i \cdot \boldsymbol{V}(\boldsymbol{x}, t; \omega)^T \cdot \frac{\partial \phi_i}{\partial \boldsymbol{x}} \right] - \mathcal{H}(\boldsymbol{x}, c, t; \omega) \frac{\partial \bar{\phi}}{\partial c} \\ & - \sum_{i=1}^{s} \left[Y_i \cdot \mathcal{H}(\boldsymbol{x}, c, t; \omega) \frac{\partial \phi_i}{\partial c} \right], \end{aligned} \tag{7.14}$$

and $C_{Y_i(t)Y_k(t)}$ is the covariance matrix given as $C_{Y_i(t)Y_k(t)} = \mathbb{E}^{\omega}[Y_i(t; \omega) \cdot Y_j(t; \omega)]$. In Equation (7.13), Einstein notation has been used. Moreover, the inner product $\langle \bullet, \bullet \rangle$ is given by Equation (7.11).

7.4.3 Stochastic Coefficients Equations

The stochastic coefficients Y_i evolve according to a set of stochastic ODEs. They are given as follows:

$$\frac{\mathrm{d}Y_i}{\mathrm{d}t} = \left\langle Q(\boldsymbol{x}, c, t; \omega), \phi_i \right\rangle - \left\langle \frac{\partial \bar{\phi}}{\partial t}, \phi_i \right\rangle. \tag{7.15}$$

7.4.4 Summary

We now summarize the key results presented in this section. Our central aim has been to solve the governing stochastic Hamilton-Jacobi equation given by Equation (7.7) for the value function $\phi(\boldsymbol{x}, c, t; \omega)$ which itself is a stochastic field. The DO method allows to solve for the evolution of the stochastic field $\phi(\boldsymbol{x}, c, t; \omega)$ by considering a special decomposition given by Equation (7.9). We then compute the solution to the

SPDE by evolving each of the components – $\bar{\phi}$, ϕ_i, and Y_i – of this decomposition, as governed by the DO equations (7.12), (7.13) and (7.15). With the stochastic value function computed, the stochastic Pareto front $\Theta_p(\omega)$ can then be readily obtained by sampling different random events $\omega^{(k)} \in \Omega$, computing the corresponding value function $\phi(\boldsymbol{x}, c, t; \omega^{(k)})$ for that realization, and applying the technique described in Section 4.4.3 to determine the Pareto front.

7.5 Numerical Implementation

The resulting DO equations given by Equations (7.12), (7.13) and (7.15) must be numerically discretized and solved. While any sufficient numerical method could be used, we employed finite difference schemes to discretize each equation in space (the augmented space including physical space and the harvesting domain) and time just as we did in the deterministic case.

For the mean equation, Equation (7.12), first order finite difference schemes were used for all spatial derivatives. Due to the hyperbolic properties of the mean equation, upwinding was used to discretize all convective terms and the normal motion term ($\left\|\frac{\partial \phi}{\partial \boldsymbol{x}}\right\|$) – we refer the reader to (Osher and Sethian, 1988; Subramani and Lermusiaux, 2016) for further details. The mode equation, Equation (7.13), and coefficient equation, Equation (7.15), used central differences to approximate all spatial derivatives.

Finally, for all equations, explicit time stepping was used, along with a first order forward Euler scheme for the temporal discretization. We note that all of these choices are preliminary and careful numerical studies is needed to validate the corresponding implementation.

7.6 Results

In this section, we present very preliminary results obtained using our present implementation of the stochastic DO harvest-time optimal path planning equations and

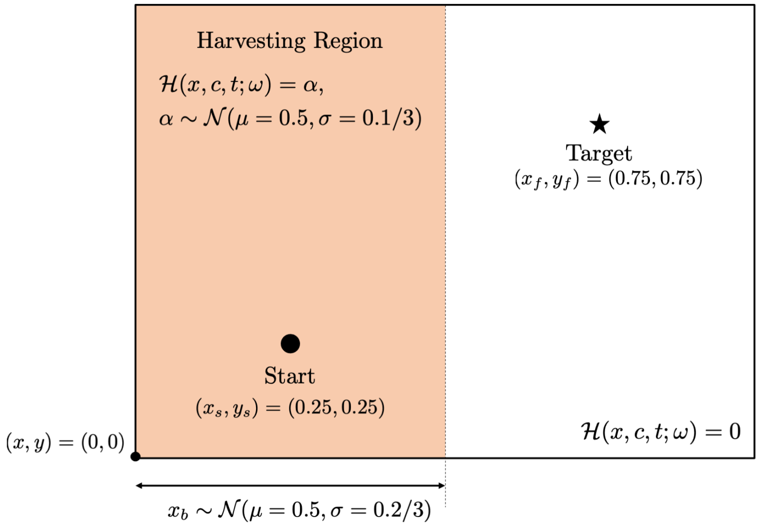


Figure 7-2: Schematic showing the case of interest.

numerical schemes.

7.6.1 Stochastic Halfspace Harvest Region Benchmark Case

As a first step in checking the validity of our approach, we considered the stochastic version of the benchmark problem presented in Section 6.1. The domain consists of a harvesting region that covers only one side of the physical space where the harvesting rate is a constant. The vehicle is tasked with harvesting from this field prior to exiting the harvesting region and proceeding to the target. Furthermore, the uncertainty in this problem arises from two sources. The first is the magnitude of the harvesting rate, which is assumed to be normally distributed. The second source of uncertainty is the boundary of the harvesting region which also is normally distributed around $x = 0.5$. The schematic in Figure 7-2 summarizes this case.

An augmented domain given by $(\boldsymbol{x}, c) \in [0, 1]^3$ was considered, with the domain discretized using 75 nodes in each coordinate dimension. Furthermore, a time step size of $\Delta t = 2.5 \cdot 10^{-3}$ was used. To keep computations tractable, only 5 modes were used in the DO decomposition, and the solution was evolved to a final non-dimensional time of $t = 1.7$. Moreover, 1000 realizations of the stochastic coefficients were used.

Figure 7-3 depicts the mean field, $\bar{\phi}(\boldsymbol{x}, c, t)$, at the final non-dimensional time of

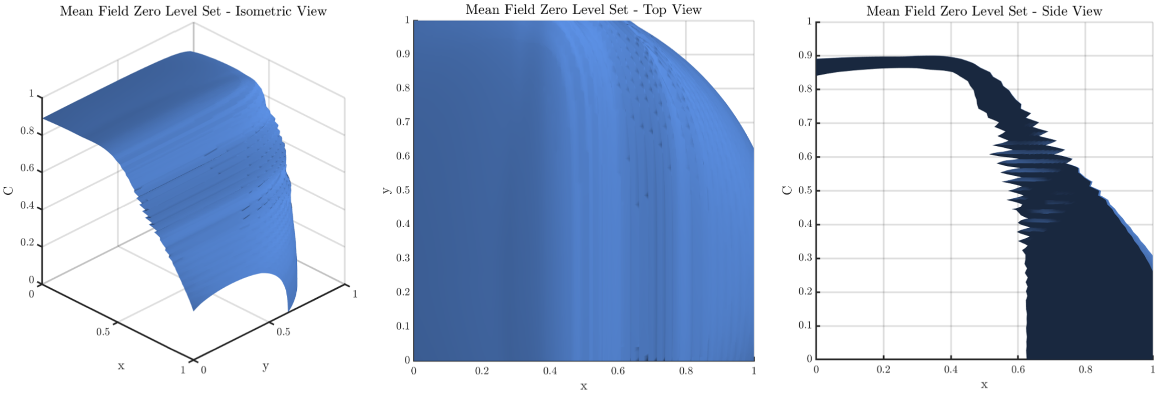


Figure 7-3: Reachability front from the mean field ($\boldsymbol{x} : \bar{\phi}(\boldsymbol{x}, c, t_f) = 0$) at the final non-dimensional time of $t_f = 1.7$.

$t = 1.7$ from the DO decomposition in Equation 7.9. Shown in the figure is the zero contour of the field (which corresponds to the mean reachability front). Figure 7-4 depicts the first four modes $\phi_i(\boldsymbol{x}, c, t)$ at the final non-dimensional time of $t = 1.7$. Finally the probability distribution of the stochastic coefficients $Y_i(t; \omega)$, at the final time $t = 1.7$, is shown in Figure 7-5.

As described in Section 7.2, the value function ϕ can be used to compute the Pareto front for the multi-objective optimization problem. Figure 7-6 depicts the stochastic Pareto front obtained for this given problem. Shown is the mean Pareto front along with a shaded region covering one standard deviation. The Pareto front as obtained from using the DO method is shown and compared to the Pareto front as determined using standard Monte Carlo (1000 Monte Carlo realizations were used). Good agreement can be noted between these solutions for later arrival times.

Furthermore, three different arrival times were considered, and the distribution of the optimal harvest state at the destination for each of these arrival times is shown in Figure 7-6. That is, each PDF depicts the distribution of Pareto optimal solutions (the best-case max possible harvested state of the vehicle travelling using the perfect optimal controls given the environment) for a given arrival time through this uncertain environment. Better agreement is noted between the DO and Monte Carlo solutions at later arrival times.

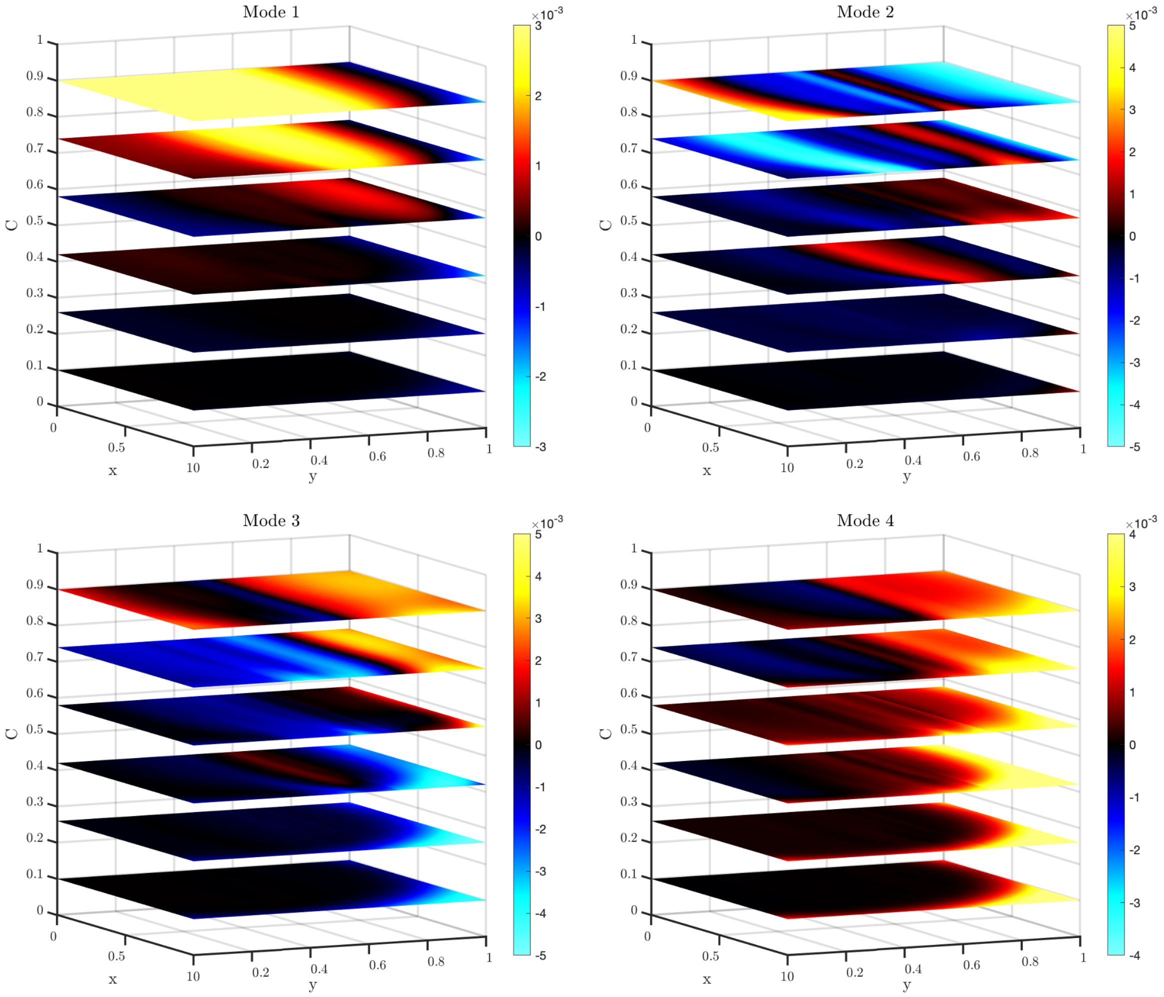


Figure 7-4: Cross sections of the first few modes $\phi_i(\boldsymbol{x}, c, t)$ at the final non-dimensional time of $t = 1.7$.

7.7 Discussion and Future Work

The stochastic best-case Pareto fronts hold important information about the environment and how it affects the vehicle mission. For instance, a low variance implies that the stochasticity in the background field does not significantly affect the amount of field that a vehicle can collect. Of course, to achieve the collection predicted by the Pareto front, the vehicle would have to choose the correct optimal control law for the given environment. This is difficult to do since the vehicle does not know the stochastic "draw" for the current environment. This best case Pareto front thus becomes an important benchmark to aim for when designing optimal controllers for the stochastic environment. In general, the concept of risk becomes the important

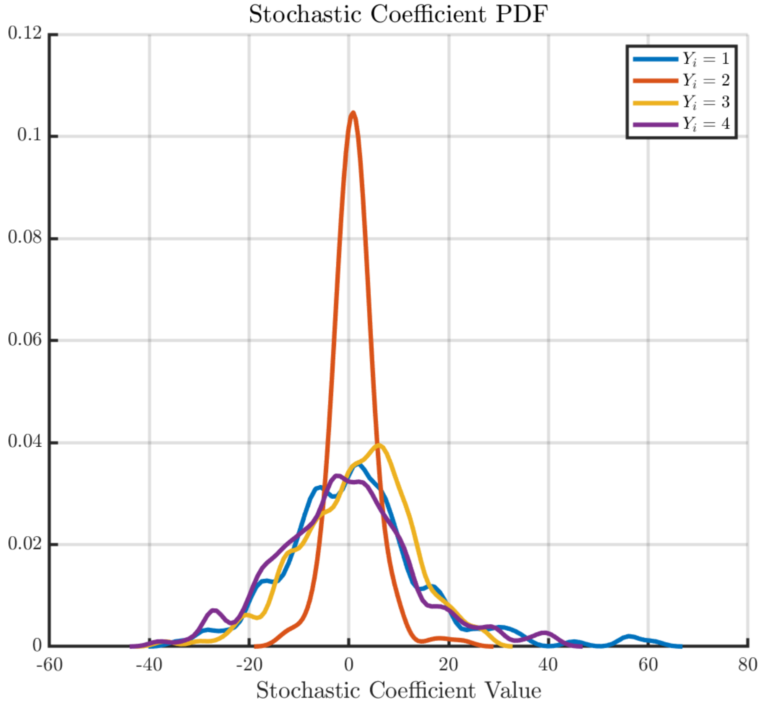


Figure 7-5: The PDF of the first few stochastic coefficients $Y_i(t;\omega)$ at the final nondimensional time of $t = 1.7$.

one, and this can be computed by our approach (Subramani and Lermusiaux, 2019), with a final answer dependent on the risk profile of the user.

A key challenge encountered when using the DO method was the production of spurious oscillations in our solutions. This can be especially noted in Figure 7-3, where oscillations can be seen in the mean reachability front at the final time. We have been actively investigating the possible sources of this. A few possible reasons are as follows. The first is that the discontinuous nature of the harvesting field considered in the case of interest may be a central reason for the production of these oscillations. We compared the DO update to $\phi(\boldsymbol{x}, c, t; \omega)$ to the MC update after a few timesteps and confirmed our suspicions, as our present numerical implementation of the DO solution was adding spurious oscillations to ϕ near the sharp boundary. Preliminary results on smoother harvesting fields have shown more promising value function fields. A second reason for these oscillations is the relatively low number of modes used for the cases considered, as a higher number of modes seems to reduce the intensity of these oscillations. Additional numerical studies and careful verification

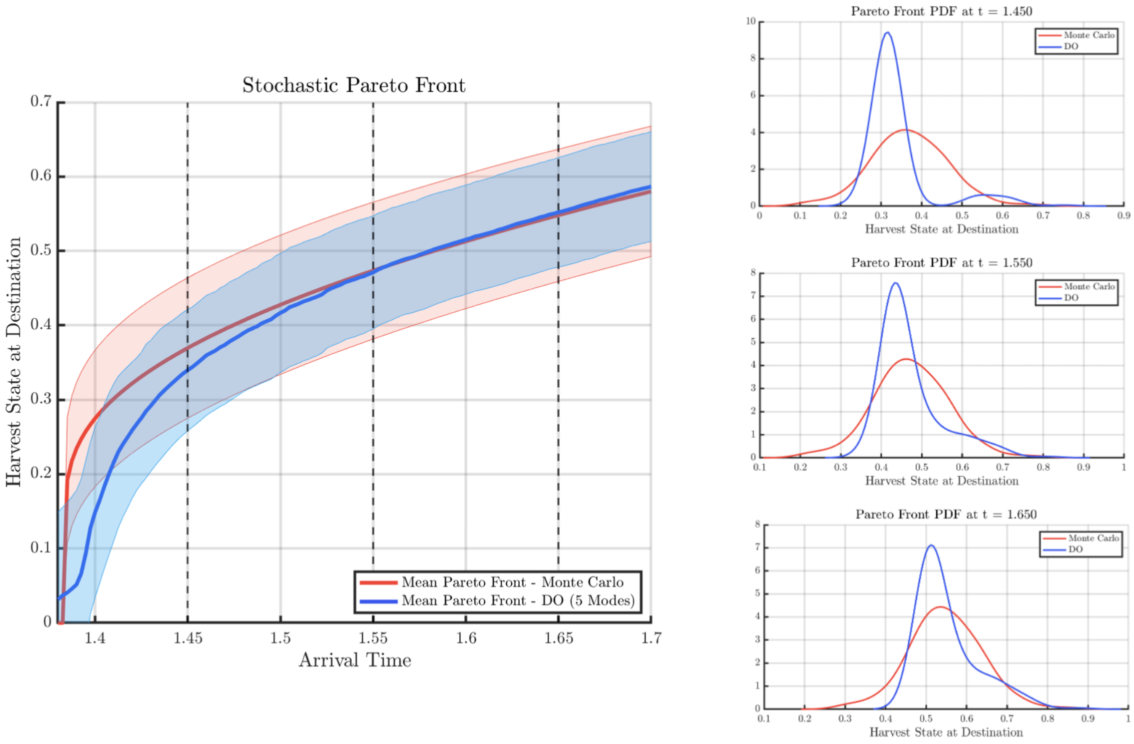


Figure 7-6: The stochastic best case Pareto front for the case of interest. Shown is the mean front along with a shaded region covering ±σ. Results obtained from the DO method are shown and compared to those found using Monte Carlo. Shown also are the PDFs of Pareto optimal solutions at 3 distinct times.

of our numerical implementation are needed.

In future work, we plan to test our approach in more realistic dynamic environments representative of the conditions autonomous marine vehicles would encounter in the open ocean. Additionally, we plan to apply the DO methodology to handle more general forms of harvesting dynamics. Furthermore, a key theoretical property we intend to address is on viscosity solutions in the context of stochastic Hamilton-Jacobi equations. In particular, we plan to study in detail what such a viscosity solution would look like in the stochastic case, whether the our present DO approach would produce this and, if not, what modifications would be needed to obtain this solution. Finally, we also intend to study efficient techniques to backtrack the family of optimal paths for a given Pareto optimal solution by using the stochastic value function along with the corresponding stochastic backtracking ODE.

Chapter 8

Conclusions and Future Work

In this final chapter, we summarize the findings of this work as well as outline avenues for future work.

The last decade has seen staggering growth in the development and use of autonomous vehicles, especially in ocean applications. Specifically, AUVs and ASVs are becoming more commonplace in marine surveillance, sampling and prediction tasks, to name just a few of their many areas of applicability. Recently, these vehicles have been further fitted with tools to allow them to harvest background dynamic fields. Our focus has been on these harvesting vehicles and addressing how to compute optimal paths for them in marine environments.

The primary contribution of this work has been the development of exact equations and efficient numerical schemes to compute harvest-time optimal paths for autonomous vehicles navigating in strong and dynamic flow fields. The method presented is based on exact PDEs which thus removes the need for heuristics. Our methodology builds upon (Lolla et al., 2014b), which uses forward reachability to compute time optimal paths in dynamic flows using the level set method. We extend this approach using state augmentation to solve a higher dimensional reachability problem in the vehicle's spatial *and* harvesting coordinates. This, in turn, allows for the ability to solve the multi-objective optimal path planning problem of minimizing the arrival time of the vehicle while simultaneously maximizing the vehicle's harvest state. We have shown, in particular, how the forward reachability front's evolution in the augmented state space can be used to efficiently obtain the Pareto front for this multi-objective optimization problem. A user then has the freedom to select any Pareto optimal solution(s) from this front and, in the backtracking process, use the time history of the forward evolution of the reachability front to determine the corresponding optimal path as well as the associated optimal controls, i.e., the speed and heading function, to be used by the vehicle to navigate to the destination.

Our methodology, which has been presented for very general harvesting dynamics agnostic of any specific application, has a number of areas of applicability. For instance, it is directly applicable to newer ASVs and AUVs equipped with energy harvesting capabilities that allow them to harvest energy from solar, wind, and wave sources from the environment. Such vehicles can use our approach to exactly compute the reachable regions accounting for energy harvested along the way. One can also exactly compute the full Pareto front holding solutions that trade-off arrival time versus the energy of the vehicle remaining upon reaching the destination. The exact paths through the environment can then be computed for any Pareto optimal solution chosen from this Pareto front. Additionally, our approach also can be applied to vehicles tasked with collecting advecting tracer fields, such as plastic waste or oil from spills. In this case, our methodology is able to determine paths through environments that additionally optimize for how much of the given tracer is harvested prior to reaching the destination.

Several different applications have been numerically investigated and presented. The first was a validation case involving planning a path in an environment with no external flow containing a harvesting field covering half the domain. This case admitted a semi-analytical solution, thus allowing us to verify the correctness of our methodology. Our second case involved a more complicated, unsteady, flow field. In particular, an idealized double-gyre flow field, which is representative of ocean circulation at mid-latitude regions, was studied and energy-time optimal paths from a fixed start point to a fixed target were found. In our third case, we applied our methodology to the highly relevant problem of harmful algae bloom collection. Finally, we investigated optimal path planning for the case of an offshore moving fish farm – a next generation offshore aquaculture concept.

Many possible directions for future work exist. The first is properly addressing the effect of harvesting from the vehicle on the environment itself. In all the cases presented in this work, we assumed the vehicle to have a negligible effect on the field it is harvesting. In certain cases, such as energy harvesting, this is valid since the environment can be considered to be a giant energy reservoir. In others, such as when collecting a tracer field, this assumption breaks down in the case where the length scales of the vehicle approach those of the background tracer field. In these cases, the effect of the reduction in the tracer field due to the influence of the vehicle cannot be imposed until the final path is computed. To account for the dynamical effects of the vehicle on the harvest field in our approach, one possibility is to consider an optimal control problem in an augmented state space consisting of the vehicle (with its harvest states) *and* the environment. The dynamics of this augmented system would then be able to account for the coupling between the vehicle's position at a given location and the corresponding reduction in the tracer field at that point. While theoretically possible, such an optimal control problem may involve a large number of states (in addition to the vehicle's position and harvest states, it would include however many degrees of freedom are present in the environment to parametrize the tracer field). In some applications, such an approach may become computationally infeasible. To account for this, there may be avenues for reduced representations of the state space in order to decrease and adapt the overall dimensionality of the system (Feppon and Lermusiaux, 2019; Heuss et al., 2020; Charous and Lermusiaux, 2021b,a; Ryu et al., 2021).

A second direction for future work is building upon the preliminary results from Chapter 7 to handle the case of path planning in uncertain environments. A stochastic treatment of our problem would further open the door to data assimilative path planning Lermusiaux (2007); Lermusiaux et al. (2017a,b). In this case, a vehicle, as it traverses to the target, would be able to use measurements along the way of its state to assimilate and predict the correct flow field (and harvest fields), and accordingly modify its optimal path.

A third avenue for future work is in addressing the increased computational cost associated with considering many harvesting states. For these high-dimensional cases, Bellman's curse of dimensionality makes numerical solves computationally intractable. In order to address this, there has been recent work in the space of high-dimensional reachability on using state space decomposition (Chen et al., 2018). This approach looks to decompose the dynamics of non-linear systems into sub-systems in lower dimensional spaces that can be analyzed instead. Such techniques could prove to be applicable in our method as well in order to handle a large number of harvesting states.

A fourth and final direction for future work, following along the same lines of addressing Bellman's curse of dimensionality, is to consider machine learning approaches for solving high-dimensional Hamilton-Jacobi-Bellman equations. Machine learning (deep learning in particular) has started to be actively studied in recent years for solving high-dimensional Hamilton-Jacobi-Bellman equations (Bansal and Tomlin, 2020; Nakamura-Zimmerer et al., 2021; Nüsken and Richter, 2021), and these techniques may prove to be highly applicable to our path planning framework.

Bibliography

M. D. Ageev. An analysis of long-range auv, powered by solar energy. In *'Challenges of Our Changing Global Environment'. Conference Proceedings. OCEANS'95 MTS/IEEE*, volume 2, pages 809–813. IEEE, 1995.

M. D. Ageev. Solar and wave powered auv concept. In *Proceedings of the 2000 International Symposium on Underwater Technology (Cat. No. 00EX418)*, pages 136–139. IEEE, 2000.

A. Alvarez, A. Caiti, and R. Onken. Evolutionary path planning for autonomous underwater vehicles in a variable ocean. *IEEE Journal of Oceanic Engineering*, 29 (2):418–429, 2004.

D. M. Anderson. Approaches to monitoring, control and management of harmful algal blooms (HABs). *Ocean & coastal management*, 52(7):342, July 2009. ISSN 0964-5691. doi: 10.1016/j.ocecoaman.2009.04.006. URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2818325/`.

A. L. Andrady. *Plastics and the Environment.* John Wiley & Sons, 2003.

AutoNaut Ltd. Autonaut, 2019. URL `http://www.autonautusv.com/`.

S. M. J. Baban. Trophic classification and ecosystem checking of lakes using remotely sensed information. *Hydrological Sciences Journal*, 41(6):939–957, Dec. 1996. ISSN 0262-6667.

S. Bansal and C. Tomlin. Deepreach: A deep learning approach to high-dimensional reachability. *arXiv preprint arXiv:2011.02082*, 2020.

S. Bansal, M. Chen, S. Herbert, and C. J. Tomlin. Hamilton-jacobi reachability: A brief overview and recent advances. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 2242–2253. IEEE, 2017.

M. Bardi and I. Capuzzo-Dolcetta. *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations.* Springer Science & Business Media, 2008.

A. Bayat, S. F. Aghamiri, A. Moheb, and G. R. Vakili-Nezhaad. Oil spill cleanup from sea water by sorbent materials. *Chemical Engineering & Technology: Industrial Chemistry-Plant Equipment-Process Engineering-Biotechnology*, 28(12):1525–1528, 2005.

Ş. T. Beşiktepe, P. F. J. Lermusiaux, and A. R. Robinson. Coupled physical and biogeochemical data-driven simulations of Massachusetts Bay in late summer: Real-time and post-cruise data assimilation. *Journal of Marine Systems*, 40–41:171–212, 2003. doi: 10.1016/S0924-7963(03)00018-6.

G. Bekey, R. Ambrose, V. Kumar, A. Sanderson, B. Wilcox, and Y. Zheng. International assessment of research and development in robotics. *World Technology Evaluation Center, Inc*, 2006.

R. Bellman. Dynamic programming. *Science*, 153(3731):34–37, 1966.

R. Bellman and R. E. Kalaba. *Dynamic programming and modern control theory*, volume 81. Citeseer, 1965.

G. Bendoricchio, G. Coffaro, and C. De Marchi. A trophic model for Ulva rigida in the Lagoon of Venice. *Ecological Modelling*, 75-76:485–496, Sept. 1994. ISSN 0304-3800. doi: 10.1016/0304-3800(94)90042-6. URL `http://www.sciencedirect.com/science/article/pii/0304380094900426`.

D. D. Benetti, G. I. Benetti, J. A. Rivera, B. Sardenberg, and B. O'Hanlon. Site selection criteria for open ocean aquaculture. *Marine Technology Society Journal*, 44(3):22–35, 2010.

D. P. Bertsekas. *Dynamic programming and optimal control: Vol. 1*. Athena scientific Belmont, 2000.

M. S. Bhabra, M. Doshi, B. C. Koenig, P. J. Haley, Jr., C. Mirabito, P. F. J. Lermusiaux, C. A. Goudey, J. Curcio, D. Manganelli, and H. Goudey. Optimal harvesting with autonomous tow vessels for offshore macroalgae farming. In *OCEANS 2020 IEEE/MTS*, pages 1–10. IEEE, Oct. 2020. doi: 10.1109/IEEECONF38699.2020.9389474.

D. R. Blidberg and M. D. Ageev. Solar powered autonomous underwater vehicles. In G. Griffiths, editor, *Technology and applications of autonomous underwater vehicles*, pages 59–76. Taylor & Francis, 2002. URL `https://eprints.soton.ac.uk/22121/`.

S. B. Borrelle, J. Ringma, K. L. Law, C. C. Monnahan, L. Lebreton, A. McGivern, E. Murphy, J. Jambeck, G. H. Leonard, M. A. Hilleary, et al. Predicted growth in plastic waste exceeds efforts to mitigate plastic pollution. *Science*, 369(6510): 1515–1518, 2020.

S. Boyd, S. P. Boyd, and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

W. Brogan. *Modern Control Theory*. Prentice Hall, 1991. ISBN 9780135897638. URL `https://books.google.com/books?id=0PFQAAAAMAAJ`.

J. Cao, J. Cao, Z. Zeng, and L. Lian. Optimal path planning of underwater glider in 3d dubins motion with minimal energy consumption. In *OCEANS 2016-Shanghai*, pages 1–7. IEEE, 2016.

K. P. Carroll, S. R. McClaran, E. L. Nelson, D. M. Barnett, D. K. Friesen, and G. N. William. Auv path planning: an a* approach to path planning with consideration of variable vehicle speeds and multiple, overlapping, time-dependent exclusion zones. In *Proceedings of the 1992 Symposium on Autonomous Underwater Vehicle Technology*, pages 79–84. IEEE, 1992.

R. H. Charlier and J. R. Justus. Ocean energies: environmental, economic and technological aspects of alternative power sources. 1993.

A. Charous and P. F. J. Lermusiaux. Dynamically orthogonal differential equations for stochastic and reduced-order modeling of ocean acoustic propagation. In *OCEANS 2021 IEEE/MTS*, pages 1–2. IEEE, Sept. 2021a. In press.

A. Charous and P. F. J. Lermusiaux. Perturbative retractions with high-order convergence to the best low-rank approximation. *SIAM Journal on Scientific Computing*, 2021b. Sub-judice.

M. Chen and C. J. Tomlin. Hamilton–jacobi reachability: Some recent theoretical advances and applications in unmanned airspace management. *Annual Review of Control, Robotics, and Autonomous Systems*, 1:333–358, 2018.

M. Chen, S. L. Herbert, M. S. Vashishtha, S. Bansal, and C. J. Tomlin. Decomposition of reachable sets and tubes for a class of nonlinear systems. *IEEE Transactions on Automatic Control*, 63(11):3675–3688, 2018.

A. Cohen. Mit tests self-propelled cage for fish farming, Sep 2008. URL `https://news.mit.edu/2008/aquaculture-0902`.

M. E. G. D. Colin, T. F. Duda, L. A. te Raa, T. van Zon, P. J. Haley, Jr., P. F. J. Lermusiaux, W. G. Leslie, C. Mirabito, F. P. A. Lam, A. E. Newhall, Y.-T. Lin, and J. F. Lynch. Time-evolving acoustic propagation modeling in a complex ocean environment. In *OCEANS - Bergen, 2013 MTS/IEEE*, pages 1–9, 2013. doi: 10.1109/OCEANS-Bergen.2013.6608051.

G. Cossarini, P. F. J. Lermusiaux, and C. Solidoro. Lagoon of Venice ecosystem: Seasonal dynamics and environmental guidance with uncertainty analyses and error subspace data assimilation. *Journal of Geophysical Research: Oceans*, 114(C6), June 2009. doi: 10.1029/2008JC005080.

B. Coxworth. Liquid robotics announces next generation of wave glider sea-going robot, Apr 2013. URL `https://newatlas.com/liquid-robotics-wave-glider-sv3-aquatic-robot/26977/`.

D. M. Crimmins, C. T. Patty, M. A. Beliard, J. Baker, J. C. Jalbert, R. J. Komerska, S. G. Chappell, and D. R. Blidberg. Long-endurance test results of the solar-powered auv system. In *OCEANS 2006*, pages 1–5. IEEE, 2006.

J. A. Cummings and O. M. Smedstad. *Variational Data Assimilation for the Global Ocean*, pages 303–343. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

C. Denniston, A. Kumaraguru, and G. S. Sukhatme. Comparison of Path Planning Approaches for Harmful Algal Bloom Monitoring. In *OCEANS 2019 MTS/IEEE SEATTLE*, pages 1–9, Oct. 2019. doi: 10.23919/OCEANS40490.2019.8962687. ISSN: 0197-7385.

J. W. Doerffer. *Oil spill response in the marine environment*. Elsevier, 2013.

M. Doshi. Energy-time optimal path planning in strong dynamic flows. Master's thesis, Massachusetts Institute of Technology, Center for Computational Science and Engineering, Cambridge, Massachusetts, Feb. 2021.

M. M. Doshi, C. S. Kulkarni, W. H. Ali, A. Gupta, P. F. J. Lermusiaux, P. Zhan, I. Hoteit, and O. Knio. Flow maps and coherent sets for characterizing residence times and connectivity in lagoons and coral reefs: The case of the Red Sea. In *OCEANS 2019 MTS/IEEE SEATTLE*, pages 1–8, Seattle, Oct. 2019. IEEE. doi: 10.23919/OCEANS40490.2019.8962643.

G. D. Egbert and S. Y. Erofeeva. Efficient inverse modeling of barotropic ocean tides. *Journal of Atmospheric and Oceanic Technology*, 19(2):183–204, 2002.

G. D. Egbert and S. Y. Erofeeva. OSU tidal inversion, 2013.

M. Ehrgott. Multiobjective optimization. *Ai Magazine*, 29(4):47–47, 2008.

L. Evans. *Partial Differential Equations*. Graduate studies in mathematics. American Mathematical Society, 2010. ISBN 9780821849743. URL `https://books.google.ca/books?id=Xnu0o_EJrCQC`.

F. Feppon and P. F. J. Lermusiaux. The extrinsic geometry of dynamical systems tracking nonlinear matrix projections. *SIAM Journal on Matrix Analysis and Applications*, 40(2):814–844, 2019. doi: 10.1137/18M1192780.

D. L. Ferris, D. N. Subramani, C. S. Kulkarni, P. J. Haley, and P. F. J. Lermusiaux. Time-optimal multi-waypoint mission planning in dynamic environments. In *OCEANS Conference 2018*, Charleston, SC, Oct. 2018. IEEE. doi: 10.1109/OCEANS.2018.8604505.

M. Fingas. *The basics of oil spill cleanup*. CRC press, 2012.

M. Fingas. *Oil spill science and technology*. Gulf professional publishing, 2016.

H. Flores, A. Zuniga, N. H. Motlagh, M. Liyanage, M. Passananti, S. Tarkoma, M. Youssef, and P. Nurmi. Penguin: aquatic plastic pollution sensing using auvs. In *Proceedings of the 6th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*, pages 1–6, 2020.

K. J. Flynn and D. J. McGillicuddy. Modeling marine harmful algal blooms: Current status and future prospects. *Harmful Algal Blooms: A Compendium Desk Reference*, pages 115–134, 2018.

A. Gangopadhyay, P. F. Lermusiaux, L. Rosenfeld, A. R. Robinson, L. Calado, H. S. Kim, W. G. Leslie, and P. J. Haley, Jr. The California Current system: A multiscale overview and the development of a feature-oriented regional modeling system (FORMS). *Dynamics of Atmospheres and Oceans*, 52(1–2):131–169, Sept. 2011. doi: 10.1016/j.dynatmoce.2011.04.003. Special issue of Dynamics of Atmospheres and Oceans in honor of Prof. A. R. Robinson.

B. Garau, A. Alvarez, and G. Oliver. Path planning of autonomous underwater vehicles in current fields with complex spatial variability: an a* approach. In *Proceedings of the 2005 IEEE international conference on robotics and automation*, pages 194–198. IEEE, 2005.

T. Garlock, F. Asche, J. Anderson, T. Bjørndal, G. Kumar, K. Lorenzen, A. Ropicki, M. D. Smith, and R. Tveterås. A global blue revolution: Aquaculture growth across regions, species, and countries. *Reviews in Fisheries Science & Aquaculture*, 28(1): 107–116, 2020.

R. R. Gentry, S. E. Lester, C. V. Kappel, C. White, T. W. Bell, J. Stevens, and S. D. Gaines. Offshore aquaculture: spatial planning principles for sustainable development. *Ecology and evolution*, 7(2):733–743, 2017.

R. Geyer, J. R. Jambeck, and K. L. Law. Production, use, and fate of all plastics ever made. *Science advances*, 3(7):e1700782, 2017.

B. C. Gibb. *Plastics are forever.* PhD thesis, Nature Publishing Group, 2019.

C. Goudey. Design and analysis of a self-propelled open-ocean fish farm. In *conference proceedings Open Ocean Aquaculture '98, Corpus Christi, TX*, 1998.

A. W. Griffith and C. J. Gobler. Harmful algal blooms: A climate change co-stressor in marine and freshwater ecosystems. *Harmful Algae*, 91:101590, Jan. 2020. ISSN 1568-9883. doi: 10.1016/j.hal.2019.03.008. URL `http://www.sciencedirect.com/science/article/pii/S1568988319300344`.

G. Griffiths. *Technology and applications of autonomous underwater vehicles*, volume 2. CRC Press, 2002.

X. Guo, M. Ji, Z. Zhao, D. Wen, and W. Zhang. Global path planning and multi-objective path control for unmanned surface vehicle based on modified particle swarm optimization (pso) algorithm. *Ocean Engineering*, 216:107693, 2020.

A. Gupta, P. J. Haley, D. N. Subramani, and P. F. J. Lermusiaux. Fish modeling and Bayesian learning for the Lakshadweep Islands. In *OCEANS 2019 MTS/IEEE SEATTLE*, pages 1–10, Seattle, Oct. 2019. IEEE. doi: 10.23919/OCEANS40490.2019.8962892.

P. J. Haley, Jr. and P. F. J. Lermusiaux. Multiscale two-way embedding schemes for free-surface primitive equations in the "Multidisciplinary Simulation, Estimation and Assimilation System". *Ocean Dynamics*, 60(6):1497–1537, Dec. 2010. doi: 10.1007/s10236-010-0349-4.

P. J. Haley, Jr., P. F. J. Lermusiaux, A. R. Robinson, W. G. Leslie, O. Logoutov, G. Cossarini, X. S. Liang, P. Moreno, S. R. Ramp, J. D. Doyle, J. Bellingham, F. Chavez, and S. Johnston. Forecasting and reanalysis in the Monterey Bay/California Current region for the Autonomous Ocean Sampling Network-II experiment. *Deep Sea Research Part II: Topical Studies in Oceanography*, 56(3–5): 127–148, Feb. 2009. doi: 10.1016/j.dsr2.2008.08.010.

P. J. Haley, Jr., A. Agarwal, and P. F. J. Lermusiaux. Optimizing velocities and transports for complex coastal regions and archipelagos. *Ocean Modeling*, 89:1–28, 2015. doi: 10.1016/j.ocemod.2015.02.005.

P. J. Haley, Jr., A. Gupta, C. Mirabito, and P. F. J. Lermusiaux. Towards Bayesian ocean physical-biogeochemical-acidification prediction and learning systems for Massachusetts Bay. In *OCEANS 2020 IEEE/MTS*, pages 1–9. IEEE, Oct. 2020. doi: 10.1109/IEEECONF38699.2020.9389210.

J. S. Hesthaven and T. Warburton. *Nodal discontinuous Galerkin methods: algorithms, analysis, and applications*. Springer Science & Business Media, 2007.

J. P. Heuss, P. J. Haley, Jr., C. Mirabito, E. Coelho, M. C. Schönau, K. Heaney, and P. F. J. Lermusiaux. Reduced order modeling for stochastic prediction onboard autonomous platforms at sea. In *OCEANS 2020 IEEE/MTS*, pages 1–10. IEEE, Oct. 2020. doi: 10.1109/IEEECONF38699.2020.9389149.

I. Hoteit, Y. Abualnaja, S. Afzal, B. Ait-El-Fquih, T. Akylas, C. Antony, C. Dawson, K. Asfahani, R. J. Brewin, L. Cavaleri, I. Cerovecki, B. Cornuelle, S. Desamsetti, R. Attada, H. Dasari, J. Sanchez-Garrido, L. Genevier, M. El Gharamti, J. A. Gittings, E. Gokul, G. Gopalakrishnan, D. Guo, B. Hadri, M. Hadwiger, M. A. Hammoud, M. Hendershott, M. Hittawe, A. Karumuri, O. Knio, A. Köhl, S. Kortas, G. Krokos, R. Kunchala, L. Issa, I. Lakkis, S. Langodan, P. Lermusiaux, T. Luong, J. Ma, O. Le Maitre, M. Mazloff, S. El Mohtar, V. P. Papadopoulos, T. Platt, L. Pratt, N. Raboudi, M.-F. Racault, D. E. Raitsos, S. Razak, S. Sanikommu, S. Sathyendranath, S. Sofianos, A. Subramanian, R. Sun, E. Titi, H. Toye, G. Triantafyllou, K. Tsiaras, P. Vasou, Y. Viswanadhapalli, Y. Wang, F. Yao, P. Zhan, and G. Zodiatis. Towards an end-to-end analysis and prediction system for weather, climate, and marine applications in the Red Sea. *Bulletin of the American Meteorological Society*, 102(1):E99–E122, Jan. 2021. doi: 10.1175/BAMS-D-19-0005.1.

V. T. Huynh, M. Dunbabin, and R. N. Smith. Predictive motion planning for auvs subject to strong time-varying currents and forecasting uncertainties. In *2015 IEEE international conference on robotics and automation (ICRA)*, pages 1144–1151. IEEE, 2015.

J. Jalbert, J. Baker, J. Duchesney, P. Pietryka, W. Dalton, D. Blidberg, S. Chappell, R. Nitzel, and K. Holappa. A solar-powered autonomous underwater vehicle. In *Oceans 2003. Celebrating the Past... Teaming Toward the Future (IEEE Cat. No. 03CH37492)*, volume 2, pages 1132–1140. IEEE, 2003.

Joyce S. The dead zones: oxygen-starved coastal waters. *Environmental Health Perspectives*, 108(3):A120–A125, Mar. 2000. doi: 10.1289/ehp.108-a120. URL `https://ehp.niehs.nih.gov/doi/abs/10.1289/ehp.108-a120`. Publisher: Environmental Health Perspectives.

S. Karaman and E. Frazzoli. Sampling-based algorithms for optimal motion planning. *The international journal of robotics research*, 30(7):846–894, 2011.

S. Karki, M. Sultan, R. Elkadiri, and T. Elbayoumi. Mapping and forecasting onsets of harmful algal blooms using modis data over coastal waters surrounding charlotte county, florida. *Remote Sensing*, 10(10):1656, 2018.

D. Kidwell. Programmatic environmental assessment for the prevention, control, and mitigation of harmful algal blooms program. Technical report, National Oceanic and Atmospheric Administration, Feb. 2015.

D. E. Kirk. *Optimal control theory: an introduction.* Courier Corporation, 2004.

T.-B. Koay and M. Chitre. Energy-efficient path planning for fully propelled auvs in congested coastal waters. In *2013 MTS/IEEE OCEANS-Bergen*, pages 1–9. IEEE, 2013.

S. Kong, M. Tian, C. Qiu, Z. Wu, and J. Yu. Iwscr: An intelligent water surface cleaner robot for collecting floating garbage. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2020.

D. Kularatne, S. Bhattacharya, and M. A. Hsieh. Time and energy optimal path planning in general flows. In *Robotics: Science and Systems*, 2016.

C. S. Kulkarni and P. F. J. Lermusiaux. Three-dimensional time-optimal path planning in the ocean. *Ocean Modelling*, 152, Aug. 2020. doi: 10.1016/j.ocemod.2020.101644.

C. S. Kulkarni, P. J. Haley, Jr., P. F. J. Lermusiaux, A. Dutt, A. Gupta, C. Mirabito, D. N. Subramani, S. Jana, W. H. Ali, T. Peacock, C. M. Royo, A. Rzeznik, and R. Supekar. Real-time sediment plume modeling in the Southern California Bight. In *OCEANS Conference 2018*, Charleston, SC, Oct. 2018. IEEE. doi: 10.1109/OCEANS.2018.8653642.

A. B. Kurzhanski and P. Varaiya. Dynamic optimization for reachability problems. *Journal of Optimization Theory and Applications*, 108(2):227–251, 2001.

L3 ASV. Unmanned marine systems - asvs, usvs and autonomous boat control system, Jun 2021. URL `https://www.unmannedsystemstechnology.com/company/autonomous-surface-vehicles-ltd/`.

J.-C. Latombe. *Robot motion planning*, volume 124. Springer Science & Business Media, 2012.

S. M. LaValle. *Planning algorithms.* Cambridge university press, 2006.

S. M. LaValle et al. Rapidly-exploring random trees: A new tool for path planning. 1998.

K. L. Law. Plastics in the marine environment. *Annual review of marine science*, 9: 205–229, 2017.

S. P. Leatherman. What's driving the huge blooms of brown seaweed piling up on florida and caribbean beaches?, Aug 2021. URL `https://theconversation.com/whats-driving-the-huge-blooms-of-brown-seaweed-piling-up-on-florida-and-caribbean-beaches-163058`.

L. Lebreton, B. Slat, F. Ferrari, B. Sainte-Rose, J. Aitken, R. Marthouse, S. Hajbane, S. Cunsolo, A. Schwarz, A. Levivier, et al. Evidence that the great pacific garbage patch is rapidly accumulating plastic. *Scientific reports*, 8(1):1–15, 2018.

J. J. Leonard and A. Bahr. *"Autonomous Underwater Vehicle Navigation"*. "Springer International Publishing", "Cham", 2016. ISBN "978-3-319-16649-0".

P. F. J. Lermusiaux. Evolving the subspace of the three-dimensional multiscale ocean variability: Massachusetts Bay. *Journal of Marine Systems*, 29(1):385–422, 2001. doi: 10.1016/S0924-7963(01)00025-2.

P. F. J. Lermusiaux. On the mapping of multivariate geophysical fields: Sensitivities to size, scales, and dynamics. *Journal of Atmospheric and Oceanic Technology*, 19(10):1602–1637, 2002. doi: 10.1175/1520-0426(2002)019<1602:OTMOMG>2.0.CO;2.

P. F. J. Lermusiaux. Adaptive modeling, adaptive data assimilation and adaptive sampling. *Physica D: Nonlinear Phenomena*, 230(1):172–196, 2007. doi: 10.1016/j.physd.2007.02.014.

P. F. J. Lermusiaux. Numerical fluid mechanics. MIT OpenCourseWare, May 2015. URL `https://ocw.mit.edu/courses/mechanical-engineering/2-29-numerical-fluid-mechanics-spring-2015/lecture-notes-and-references/`.

P. F. J. Lermusiaux, C. Evangelinos, R. Tian, P. J. Haley, Jr, J. J. McCarthy, N. M. Patrikalakis, A. R. Robinson, and H. Schmidt. Adaptive coupled physical and biogeochemical ocean predictions: A conceptual basis. In *Computational Science - ICCS 2004*, volume 3038 of *Lecture Notes in Computer Science*, pages 685–692. Springer Berlin Heidelberg, 2004. ISBN 978-3-540-22116-6. doi: 10.1007/978-3-540-24688-6_89.

P. F. J. Lermusiaux, P. J. Haley, Jr, and N. K. Yilmaz. Environmental prediction, path planning and adaptive sampling: sensing and modeling for efficient ocean monitoring, management and pollution control. *Sea Technology*, 48(9):35–38, 2007.

P. F. J. Lermusiaux, P. J. Haley, W. G. Leslie, A. Agarwal, O. Logutov, and L. J. Burton. Multiscale physical and biological dynamics in the Philippine Archipelago: Predictions and processes. *Oceanography*, 24(1):70–89, 2011. doi: 10.5670/oceanog.2011.05. Special Issue on the Philippine Straits Dynamics Experiment.

P. F. J. Lermusiaux, T. Lolla, P. J. Haley, Jr., K. Yigit, M. P. Ueckermann, T. Sondergaard, and W. G. Leslie. Science of autonomy: Time-optimal path planning and adaptive sampling for swarms of ocean vehicles. In T. Curtin, editor, *Springer Handbook of Ocean Engineering: Autonomous Ocean Vehicles, Subsystems and Control*, chapter 21, pages 481–498. Springer, 2016.

P. F. J. Lermusiaux, P. J. Haley, Jr., S. Jana, A. Gupta, C. S. Kulkarni, C. Mirabito, W. H. Ali, D. N. Subramani, A. Dutt, J. Lin, A. Shcherbina, C. Lee, and A. Gangopadhyay. Optimal planning and sampling predictions for autonomous and Lagrangian platforms and sensors in the northern Arabian Sea. *Oceanography*, 30(2): 172–185, June 2017a. doi: 10.5670/oceanog.2017.242. Special issue on Autonomous and Lagrangian Platforms and Sensors (ALPS).

P. F. J. Lermusiaux, D. N. Subramani, J. Lin, C. S. Kulkarni, A. Gupta, A. Dutt, T. Lolla, P. J. Haley, Jr., W. H. Ali, C. Mirabito, and S. Jana. A future for intelligent autonomous ocean observing systems. *Journal of Marine Research*, 75 (6):765–813, Nov. 2017b. doi: 10.1357/002224017823524035. The Sea. Volume 17, The Science of Ocean Prediction, Part 2.

P. F. J. Lermusiaux, M. Doshi, C. S. Kulkarni, A. Gupta, P. J. Haley, Jr., C. Mirabito, F. Trotta, S. J. Levang, G. R. Flierl, J. Marshall, T. Peacock, and C. Noble. Plastic pollution in the coastal oceans: Characterization and modeling. In *OCEANS 2019 MTS/IEEE SEATTLE*, pages 1–10, Seattle, Oct. 2019. IEEE. doi: 10.23919/OCEANS40490.2019.8962786.

W. G. Leslie, A. R. Robinson, P. J. Haley, Jr, O. Logutov, P. A. Moreno, P. F. J. Lermusiaux, and E. Coelho. Verification and training of real-time forecasting of multi-scale ocean dynamics for maritime rapid environmental assessment. *Journal of Marine Systems*, 69(1):3–16, 2008. doi: 10.1016/j.jmarsys.2007.02.001.

D. Li, P. Wang, and L. Du. Path planning technologies for autonomous underwater vehicles-a review. *IEEE Access*, 7:9745–9768, 2018.

Z. Li, W. Liu, L.-E. Gao, L. Li, and F. Zhang. Path planning method for auv docking based on adaptive quantum-behaved particle swarm optimization. *IEEE Access*, 7: 78665–78674, 2019.

Liquid Robotics Inc., 2020. URL `https://www.liquid-robotics.com/`.

S. V. T. Lolla. Path planning in time dependent flows using level set methods. Master's thesis, Massachusetts Institute of Technology, Dept. of Mechanical Engineering, Cambridge, Massachusetts, Sept. 2012.

T. Lolla, M. P. Ueckermann, K. Yiğit, P. J. Haley, Jr., and P. F. J. Lermusiaux. Path planning in time dependent flow fields using level set methods. In *IEEE International Conference on Robotics and Automation (ICRA), 14-18 May 2012*, pages 166–173, 2012. doi: 10.1109/ICRA.2012.6225364.

T. Lolla, P. J. Haley, Jr., and P. F. J. Lermusiaux. Time-optimal path planning in dynamic flows using level set equations: Realistic applications. *Ocean Dynamics*, 64(10):1399–1417, 2014a. doi: 10.1007/s10236-014-0760-3.

T. Lolla, P. F. J. Lermusiaux, M. P. Ueckermann, and P. J. Haley, Jr. Time-optimal path planning in dynamic flows using level set equations: Theory and schemes. *Ocean Dynamics*, 64(10):1373–1397, 2014b. doi: 10.1007/s10236-014-0757-y.

T. Lolla, P. F. J. Lermusiaux, M. P. Ueckermann, and P. J. Haley, Jr. Modified level set approaches for the planning of time-optimal paths for swarms of ocean vehicles. MSEAS Report 15, Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA, USA, 2014c.

T. Lolla, P. J. Haley, Jr., and P. F. J. Lermusiaux. Path planning in multiscale ocean flows: Coordination and dynamic obstacles. *Ocean Modelling*, 94:46–66, 2015. doi: 10.1016/j.ocemod.2015.07.013.

H. Lomax, T. H. Pulliam, and D. W. Zingg. *Fundamentals of computational fluid dynamics*. Springer Science & Business Media, 2013.

C. Louime, J. Fortune, and G. Gervais. Sargassum invasion of coastal environments: a growing concern. *American Journal of Environmental Sciences*, 13(1):58–64, 2017.

C. C. Mann. The bluewater revolution, May 2004. URL `https://www.wired.com/2004/05/fish/?pg=2`.

R. T. Marler and J. S. Arora. Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization*, 26(6):369–395, 2004.

K. Miettinen. *Nonlinear multiobjective optimization*, volume 12. Springer Science & Business Media, 2012.

I. M. Mitchell. The flexible, extensible and efficient toolbox of level set methods. *Journal of Scientific Computing*, 35(2):300–329, 2008.

MSEAS Group. MSEAS Software, 2013. URL `http://mseas.mit.edu/software/`.

T. Nakamura-Zimmerer, Q. Gong, and W. Kang. Adaptive deep learning for high-dimensional hamilton–jacobi–bellman equations. *SIAM Journal on Scientific Computing*, 43(2):A1221–A1247, 2021.

National Centers for Environmental Prediction (NCEP). North American Mesoscale Forecast System (NAM), Sept. 2019.

N. Nüsken and L. Richter. Solving high-dimensional hamilton–jacobi–bellman pdes using neural networks: perspectives from the theory of controlled diffusions and measures on path space. *Partial Differential Equations and Applications*, 2(4): 1–48, 2021.

Ocean News and Technology. Autonaut to launch new 3.5 meter communications hub usv, Mar 2019. URL `https://www.oceannews.com/news/subsea-cable/autonaut-to-launch-new-3-5-meter-communications-hub-usv`.

R. Onken, A. Álvarez, V. Fernández, G. Vizoso, G. Basterretxea, J. Tintoré, P. Haley, Jr., and E. Nacini. A forecast experiment in the Balearic Sea. *Journal of Marine Systems*, 71(1-2):79–98, 2008.

S. Osher and R. P. Fedkiw. *Level set methods and dynamic implicit surfaces*, volume 1. Springer New York, 2005.

S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations. *Journal of computational physics*, 79(1):12–49, 1988.

C. Ostle, R. C. Thompson, D. Broughton, L. Gregory, M. Wootton, and D. G. Johns. The rise in ocean plastics evidenced from a 60-year time series. *Nature communications*, 10(1):1–6, 2019.

M. Pal, P. J. Yesankar, A. Dwivedi, and A. Qureshi. Biotic control of harmful algal blooms (HABs): A brief review. *Journal of Environmental Management*, 268: 110687, Aug. 2020. ISSN 0301-4797. doi: 10.1016/j.jenvman.2020.110687. URL `http://www.sciencedirect.com/science/article/pii/S0301479720306198`.

M. Panda, B. Das, B. Subudhi, and B. B. Pati. A comprehensive review of path planning algorithms for autonomous underwater vehicles. *International Journal of Automation and Computing*, 17(3):321–352, 2020.

A. Perez, R. Platt, G. Konidaris, L. Kaelbling, and T. Lozano-Perez. Lqr-rrt*: Optimal sampling-based motion planning with automatically derived extension heuristics. In *2012 IEEE International Conference on Robotics and Automation*, pages 2537–2542. IEEE, 2012.

C. Petres, Y. Pailhas, Y. Petillot, and D. Lane. Underwater path planing using fast marching algorithms. In *Europe Oceans 2005*, volume 2, pages 814–819. IEEE, 2005.

V. Plot and J.-Y. Georges. Plastic debris in a nesting leatherback turtle in french guiana. *Chelonian Conservation and Biology*, 9(2):267–270, 2010.

J. C. Prata, A. L. P. Silva, J. P. Da Costa, C. Mouneyrac, T. R. Walker, A. C. Duarte, and T. Rocha-Santos. Solutions and integrated strategies for the control and mitigation of plastic and microplastic pollution. *International journal of environmental research and public health*, 16(13):2411, 2019.

S. R. Ramp, P. F. J. Lermusiaux, I. Shulman, Y. Chao, R. E. Wolf, and F. L. Bahr. Oceanographic and atmospheric conditions on the continental shelf north of the Monterey Bay during August 2006. *Dynamics of Atmospheres and Oceans*, 52(1–2):192–223, Sept. 2011. doi: 10.1016/j.dynatmoce.2011.04.005. Special issue of Dynamics of Atmospheres and Oceans in honor of Prof. A. R. Robinson.

RanMarine Technology. Ranmarine, 2021. URL `https://www.ranmarine.io/`.

D. Rao and S. B. Williams. Large-scale path planning for underwater gliders in ocean currents. In *Australasian conference on robotics and automation (ACRA)*, pages 2–4, 2009.

G. Reader, J. Potter, and J. Hawley. The evolution of auv power systems. In *OCEANS'02 MTS/IEEE*, volume 1, pages 191–198. IEEE, 2002.

C. Rees. Asv's c-enduro lemusv begins uk sea trials, Dec 2013. URL `https://www.unmannedsystemstechnology.com/2013/12/asvs-c-enduro-lemusv-begins-uk-sea-trials/`.

J. Ren, N. Barr, K. Scheuer, D. Schiel, and J. Zeldis. A dynamic growth model of macroalgae: Application in an estuary recovering from treated wastewater and earthquake-driven eutrophication. *Estuarine, Coastal and Shelf Science*, 148, July 2014. doi: 10.1016/j.ecss.2014.06.014.

C. J. Rhodes. Plastic pollution and potential solutions. *Science progress*, 101(3): 207–260, 2018.

H. Ritchie and M. Roser. Plastic pollution. *Our World in Data*, 2018. https://ourworldindata.org/plastic-pollution.

I. C. Robbins, G. J. Kirkpatrick, S. M. Blackwell, J. Hillier, C. A. Knight, and M. A. Moline. Improved monitoring of HABs using autonomous underwater vehicles (AUV). *Harmful Algae*, 5(6):749–761, Dec. 2006. ISSN 1568-9883. doi: 10.1016/j.hal.2006.03.005. URL `http://www.sciencedirect.com/science/article/pii/S1568988306000370`.

G. N. Roberts and R. Sutton. *Advances in unmanned marine vehicles*, volume 69. Iet, 2006.

J. Rojas. Plastic waste is exponentially filling our oceans, but where are the robots? In *2018 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*, pages 1–6. IEEE, 2018.

D. L. Rudnick, R. E. Davis, C. C. Eriksen, D. M. Fratantoni, and M. J. Perry. Underwater gliders for ocean research. *Marine Technology Society Journal*, 38(2): 73–84, 2004.

T. Ryu, J. P. Heuss, P. J. Haley, Jr., C. Mirabito, E. Coelho, M. C. Schönau, K. Heaney, and P. F. J. Lermusiaux. Adaptive stochastic reduced order modeling for autonomous ocean platforms. In *OCEANS 2021 IEEE/MTS*, pages 1–2. IEEE, Sept. 2021. In press.

T. P. Sapsis and P. F. J. Lermusiaux. Dynamically orthogonal field equations for continuous stochastic dynamical systems. *Physica D: Nonlinear Phenomena*, 238 (23–24):2347–2360, Dec. 2009. doi: 10.1016/j.physd.2009.09.017.

R. Scammell. Optical remote sensing provides ocean oil spill data, May 2018. URL `https://www.offshore-technology.com/news/optical-remote-sensing-provides-ocean-oil-spill-data/`.

J. M. Schell, D. S. Goodwin, and A. N. Siuda. Recent sargassum inundation events in the caribbean: Shipboard observations reveal dominance of a previously rare form. *Oceanography*, 28(3):8–11, 2015.

D. G. I. Schmale, A. P. Ault, W. Saad, D. T. Scott, and J. A. Westrick. Perspectives on Harmful Algal Blooms (HABs) and the Cyberbiosecurity of Freshwater Systems. *Frontiers in Bioengineering and Biotechnology*, 7, 2019. ISSN 2296-4185. doi: 10.3389/fbioe.2019.00128. URL `https://www.frontiersin.org/articles/10.3389/fbioe.2019.00128/full`. Publisher: Frontiers.

H. Schmidt, J. G. Bellingham, M. Johnson, D. Herold, D. M. Farmer, and R. Pawlowicz. Real-time frontal mapping with auvs in a coastal environment. In *OCEANS 96 MTS/IEEE Conference Proceedings. The Coastal Ocean-Prospects for the 21st Century*, volume 3, pages 1094–1098. IEEE, 1996.

O. Schofield, J. Bosch, S. Glenn, G. Kirkpatrick, J. Kerfoot, S. Lohrenz, M. A. Moline, M. Oliver, and P. Bissett. Bio-optics in Integrated Ocean Observing Networks: Potential for Studying Harmful Algal Blooms. In *Real-Time Observation Systems for Ecosystem Dynamicsand Harmful Algal Blooms*, page 23. UNESCO, Jan. 2007.

M. Schwaab. How to make fish farming environmentally friendly, sustainable and successful: Column, Apr 2020. URL `https://www.tampabay.com/opinion/2020/04/26/how-to-make-fish-farming-environmentally-friendly-sustainable-and-successful-column/`.

J. A. Sethian. *Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science*, volume 3. Cambridge university press, 1999.

S. A. Sharkh, G. Griffiths, and A. Webb. Power sources for unmanned underwater vehicles. In G. Griffiths, editor, *Technology and applications of autonomous underwater vehicles*, pages 19–35. Taylor & Francis, 2002. URL `https://eprints.soton.ac.uk/22121/`.

C.-W. Shu. High order numerical methods for time dependent hamilton-jacobi equations. In *Mathematics and computation in imaging science and information processing*, pages 47–91. World Scientific, 2007.

C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of computational physics*, 77(2):439–471, 1988.

M. Sigler. The effects of plastic pollution on aquatic wildlife: current situations and future solutions. *Water, Air, & Soil Pollution*, 225(11):1–9, 2014.

SOS Carbon. Sos carbon, 2021. URL `https://soscarbon.com/`.

D. Soto and C. Wurmann. Offshore aquaculture: A needed new frontier for farmed fish at sea. In *The Future of Ocean Governance and Capacity Development*, pages 379–384. Brill Nijhoff, 2019.

B. A. Stauffer, H. A. Bowers, E. Buckley, T. W. Davis, T. H. Johengen, R. Kudela, M. A. McManus, H. Purcell, G. J. Smith, A. Vander Woude, and M. N. Tamburri. Considerations in Harmful Algal Bloom Research and Monitoring: Perspectives From a Consensus-Building Workshop and Technology Testing. *Frontiers in Marine Science*, 6, 2019. ISSN 2296-7745. doi: 10.3389/fmars.2019.00399. URL `https://www.frontiersin.org/articles/10.3389/fmars.2019.00399/full`. Publisher: Frontiers.

R. Subasinghe, D. Soto, and J. Jia. Global aquaculture and its role in sustainable development. *Reviews in Aquaculture*, 1(1):2–9, 2009.

D. N. Subramani. Energy Optimal Path Planning Using Stochastic Dynamically Orthogonal Level Set Equations. Master's thesis, Massachusetts Institute of Technology, Department of Mechanical Engineering, Cambridge, Massachusetts, September 2014.

D. N. Subramani and P. F. J. Lermusiaux. Energy-optimal path planning by stochastic dynamically orthogonal level-set optimization. *Ocean Modeling*, 100:57–77, 2016. doi: 10.1016/j.ocemod.2016.01.006.

D. N. Subramani and P. F. J. Lermusiaux. Risk-optimal path planning in stochastic dynamic environments. *Computer Methods in Applied Mechanics and Engineering*, 353:391–415, Aug. 2019. doi: 10.1016/j.cma.2019.04.033.

D. N. Subramani, P. F. J. Lermusiaux, P. J. Haley, Jr., C. Mirabito, S. Jana, C. S. Kulkarni, A. Girard, D. Wickman, J. Edwards, and J. Smith. Time-optimal path planning: Real-time sea exercises. In *Oceans '17 MTS/IEEE Conference*, Aberdeen, June 2017. doi: 10.1109/OCEANSE.2017.8084776.

D. N. Subramani, Q. J. Wei, and P. F. J. Lermusiaux. Stochastic time-optimal path-planning in uncertain, strong, and dynamic flows. *Computer Methods in Applied Mechanics and Engineering*, 333:218–237, 2018. doi: 10.1016/j.cma.2018.01.004.

C. S. Tan, R. Sutton, and J. Chudley. Quasi-random, manoeuvre-based motion planning algorithm for autonomous underwater vehicles. *IFAC Proceedings Volumes*, 38(1):103–108, 2005.

R. Tedrake. Underactuated robotics: Algorithms for walking, running, swimming, flying, and manipulation. *Course Notes for MIT 6.832*, 2021.

The Ocean Cleanup, Jul 2021. URL `https://theoceancleanup.com/`.

The SeaCleaners, n.d. URL `https://www.theseacleaners.org/`.

E. R. Twomey and R. P. Signell. Construction of a 3-arcsecond digital elevation model for the Gulf of Maine. Open-File Report 2011–1127, U.S. Geological Survey, 2013. URL `https://pubs.usgs.gov/of/2011/1127/`.

M. P. Ueckermann and P. F. J. Lermusiaux. High order schemes for 2D unsteady biogeochemical ocean models. *Ocean Dynamics*, 60(6):1415–1445, Dec. 2010. doi: 10.1007/s10236-010-0351-x.

M. P. Ueckermann and P. F. J. Lermusiaux. 2.29 Finite Volume MATLAB Framework Documentation. MSEAS Report 14, Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA, 2012. URL `http://mseas.mit.edu/?p=2567`.

M. P. Ueckermann, P. F. J. Lermusiaux, and T. P. Sapsis. Numerical schemes for dynamically orthogonal equations of stochastic fluid and ocean flows. *Journal of Computational Physics*, 233:272–294, Jan. 2013. doi: 10.1016/j.jcp.2012.08.041.

Unmanned Marine Systems. Asv c-enduro, autonomous surface vehicle, 2013. URL `https://www.unmannedsystemstechnology.com/wp-content/uploads/2013/12/C-Enduro-brochure.pdf`.

A. C. Vegter, M. Barletta, C. Beck, J. Borrero, H. Burton, M. L. Campbell, M. F. Costa, M. Eriksen, C. Eriksson, A. Estrades, et al. Global research priorities to mitigate plastic pollution impacts on marine wildlife. *Endangered Species Research*, 25(3):225–247, 2014.

P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, et al. Scipy 1.0: fundamental algorithms for scientific computing in python. *Nature methods*, 17(3):261–272, 2020.

D. Xiu. *Numerical methods for stochastic computations: a spectral method approach*. Princeton university press, 2010.

L. Xiujuan and S. Zhongke. Overview of multi-objective optimization methods. *Journal of Systems Engineering and Electronics*, 15(2):142–146, 2004.

E. Yong. Why waves of seaweed have been smothering caribbean beaches, Jul 2019. URL `https://www.theatlantic.com/science/archive/2019/07/great-atlantic-sargassum-belt-here-stay/593290/`.

Z. Zeng, K. Sammut, L. Lian, F. He, A. Lammas, and Y. Tang. A comparison of optimization techniques for auv path planning in environments with ocean currents. *Robotics and Autonomous Systems*, 82:61–72, 2016.