

TOPICS IN OPTIMAL CONTROL AND GAME THEORY

A Dissertation

Presented to the Faculty of the Graduate School
of Cornell University

in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy

by

Elliot Virgin Cartee

August 2020

© 2020 Elliot Virgin Cartee
ALL RIGHTS RESERVED

TOPICS IN OPTIMAL CONTROL AND GAME THEORY

Elliot Virgin Cartee, Ph.D.

Cornell University 2020

We consider a number of “non-classical” problems in optimal control and game theory, with special attention to random perturbations, trade-offs due to conflicting objectives, and possible adversarial interactions of many selfish agents.

First, we consider models of pedestrian flow, with the goal of guaranteeing that anisotropic interactions in the models were handled in an internally consistent manner. We focus on two classes of models, one based on inter-crowd interactions and another based on intra-crowd interactions. For both cases, we prove sufficient conditions under which the direction of motion is determined uniquely almost everywhere in the domain. For the inter-crowd models, we also prove sufficient conditions under which the Nash Equilibrium is guaranteed to be unique.

Next, we investigate time-dependent Surveillance-Evasion Games. These are modeled as a semi-infinite zero-sum game between two players: an Observer who chooses a probability distribution over a pre-defined set of patrol trajectories, and an Evader who chooses a probability distribution over all possible trajectories from a starting location to a given target set. We develop efficient numerical methods for finding the Nash Equilibrium of these games.

We also formulate control-theoretic models of environmental crime in protected areas, such as wildlife poaching or illegal logging. We carefully model the optimal control problem faced by extractors aiming to illegally extract resources from a protected area such as a national park. The extractors are assumed to rationally choose their paths in and out of the protected area in order to maximize their expected payoffs, balancing the resources obtained from extraction with their travel time and risk of detection by authorities.

Finally, we look into uncertainty quantification in piecewise-deterministic Markov processes (PDMPs), where the dynamics of the system switch between deterministic modes at random times. We develop efficient numerical methods for approximating the cumulative distribution function (CDF) of the exit time, and bounds on this CDF when the transition rates between modes are only partially known. For controlled PDMPs, we further develop efficient numerical methods for optimizing this CDF.

BIOGRAPHICAL SKETCH

Elliot Cartee was born to parents Lianne Cartee and Lawrence Virgin on February 25, 1993, in Durham, North Carolina. There he went to school at Durham Academy and graduated from the North Carolina School of Science and Mathematics in 2011. In the fall of 2011, he moved to Ithaca, New York to attend Cornell University as an undergraduate, finishing with a Bachelor of Arts in Mathematics in 2014. Afterwards, he began graduate school at Cornell University, obtaining a Masters of Science in Mathematics in 2017, and receiving the Robert J. Bättig Graduate Prize for Excellent Achievements in Research in 2018. In the fall of 2020, he will begin his position as an L.E. Dickson Instructor in the Department of Mathematics at the University of Chicago.

ACKNOWLEDGEMENTS

This thesis would not have been possible without the generous support of many colleagues, family, friends, and others.

Firstly, I would like to thank my Ph.D. advisor Alexander Vladimirsky, whose guidance and mentorship has proved invaluable to me throughout my graduate studies. I also have benefited greatly from the support and advice of the faculty and staff in my many years in the Cornell Math department.

I would also like to thank my family for all of their support and advice. In particular my parents, Lianne and Lawrie, and my sister Hayley. I also never would have gotten this far without the support and encouragement of many friends throughout both my undergraduate and graduate years at Cornell, including Dan, Jared, Soham, Joe, Tyler, Mark, Teddy, Christine, Aaron, Ryan, and many others.

Chapter 4 was mostly written during the semester I spent visiting the ORFE department at Princeton University, and I would like to thank the students, faculty, and staff at ORFE for their warm welcome and hospitality.

The research in this thesis would never have been possible without many collaborators, and many insightful discussions with colleagues. First I want to thank Emiliano Cristiani, Yanqun Jiang, and Chi-Wang Shu for helpful discussions of their pedestrian flow models, which formed the basis of Chapter 2. Qianli (Charlie) Song, Lexiao (Sam) Lai, Casey Garner, and Tristan Reynoso provided valuable help in implementation and testing of the numerical codes used in Chapter 3 during the summer REU-2018 program at Cornell University. I am also indebted to Marc Gilles, whose algorithms and implementation for the stationary case of surveillance-evasion games served as a foundation for this work. I would also like to take the opportunity to thank all the authors of [8] for answering our questions about their model and for sharing the terrain data used in their numerical experiments, which proved to be very useful in Chapter 4. I am also grateful to Antonio Farah, Jacob van Hook, and

April Nellis for all of their hard work during (and after) the summer REU-2018 program at Cornell University, which proved invaluable to Chapter 5.

Finally, I would like to thank the NSF for their generous financial support of this work, through the grants DMS-1016150, DMS-1738010, DMS-0739164, and DMS-1645643.

TABLE OF CONTENTS

Biographical Sketch	iii
Acknowledgements	iv
Table of Contents	vi
1 Introduction	1
2 Anisotropic Challenges in Pedestrian Flow Modeling	6
2.1 Introduction	6
2.2 Optimal Control Formulation	9
2.3 Optimal directions and velocity profiles	12
2.3.1 Notes on efficiency in numerical implementation	14
2.3.2 Inter-crowd anisotropy	15
2.3.3 Intra-crowd anisotropy	22
2.4 Nash equilibria	26
2.4.1 NE in intercrowd models	29
2.4.2 Examples	31
2.4.3 Implementation notes for two-crowd models	35
2.5 Conclusions	37
3 Time-Dependent Surveillance-Evasion Games	40
3.1 Introduction	40
3.2 Evasive Path-Planning	41
3.3 Surveillance-Evasion Games	44
3.3.1 The “Evader-goes-second” problem	44
3.3.2 Nash Equilibria	47
3.4 Numerical Implementation	49
3.4.1 Numerics for HJB equation	49
3.4.2 Time-dependent Visibility	51
3.4.3 Time-dependent Optimal Path Tracer	51
3.5 Numerical Experiments	52
3.6 Conclusions	54
4 Control-Theoretic Models of Environmental Crime	57
4.1 Introduction	57
4.2 Prior Work	59
4.3 Extractor’s path-planning	63
4.3.1 Getting to the extraction point	66
4.3.2 Returning with the loot (Model G: ground patrols)	68
4.3.3 Returning with the loot (Model A: aerial patrols)	70
4.4 Multi-Objective Approach (Model A)	72
4.4.1 Pareto Front	73
4.4.2 Scalarized Eikonal	76
4.4.3 Auxiliary PDEs	77

4.5	Numerical Methods	78
4.5.1	Scalarized Eikonal Equation (Model A)	78
4.5.2	Randomly-terminated Eikonal equation (Model G)	81
4.6	Examples	82
4.6.1	Example 1: a circle	83
4.6.2	Example 2: banded patrol density on a square	84
4.6.3	Example 3: placing a patrol station	85
4.6.4	Example 4: allocating surveillance resources	86
4.6.5	Example 5: exploiting patrol gaps	89
4.6.6	Example 6: Yosemite National Park	89
4.6.7	Example 7: Kangaroo Island	93
4.7	Conclusions	93
5	Uncertainty Quantification and Management in Piecewise Deterministic Markov Processes	97
5.1	Introduction	97
5.2	Computing the CDF	100
5.2.1	Discrete PDMPs	100
5.2.2	Continuous PDMPs	103
5.2.3	Numerics for CDF computation	107
5.2.4	Experimental Results	111
5.3	Bounds on CDF	114
5.3.1	Deriving PDEs	116
5.3.2	Calculating Bounds	117
5.3.3	Experimental Results	118
5.4	Optimizing the CDF	119
5.4.1	Controlled PDMPs and expectation-optimal policies	122
5.4.2	PDEs for threshold-specific optimization	123
5.4.3	Discretization of PDEs and control synthesis	125
5.4.4	Numerical experiments	127
5.5	Conclusion	131
6	Conclusions	135
A	Velocity Profile for Linear Density Intra-Crowd Anisotropy	138
B	Convexity for Linear Density Intra-Crowd Anisotropy	140
	Bibliography	142

CHAPTER 1
INTRODUCTION

A classical problem in optimal control is finding minimum cost trajectories through a continuous domain. Suppose an agent's position $\mathbf{y}(t)$ in the domain Ω evolves according to the dynamics

$$\mathbf{y}'(t) = \mathbf{f}(\mathbf{y}(t), \mathbf{a}(t)), \quad \mathbf{y}(0) = \mathbf{x},$$

where $\mathbf{a} : \mathbb{R}^+ \rightarrow A$ is the agent's chosen control and A is the set of possible control values. Suppose also that the agent seeks to minimize the cost functional

$$\mathcal{J}(\mathbf{x}, \mathbf{a}(\cdot)) = \int_0^{T_{\mathbf{x}, \mathbf{a}(\cdot)}} K(\mathbf{y}(s), \mathbf{a}(s)) ds$$

where $T_{\mathbf{x}, \mathbf{a}(\cdot)} = \min\{t > 0 \mid \mathbf{y}(t) \in \partial\Omega\}$ is the exit time and K is a running cost. Standard results [9, 22] show that the value function

$$u(\mathbf{x}) = \inf_{\mathbf{a}(\cdot)} \{\mathcal{J}(\mathbf{x}, \mathbf{a}(\cdot))\}$$

is the unique viscosity solution [38] of the Hamilton-Jacobi-Bellman equation

$$\min_{\mathbf{a} \in A} \left\{ \mathbf{f}(\mathbf{x}, \mathbf{a}) \cdot \nabla u(\mathbf{x}) + K(\mathbf{x}, \mathbf{a}) \right\} = 0 \tag{1.1}$$

Equation (1.1) can be numerically solved using a number of efficient methods, including both Dijkstra-like methods [109, 111] and iterative methods [119] among others. This approach has been successfully used in many different problems across a wide range of applications.

However, the above assumes that the dynamics f and running cost K are fully known ahead of time. In many applications, these assumptions are too restrictive to be useful, particularly when there are interactions between multiple agents, or trade-offs between multiple (possibly conflicting) objectives must be taken into account. In this thesis we develop extensions of this classical optimal control approach to a number of new problems where the

dynamics and costs may not be known in advance, either due to game-theoretic interactions or random perturbations.

For instance, interactions between pedestrians are a key component of models of pedestrian flow. More specifically, each pedestrian’s dynamics are affected by other pedestrians, since they cannot move as quickly through crowded areas. In one popular macroscopic model put forth by Hughes [70], the crowd is described by a density $\rho(\mathbf{x}, t)$, and moves according to a velocity field $\mathbf{v}(\rho, \mathbf{a})$ where each pedestrian selfishly chooses their control \mathbf{a} in order to try and minimize the time required to reach a target set. In particular, we are interested in generalizations of Hughes’ model where the interactions are assumed to be anisotropic, as this presents a number of analytical challenges. One especially important concern is guaranteeing the “internal consistency” of the system by ensuring that the optimal choice of direction \mathbf{a} is uniquely defined almost everywhere.

In Chapter 2, we consider two different models with anisotropic interactions, one based on *inter-crowd* interactions [75], and another based on *intra-crowd* interactions [41]. For both cases, we prove sufficient conditions under which the optimal controls \mathbf{a}_* are determined uniquely almost everywhere in the domain. For the inter-crowd model, we are also able to prove sufficient conditions under which the Nash Equilibrium [94] is unique almost everywhere in the domain.

Another interesting class of problems arises when multiple agents interact not through the dynamics, but through the running cost. One particular example of this can be found in the framework of Surveillance-Evasion games as introduced in [56]. These are treated as a semi-infinite zero-sum game between two players: the first is an Observer who chooses a probability distribution λ over a predefined set of patrol trajectories. Each value of λ corresponds to a pointwise instantaneous observability function $K^\lambda(\mathbf{x})$. Meanwhile, the Evader chooses a probability distribution θ over infinitely many possible trajectories $\mathbf{y}(t)$ from a starting

location \mathbf{x}_0 to a given target set Q . This is a zero-sum game where the goal of the Observer is to choose λ to maximize the expected cumulative observability, while the Evader chooses θ in order to minimize this same quantity. It is also assumed that neither player is aware of the others' current location, and must commit to their probabilistic strategy in advance. In Chapter 3, we describe efficient numerical methods for finding the Nash Equilibrium of the semi-infinite zero-sum game, expanding the results of [56] from stationary Observer positions to time-dependent trajectories.

While both Chapter 2 and Chapter 3 focus on finding Nash Equilibria, where both players are assumed to act simultaneously, in some applications it is more accurate to assume that one player acts before the other. For instance, when modeling environmental crimes such as wildlife poaching and illegal logging, the standard assumption is that the authorities first choose a patrol strategy in order to reduce illegal resource extraction. The extractors then choose their extraction strategy based upon the patrol strategy used by the authorities.

In Chapter 4, we develop extensions of previous spatially-dependent models of environmental crime [3, 8], with particular focus on careful modeling of the optimal control problem faced by extractors. More specifically, we assume that the extractors must choose an extraction location, as well as paths to and from this extraction location in order to maximize their expected profit, balancing the amount of resources extracted with the risk of detection and travel costs due to terrain. We introduce two different models, one based on ground patrols and one based on aerial patrols, with differing information available to the perpetrator in each model. For both models, we also develop efficient numerical methods, and test them on examples based on real terrain data from Yosemite National Park in California, and Kangaroo Island in Australia.

For some of the applications discussed so far, randomness plays a significant role in the optimal control problems faced by agents. In such situations, the vast majority of the

optimal control literature has focused on optimizing only the average outcome. In Chapter 5, we instead show how to efficiently calculate the cumulative distribution function (CDF) of outcomes for a specific class of processes known as piecewise-deterministic Markov processes (PDMPs). In uncontrolled PDMPs, the state of the system is described by a continuous component $\mathbf{y} \in \Omega \subset \mathbb{R}^d$, and a discrete component $m \in \{1, 2, \dots, M\}$. The mode $m(t)$ switches randomly according to a continuous-time Markov process, where λ_{ij} gives the rate of $(i \rightarrow j)$ switching. Meanwhile, the position $\mathbf{y}(t)$ evolves according to $\mathbf{y}' = \mathbf{f}_{m(t)}(\mathbf{y}(t))$. For the initial state (\mathbf{x}, i) , the exit time $T_{\mathbf{x},i} = \min\{t > 0 \mid \mathbf{y}(t) \in \partial\Omega\}$. The CDFs $w_i(\mathbf{x}, s) = Pr(T_{\mathbf{x},i} \leq s)$ can be formally shown to satisfy a weakly-coupled system of hyperbolic linear PDEs. We also show how to efficiently compute bounds for these CDFs when the transition rates (λ_{ij}) are not fully known. We also consider the case of controlled dynamics $\mathbf{y}'(t) = \mathbf{f}_{m(t)}(\mathbf{y}(t), \mathbf{a}(t))$, where $\mathbf{a} : \mathbb{R}^{+,0} \rightarrow A$ is a measurable control function. For controlled PDMPs, we solve the problem of choosing controls so as to maximize the value of the CDF at any given threshold, and also develop efficient numerical algorithms for finding these controls by solving a nonlinear system of coupled PDEs.

Overall, we see that optimal control theory provides many powerful tools for solving path-planning problems, with a wide range of applications. Meanwhile, game theory gives us the ability to predict and understand non-cooperative interactions between many agents. Throughout this thesis we will explore many interesting problems at the intersection of these two fields, where agents aim to choose a path through some domain, while taking random perturbations and the behavior of other agents into account.

Chapters 2 and 4 are based on two papers coauthored with Alex Vladimirovsky [29, 30], while Chapter 3 is based on a paper together with Alex Vladimirovsky, Qianli (Charlie) Song, and Lexiao (Sam) Lai [27]. Chapter 5 is based on a paper with Antonio Farah, April Nellis, Jacob van Hook, and Alex Vladimirovsky that has been submitted to the SIAM Journal on

Uncertainty Quantification [28].

ANISOTROPIC CHALLENGES IN PEDESTRIAN FLOW MODELING

2.1 Introduction

Modeling the dynamics of crowds is a very active area in traffic engineering. A good overview and classification of most popular approaches can be found in [16]. For some of the macroscopic models incorporating individual choices of pedestrians, significant efforts have already been invested to build efficient implementations and conduct extensive numerical experiments (e.g., [68, 75, 124, 116, 74, 76, 41, 63, 61, 72, 73, 26]), despite the fact that some of the theoretical underpinnings are still missing. This includes both the well-posedness of PDE systems and the convergence under grid refinement of approximate solutions. Some practitioners may even argue that the issues of numerical convergence are not particularly relevant [63] since these PDEs are only approximations of crowd dynamics valid in intermediate asymptotics and the actual traffic engineering applications assume a specific length-scale. In the absence of rigorous error estimates, the models are judged instead phenomenologically – based on the perceived realism of their numerical predictions and on their ability to generate some of the self-organized behavior (e.g., “lane formation”) observed in real crowds. This tendency seems problematic given the blurring of boundaries between qualitative models (used to highlight the underlying mechanisms) and quantitative models (used to make specific predictions or even change the shape of the environment – to avoid stampedes or to improve the chances of a successful evacuation).

In macroscopic models the crowd is represented as a density function $\rho(\mathbf{x}, t)$, whose evolution is described by a conservation law. A typical (hyperbolic) version is

$$\rho_t(\mathbf{x}, t) + \nabla \cdot (\rho(\mathbf{x}, t)\mathbf{V}(\mathbf{x}, t)) = 0, \quad \mathbf{x} \in \Omega \subseteq \mathbb{R}^2, t > 0, \quad (2.1)$$

where \mathbf{V} is the velocity of motion for all pedestrians located at $\mathbf{x} \in \Omega$ at the time t . The initial density $\rho(\mathbf{x}, 0)$ is assumed to be known and additional boundary conditions are used to specify the obstacles and inflow/outflow rates through the exits. In many approaches, the velocity field \mathbf{V} is obtained by modeling pedestrians' individual choices – they plan their paths taking into account the current (and possibly future) crowd density on different parts of Ω ; see, for example, [68, 75, 41]. This path optimization is performed by solving a nonlinear Hamilton-Jacobi equation coupled to (2.1). The well-posedness of the resulting PDE systems is far from trivial and is an area of active research [55, 5, 99]. Our goals in this paper are much more modest: we simply aim to verify the internal/logical model consistency, by which we mean that the velocity field \mathbf{V} should be uniquely defined almost everywhere on Ω .

The internal consistency requirement is trivially satisfied when the pedestrians' velocities are isotropic; i.e., they can move in any direction with the same speed dependent on ρ at the current position (\mathbf{x}, t) . This is precisely the setting described by Hughes [70] and reviewed in section 2.2. However, anisotropic velocities arise naturally in many generalizations: due to the landscape (e.g., walking on a steep hill), or interactions of multiple crowds (e.g., [75, 74, 76, 61]), or non-local dependencies on ρ (e.g., [41]). The latter case presents new analytical challenges even if the paths are prescribed in advance rather than chosen by individual pedestrians. In dynamical particle-based models, anisotropic non-local interactions can yield non-smoothness and non-uniqueness of solutions to differential algebraic equations [50]. In macro-scale models of non-local interactions, the well-posedness of the multi-crowd PDE system requires additional assumptions [37]. In this paper we show that with individual path-planning and non-local interactions [41], the uniqueness of the velocity field is not guaranteed even for a single crowd. More generally, the notion of “optimal direction of motion” becomes far more delicate in all anisotropic models and can result in subtle mistakes in computational implementations.

A *velocity profile* $\mathcal{V}(\mathbf{x}, t)$ represents the set of all velocities that a pedestrian can use at a specific time and location; its formal definition is included in section 2.3, where we show that the strict convexity of \mathcal{V} is needed to guarantee that anisotropic models are internally consistent.

Unfortunately, computational experiments by themselves are insufficient to uncover model inconsistencies since they might be masked by the implementation details. Throughout this paper, we conduct a review of implicit assumptions in prior models. However, our real focus is on developing minimal consistency conditions, which should be easy to verify – either beforehand or in conjunction with any numerical implementation. Whether or not our conditions are actually satisfied in existing models depends on the crowd densities and parameter values. This brings us to a discussion of some of the experimental work to determine the latter (see section 2.3.2). We also include a review of strategies for a numerical implementation (sections 2.3.1 and 2.4.3) and suggest a new approach for decreasing the computational cost in the case of non-local dependencies on ρ (see Remark 1 and Appendix A).

For the 2-crowd scenarios, we argue that models are consistent only if there is no need to specify which crowd “goes first” in choosing its velocity field. This is essentially the requirement for the uniqueness of a Nash equilibrium in a non-zero-sum differential game, and in section 2.4 we derive the sufficient conditions to guarantee it.

We conclude by discussing several desirable extensions in section 2.5.

2.2 Optimal Control Formulation

Finding time-optimal trajectories to a target is among the most studied problems in optimal control theory. Here we provide a brief overview of the dynamic programming approach, focusing on the geometric interpretation and referring readers to [9, 22] for a detailed treatment. Suppose the local velocity depends on one's location and the chosen control value; i.e., $\mathbf{v} : \bar{\Omega} \times \mathcal{U} \mapsto \mathbb{R}^2$, where \mathcal{U} is a suitable compact set of control values. In our context, it will be enough to identify controls as the pedestrians' preferred directions of motion; i.e., $\mathcal{U} = S^1 = \{\mathbf{u} \in \mathbb{R}^2 \mid |\mathbf{u}| = 1\}$. We will focus on two types of pedestrian velocities:

$$\mathbf{v}(\mathbf{x}, \mathbf{u}) = v(\mathbf{x}, \mathbf{u})\mathbf{u}, \quad (2.2)$$

where $v > 0$ is the speed of motion through \mathbf{x} in the direction \mathbf{u} ; and

$$\mathbf{v}(\mathbf{x}, \mathbf{u}) = \mathbf{u} + \mathbf{w}_i(\mathbf{x}, \mathbf{u}), \quad (2.3)$$

where \mathbf{w}_i is a known “interactional velocity” perturbation.

The dynamics are governed by $\mathbf{y}'(t) = \mathbf{v}(\mathbf{y}(t), \mathbf{u})$, and we want to choose a feedback control function $\mathbf{u} = \mathbf{u}(\mathbf{y}(t))$ to minimize the time of travel to the target Γ_d . The *value function* $\varphi(\mathbf{x})$ is defined to be the minimum time to target from the starting position $\mathbf{y}(0) = \mathbf{x} \in \Omega$, and it can be recovered as the unique viscosity solution [38] of the Hamilton-Jacobi-Bellman PDE

$$\begin{aligned} \max_{\mathbf{u} \in \mathcal{U}} \{-\nabla \varphi(\mathbf{x}) \cdot \mathbf{v}(\mathbf{x}, \mathbf{u})\} &= 1, & \mathbf{x} \in \Omega \\ \varphi(\mathbf{x}) &= 0, & \mathbf{x} \in \Gamma_d \end{aligned} \quad (2.4)$$

with additional boundary conditions (e.g., $\varphi = +\infty$ on $\partial\Omega \setminus \Gamma_d$) to ensure that all trajectories stay inside Ω . The characteristics of this PDE are in fact the time-optimal trajectories and a time-optimal feedback control is obtained by choosing $\mathbf{u}(\mathbf{x})$ from the argmax set of equation (2.4).

The above describes path-planning for a pedestrian whose velocity is only affected by the current location and the chosen direction. But the dependence on crowd density can be treated similarly; e.g., with the speed $v(\rho, \mathbf{u})$ monotone decreasing in ρ . If the rest of the crowd is not moving and the current density $\hat{\rho}$ is known, our pedestrian can simply use $v(\mathbf{x}, \mathbf{u}) = v(\hat{\rho}(\mathbf{x}), \mathbf{u})$ to plan their path to Γ_d . We will refer to this approach as “stroboscopic” since in reality all pedestrians are moving using the same logic, but our planning is based on the current snapshot of the crowd density instead of trying to predict the correct/updated density at each point on the trajectory¹. So, the current φ is only good enough to choose the initial direction of motion and has to be continuously recomputed as the density changes. Putting this all together, our system of PDEs is

$$\rho_t(\mathbf{x}, t) + \nabla \cdot (\rho(\mathbf{x}, t)\mathbf{V}(\mathbf{x}, t)) = 0, \quad (2.5a)$$

$$\max_{\mathbf{u} \in \mathcal{U}} \{-\nabla\varphi(\mathbf{x}, t) \cdot \mathbf{v}(\rho(\mathbf{x}, t), \mathbf{u})\} = 1, \quad (2.5b)$$

$$\mathbf{u}_*(\mathbf{x}, t) \in \arg \max_{\mathbf{u} \in \mathcal{U}} \{-\nabla\varphi(\mathbf{x}, t) \cdot \mathbf{v}(\rho(\mathbf{x}, t), \mathbf{u})\}, \quad (2.5c)$$

$$\mathbf{V}(\mathbf{x}, t) = \mathbf{v}(\rho(\mathbf{x}, t), \mathbf{u}^*(\mathbf{x}, t)), \quad (2.5d)$$

with the specified initial density $\rho(\mathbf{x}, 0) = \rho_0(\mathbf{x})$ and $\varphi(\mathbf{x}, t) = 0$ on $\Gamma_d \times R$.

In the isotropic case (where $v(\rho, \mathbf{u}) = v(\rho)$), each $\varphi(\cdot, t)$ is a solution of the Eikonal equation:

$$|\nabla\varphi(\mathbf{x}, t)|v(\rho(\mathbf{x}, t)) = 1 \quad (2.6)$$

and the crowd velocity is $\mathbf{V}(\mathbf{x}, t) = -v(\rho(\mathbf{x}, t))\frac{\nabla\varphi(\mathbf{x}, t)}{|\nabla\varphi(\mathbf{x}, t)|} = -v^2(\rho(\mathbf{x}, t))\nabla\varphi(\mathbf{x}, t)$. (A typical isotropic evolution of crowd density on a domain with a single obstacle is shown in Figure 2.1.) This is essentially the set up considered in the influential paper by Hughes [70], who was among the first to model the influence of pedestrian’s choices on the macroscopic

¹The latter approach is also present in the literature [66, 67, 82, 24, 41, 99] and is related to *Mean Field Games* [83, 84, 69, 59, 58]. Many of the issues we discuss are relevant in that context as well, but we focus on the stroboscopic models for the sake of simplicity.

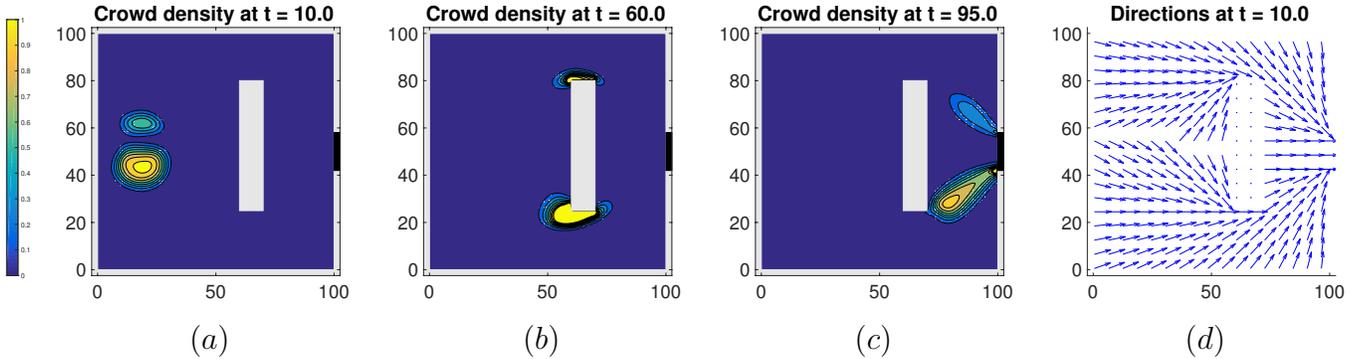


Figure 2.1: Simulation of a single (isotropic) crowd evacuating through a doorway (shown in black) on the right edge of the domain. Contour plots of ρ for various time-slices (in a,b,c) and a quiver plot of \mathbf{u}_* in d. The speed function is $v(\rho) = e^{-0.075\rho^2}$. Computed on a 100×100 grid with a timestep $\Delta t = 0.005$.

dynamics². In [70] the equations for “the potential” φ were derived based on a physical analogy: “Pedestrians have a common sense of the task (called potential) they face to reach their common destination such that any two individuals at different locations having the same potential would see no advantage to either of exchanging places. There is no perceived advantage of moving along a line of constant potential. Thus, the motion of any pedestrian is in the direction perpendicular to the potential [level sets].” However, this suggestive analogy is somewhat misleading. After all, the gradient of the potential indicates the direction of forces/accelerations, while we are choosing the direction for the velocity.

Hughes further claims that “Many physical situations are potentially both anisotropic and inhomogeneous. Fortunately, [this] does not change the derivation of the equations of motion and [the equivalent of $\mathbf{V}(\mathbf{x}, t) = -v^2(\rho(\mathbf{x}, t))\nabla\varphi(\mathbf{x}, t)$] with [the speed] redefined for the new arguments, remains valid.” But it is easy to see that this is not quite correct: in the general anisotropic setting, the optimal direction can be different from $\frac{-\nabla\varphi(\mathbf{x}, t)}{|\nabla\varphi(\mathbf{x}, t)|}$. (Consider the example of someone trying to swim to the opposite shore of a river of finite width and infinite length. Then, by symmetry, $\nabla\varphi$ points directly across the river from one shore to the

²The equation introduced by Hughes was slightly more general than the time-optimal path planning – it allowed for an additional high-density-discomfort/penalty factor in the Eikonal equation. We omit it here for the sake of simplicity.

other. However, the optimal path of the swimmer is to move in a diagonal direction, with the flow of the river, rather than fighting against the flow to go straight across. See also a detailed discussion and Figure 2.2 in section 2.3.2.) Even though anisotropic problems were not considered by Hughes in [70], we believe that this comment may have been responsible for the misidentification of optimal directions in several subsequent papers.

2.3 Optimal directions and velocity profiles

To evolve the crowd density ρ , it might seem natural to require that the velocity field \mathbf{V} should be uniquely defined for all (\mathbf{x}, t) . However, the arg max approach for choosing the optimal direction of motion assumes that $\nabla\varphi$ is well-defined. Unfortunately, equation (2.4) typically does not have a smooth/classical solution, and admits infinitely many weak solutions. Additional criteria based on test functions were introduced by Crandall and Lions in [38] to select the unique *viscosity solution*, coinciding with the value function of the corresponding control problem. For starting points where $\nabla\varphi$ is undefined, it is easy to see that optimal trajectories are usually non-unique. Suppose $\{\tilde{\mathbf{x}}_n\}$ and $\{\hat{\mathbf{x}}_n\}$ are two sequences converging to the same point \mathbf{x} , but $\tilde{\mathbf{p}} = \lim_{n \rightarrow \infty} \nabla\varphi(\tilde{\mathbf{x}}_n)$ is different from $\hat{\mathbf{p}} = \lim_{n \rightarrow \infty} \nabla\varphi(\hat{\mathbf{x}}_n)$. Then it is clear that the optimal direction of motion starting from \mathbf{x} will depend on whether we use $\tilde{\mathbf{p}}$ or $\hat{\mathbf{p}}$ in place of $\nabla\varphi$ in equation (2.4). Luckily, mild technical conditions on \mathbf{v} guarantee that the viscosity solution is Lipschitz-continuous (and, as a result, differentiable almost everywhere). Since the crowd density ρ is assumed to be in L^1 , it is sufficient to have \mathbf{V} uniquely defined almost everywhere, making the above complications less relevant.

Wherever φ is differentiable, the optimization in (2.4) has a simple geometric structure: if we view $\mathbf{n} = \frac{\nabla\varphi(\mathbf{x})}{|\nabla\varphi(\mathbf{x})|}$ as a known/fixed unit vector, $\mathbf{u} \in \mathcal{U}$ is simply chosen to maximize

the projection of $\mathbf{v}(\mathbf{x}, \mathbf{u})$ onto $(-\mathbf{n})$. Two important consequences of this are related to the shape of the *velocity profile* $\mathcal{V}(\mathbf{x}) = \{\mathbf{v}(\mathbf{x}, \mathbf{u}) \mid \mathbf{u} \in \mathcal{U}\}$:

1. The argmax in (2.4) is a singleton if and only if the curve $\mathcal{V}(\mathbf{x})$ is strictly convex.
2. If the curve $\mathcal{V}(\mathbf{x})$ is smooth, each $\mathbf{u} \in \mathcal{U}$ is optimal for at most one unit vector \mathbf{n} .

(The latter property becomes “if and only if” for convex velocity profiles.)

In the isotropic case $\mathbf{v}(\mathbf{x}, \mathbf{u}) = v(\mathbf{x})\mathbf{u}$ and the velocity profile $\mathcal{V}(\mathbf{x})$ is just a circle centered at the origin. Thus, the optimal direction of motion is just $\mathbf{u}_*(\mathbf{x}) = -\mathbf{n} = \frac{-\nabla\varphi(\mathbf{x})}{|\nabla\varphi(\mathbf{x})|}$, as noted in the previous section for the Eikonal equation. But as shown in Figure 2.4, this is not the case for general velocity profiles.

The lack of strict convexity for $\mathcal{V}(\mathbf{x})$ implies that, for some \mathbf{n}^{bad} there are at least two different optimal directions of motion; e.g., see Figure 2.4d. Unlike the $\hat{\mathbf{p}} \neq \tilde{\mathbf{p}}$ scenario discussed earlier, this presents a significant problem since this \mathbf{n}^{bad} might be in effect on a large part of Ω .

In subsections 2.3.2-2.3.3 we focus on two sources of anisotropy in pedestrian path planning. The first is to consider the case of two pedestrian crowds (A and B) moving towards different exits, with their speed depending on the angle between the directions of motion of the two crowds (\mathbf{u}_A and \mathbf{u}_B). Examples of such models can be found in [75, 124, 116, 74, 76, 61]. Another source of anisotropy is to have the velocity of the pedestrians depend on the density of the crowd in a non-local, non-radially symmetric way, as in [41]. Before examining the convexity and smoothness of the corresponding velocity profiles, we provide a brief overview of numerical approaches for the system (2.5).

2.3.1 Notes on efficiency in numerical implementation

If ρ is known and $t > 0$ is fixed, then φ is the viscosity solution of the Hamilton-Jacobi-Bellman equation (2.5b), and \mathbf{u}_* can be recovered from (2.5c). In the special case of isotropic dynamics (i.e. when $\mathbf{v}(\rho(\mathbf{x}, t), \mathbf{u}) = v(\rho(\mathbf{x}, t))\mathbf{u}$ and $\mathbf{u} \in S^1$), this HJB PDE reduces to an Eikonal equation (2.6), for which many competing efficient algorithms have been developed in the last 20 years. Dijkstra-like (“Fast Marching”) methods [120, 109, 112] provide non-iterative solvers by dynamically recovering the correct ordering of discretized equations and solving them one at a time. These methods have been extended to meshes on manifolds [80, 108], higher-order accurate Eulerian [110, 1] and semi-Lagrangian discretizations [40], and more general anisotropic cases [111, 4, 92]. An alternative (“Fast Sweeping”) approach is based on alternating through a set of natural geometric orderings to speed up the convergence of Gauss-Seidel iterations [19, 127, 119, 78, 90, 125, 126, 17]. Each of these two approaches is particularly efficient on its own subset of problems. More recently, hybrid (two-scale) methods have been introduced in [31, 32] to combine the advantages of both marching and sweeping. These papers also include a review of several other “fast Eikonal solvers” (mirroring the logic of label-correcting algorithms on graphs) and their parallel implementations.

In principle, the computational cost of solving (2.5b) in each time-slice can be further reduced by only computing φ on the relevant subset of $\bar{\Omega}$. Such “causal domain restriction” techniques are reminiscent of the classical A* algorithm on graphs [62] and have been also extended to Eikonal equations; see [35] and references therein. In pedestrian flow modeling, similar ideas have been used to speed up the “dynamic floor field” computation in [63].

For the general/anisotropic version of the pedestrian direction field problem, it usually is not possible to write down the analytic expression for \mathbf{u}_* even if $\nabla\varphi$ is already known. The HJB equation (2.5b) is then most naturally discretized in a semi-Lagrangian framework

[51], with \mathbf{u}_* recovered at each gridpoint through a numerical optimization over all $\mathbf{u} \in \mathcal{U}$. Aside from being a significant computational bottleneck, this serves to “mask” some of the possible model inconsistencies: if the argmax is not unique (e.g., when the velocity profile is not strictly convex) this can easily go unnoticed and the simulation results will become dependent on the specific implementation of optimization on \mathcal{U} .

The fully coupled system (2.5) is typically solved by explicit time-stepping: the current density ρ is used to compute φ in the current time slice, and the recovered \mathbf{u}_* is then used to find ρ in the next time-slice via (2.5a, 2.5d). In [68] this approach was used with WENO-based discretizations for both the conservation law and the Eikonal PDE. Our own simpler implementation relies on a Lax-Friedrichs scheme [86] for the hyperbolic conservation law (2.5a) and a first-order accurate semi-Lagrangian discretization of the HJB equation (2.5b). The latter is treated by fast sweeping – primarily because this approach is easier to extend to multi-crowd scenarios discussed in section 2.4.

2.3.2 Inter-crowd anisotropy

Hughes’ paper used isotropic pedestrian speed functions based on the classical Lighthill and Whitham model for vehicular flow[91]; e.g., $v(\rho, \mathbf{u}) = \bar{v}(1 - \rho/\rho_{\max})$, where \bar{v} and ρ_{\max} are positive constants. Other monotone non-increasing functions of ρ have been similarly explored; e.g., the control-theoretic reinterpretation of Hughes’ work in [68] used a version of Drake’s fundamental diagram [48]

$$v(\rho, \mathbf{u}) = \bar{v}e^{-\alpha\rho^2}. \tag{2.7}$$

Hughes also considered multi-crowd scenarios, with each crowd distinguished by its desired destination and a different value of \bar{v} . However, in his models the density induced

slow-downs were due to the *total* density only. E.g., given two crowds A and B , their speeds are specified by $v^c(\rho, \mathbf{u}) = \bar{v}^c (1 - \rho/\rho_{\max})$, where $c \in \{A, B\}$ and $\rho = \rho^A + \rho^B$. This yields a total of four PDEs for $(\rho^A, \varphi^A, \rho^B, \varphi^B)$, with the coupling between the crowds based on ρ only. Importantly, these models are still isotropic and the equations for both φ^A and φ^B are Eikonal; i.e., the differences in the directions chosen by two crowds have no effect on their speeds even when they are passing through each other.

More recent work by Jiang et al. [75] aimed to penalize for such directional differences explicitly by extending Formula (2.7). The same approach was later adopted in [124, 116, 74, 76] and served as a background in [71]. Assuming that \mathbf{u}^A and \mathbf{u}^B are the crowds' respective directions and the angle between them is $\psi = \arccos(\mathbf{u}^A \cdot \mathbf{u}^B)$, the “disagreement penalty” factor is defined as

$$f(\bar{\rho}, \psi) = e^{-\beta(1-\cos\psi)\bar{\rho}^2}, \quad (2.8)$$

where $\bar{\rho}$ is the density of the “other” crowd. With this notation the new crowd speeds are defined as

$$v^A(\rho^A, \rho^B, \mathbf{u}^A, \mathbf{u}^B) = \bar{v} e^{-\alpha(\rho^A + \rho^B)^2} f(\rho^B, \psi) \quad \text{and} \quad v^B(\rho^A, \rho^B, \mathbf{u}^A, \mathbf{u}^B) = \bar{v} e^{-\alpha(\rho^A + \rho^B)^2} f(\rho^A, \psi) \quad (2.9)$$

for motion in the respective crowd directions \mathbf{u}^A and \mathbf{u}^B ; see the “geometric velocity” definition (2.2). The idea is that $f = 1$ when both crowds move in unison, and f monotonically decreases with both $\bar{\rho}$ and ψ , reaching the minimum when crowds move in opposite directions. The values of parameters $\bar{v} = 1.034$, $\alpha = 0.075$, and $\beta = 0.019$ were selected based on the physical experiments described in [123] (crowds of pedestrians asked to pass through each other at oblique angles with their aggregate speeds determined by video cameras).

The dependence on ψ clearly introduces a game-theoretic aspect: both crowds choose their directions simultaneously and their decisions affect each other immediately. Significant technical complications resulting from this strong coupling are discussed in section 2.4.1. But

for now we note that even if \mathbf{u}_*^B is already chosen and known, the task of selecting an optimal \mathbf{u}_*^A is inherently anisotropic: the value function φ^A satisfies a version of the Hamilton-Jacobi equation (2.4)

$$\max_{\mathbf{u}^A \in \mathcal{U}} \{-\nabla \varphi^A(\mathbf{x}, t) \cdot \mathbf{v}^A(\rho^A(\mathbf{x}, t), \rho^B(\mathbf{x}, t), \mathbf{u}^A, \mathbf{u}_*^B)\} = 1.$$

Figure 2.4 illustrates the anisotropy in velocity profiles of φ^A . In [75] and the more recent extensions [124, 116, 74, 76, 71] this anisotropic effect was ignored partly due to rewriting the above equation in “Eikonal-type” notation and partly due to the interpretation of φ^A as a “potential”, mirroring the original model by Hughes. This led them to take $\frac{-\nabla \varphi^A}{|\nabla \varphi^A|}$ as the “optimal” direction, which in general can be quite different from the correct \mathbf{u}_*^A .

However, it is worth noting that at these parameter values, the problem is very nearly isotropic at lower densities. At $\beta = 0.019$ and $\rho^B = 1$, we found the maximum angular difference between $\frac{-\nabla \varphi^A}{|\nabla \varphi^A|}$ and \mathbf{u}_*^A to be less than 0.02 radians (See Figure 2.4a for a plot of the nearly circular velocity profile). This might explain why this discrepancy was not noticed in the numerical experiments reported in [75, 124, 116, 74, 76, 71].

To highlight how the optimal direction is in fact \mathbf{u}_*^A and not $\frac{-\nabla \varphi^A}{|\nabla \varphi^A|}$, let us consider an example with Crowd *A* consisting of a single pedestrian (i.e., $\rho^A = 0$ everywhere). Let Crowd *B* form a “river” of infinite length moving to the right, with density $\rho^B = 1$ for $0.3 \leq y \leq 0.7$, and $\rho^B = 0$ everywhere else. Then let us consider the path planning of our single pedestrian of Crowd *A* starting at $x = 0.3$ and $y = 0$, trying to reach the upper boundary $y = 1$ as a target. Then $\frac{-\nabla \varphi^A}{|\nabla \varphi^A|}$ always points directly upward. However, the optimal path is actually to move diagonally with crowd *B* while in the “river”. For parameter values of $\alpha = 0.075$ and $\beta = 0.347$, the exit time for the gradient path planning is $T = 1.21$. But with the correct \mathbf{u}_*^A path planning, the exit time is instead $T = 1.1762$; see Figure 2.2. For comparison, the individual’s velocity profile while in the “river” can be found in Figure 2.2b. Here we can see that the largest vertical component of the speed corresponds to moving diagonally in the

orange direction, the direction of the individual’s path through the “river” in Figure 2.2a. In particular, it has a larger vertical component than would result from moving strictly up (see the red arrow in Figure 2.2b).

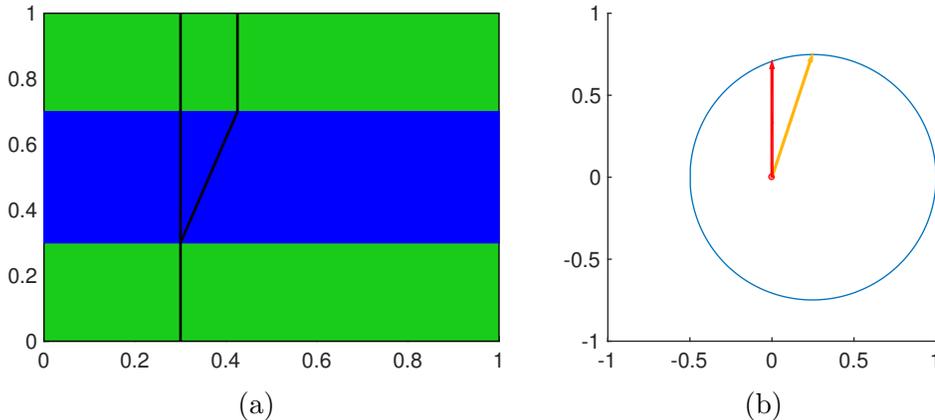


Figure 2.2: Optimal versus gradient path planning in subfigure (a). Corresponding velocity profile while “in the river” in subfigure (b).

A more “organic” example highlighting the difference between optimal and gradient path-planning is included in Figure 2.3. Here two crowds are moving through a domain with a single obstacle, trying to reach the exits on two opposite sides of the domain. The crowds are initially intermixed and their optimal direction fields (shown in blue) are different from the gradient fields of their respective value functions (shown in red). The red directions appear to indicate shorter paths toward the desired exits, but using them would be suboptimal since the corresponding “disagreement penalty” to their speeds is higher.

We note that the parameter values used in [75] and subsequent publications seem highly unrealistic despite the experimental data used for calibration in [123]. For instance, consider two crowds with densities $\rho^A = \rho^B = 1$ person per square meter. For convenience, we will define $\xi = f(1, 0) - f(1, \pi) = 1 - e^{-2\beta}$, which gives the speed reduction resulting from these crowds moving in opposite directions rather than in unison. At $\beta = 0.019$, if these crowds move in opposite directions, they are only slowed down by $\xi \approx 4.7\%$ of the

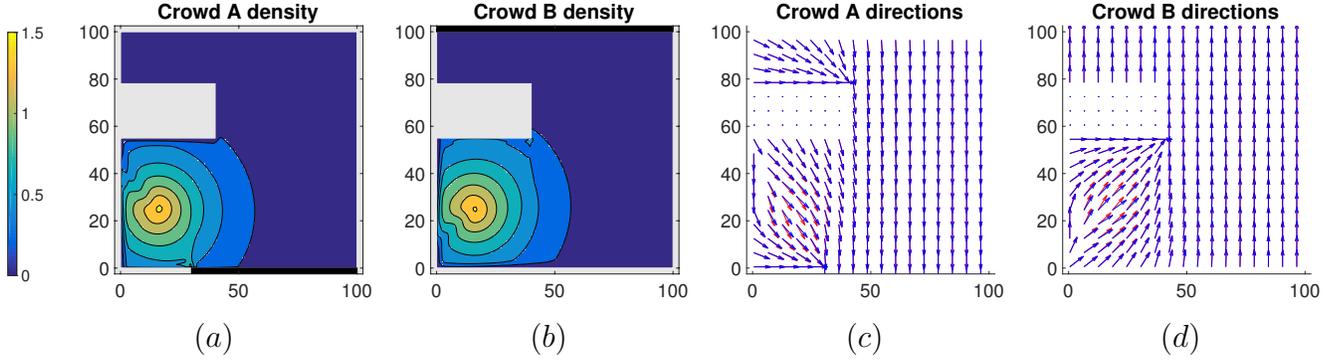


Figure 2.3: Two interacting crowds aiming for different exits on a domain with a single obstacle. Subfigures (a) and (b): contour plots of ρ^A and ρ^B at $t = 0.015$ with the respective exits shown in black. Subfigures (c) and (d): their corresponding optimal direction fields (in blue) and gradient fields (in red). Grid size is 100×100 , timestep is $\Delta t = 0.005$.

full speed achievable when they move in unison. To provide some perspective, the classical estimate by Navin and Wheeler [95] used to calibrate many traffic engineering models yields³ a significantly higher $\beta = 0.078$. For our example, with $\rho^A = \rho^B = 1$, this corresponds to a slow down of $\xi \approx 14.5\%$ relative to moving in unison. Even this last estimate appears rather conservative: anyone who has ever tried to catch a subway train during rush hour would attest that a speed reduction of $\xi \in [30\%, 50\%]$ (corresponding to $\beta \in [0.178, 0.347]$) would be much closer to their experience.

For high enough values of ρ^B or β , the velocity profile of crowd A can actually become non-convex. This scenario is highlighted in Figure 2.4d. If a pedestrian tries to maximize the component of \mathbf{u}^A in the $-x$ direction, then both choices of \mathbf{u}^A denoted by the magenta arrows are optimal.

When looking at the velocity profile \mathcal{V} , we are concerned with only the smoothness and convexity of the profile, and not the scaling. Thus we need only look at the “disagreement penalty” factor $f(\rho^B, \psi)$. Assuming that ρ^B is fixed, we will further refer to it as $f(\psi)$ to

³ Navin and Wheeler [95] have examined the case of two crowds heading in opposite directions with $\rho^A + \rho^B = 1$ and different ratios of $\nu = \rho^A / (\rho^A + \rho^B)$. The case of $\nu = 1$ is equivalent to 2 crowds moving in unison, while $\nu = 1/2$ yields $\Psi = \pi$ and $\rho^A = \rho^B = 1/2$. In the latter case they have observed that the speeds were reduced by 4%.

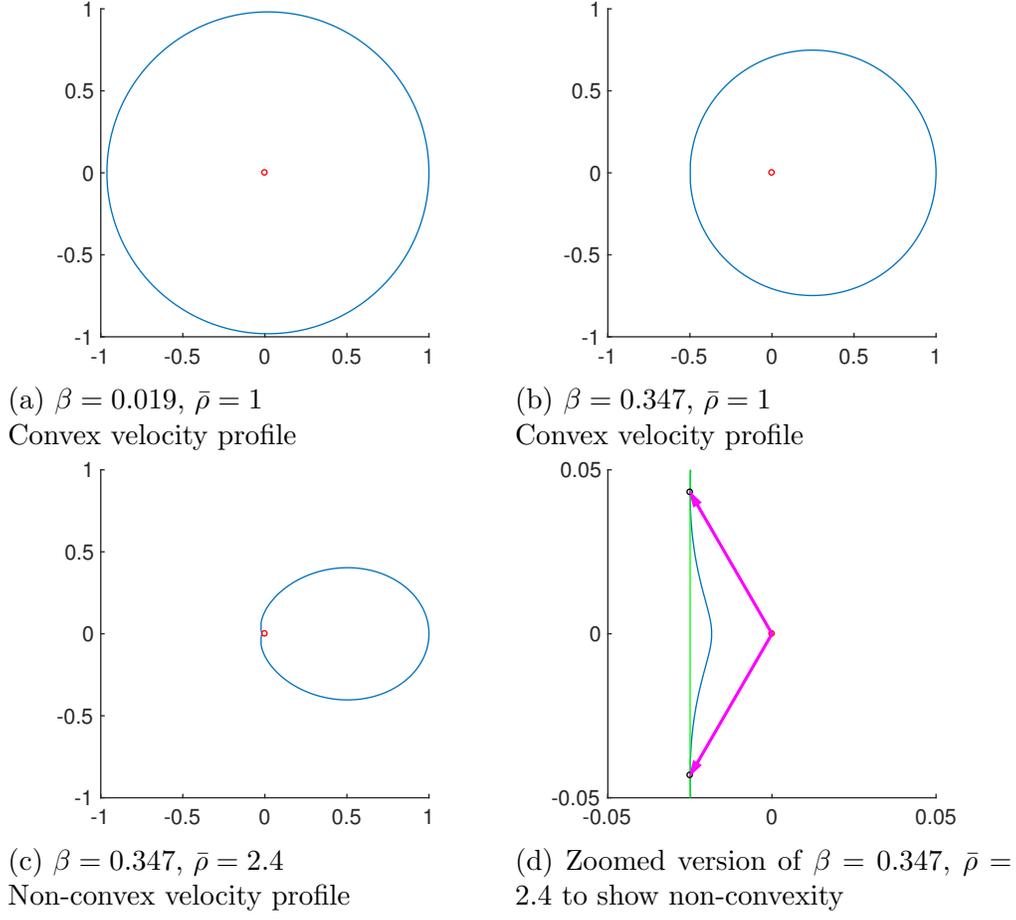


Figure 2.4: Anisotropic velocity profiles for Crowd A at various parameter values. We assume $\mathbf{u}_*^B = (1, 0)$.

simplify the notation. For a fixed \mathbf{x} , we only need to check that the polar curve $\mathbf{f}(\psi) = (f(\psi) \cos(\psi), f(\psi) \sin(\psi))$ is strictly convex, where ψ is the angle between \mathbf{u}^A and \mathbf{u}^B . If we consider its tangent vector $\mathbf{T}(\psi) = \dot{\mathbf{f}}(\psi)$, we can then write the derivative of the tangent vector as $\mathbf{T}'(\psi) = k(\psi)\mathbf{N}(\psi)$, where k is the curvature, and \mathbf{N} is the unit normal vector. The sufficient condition for strict convexity is that $\mathbf{T}(\psi) \times \mathbf{N}(\psi)$ does not change sign. This reduces to a simple condition on f and its derivatives:

$$f(\psi)^2 + 2f'(\psi)^2 - f(\psi)f''(\psi) > 0, \quad \forall \psi \quad (2.10)$$

For the particular model (2.8), this condition simplifies to:

$$1 + \beta\bar{\rho}^2 \cos(\psi) + \beta\bar{\rho}^2 \sin^2(\psi) > 0.$$

We see that the left hand side is minimized when $\psi = \pi$. The sufficient condition for strict convexity then becomes

$$\beta\bar{\rho}^2 < 1.$$

At the parameter values used in [75], we see that we only lose strict convexity at densities exceeding $\bar{\rho} \approx 7.25$ people per square meter – a threshold which is too high to be relevant in practice. However, as discussed previously, their value of β is rather small and with $\beta = 0.078$ (consistent with Navin and Wheeler’s data [95]), we instead get that we can lose strict convexity when the density of either crowd exceeds a more physically relevant threshold of $\bar{\rho} = \sqrt{1/0.078} \approx 3.57$.

We summarize the results of using different values of β in Table 2.1. The first two columns are for β values consistent with Wong et al. [123] and Navin et al. [95] respectively. The third and fourth columns correspond to a slowdown of $\xi = 0.3$ and 0.5 respectively. In each case we list the critical density $\hat{\rho}$ at which the velocity profile loses strict convexity. We note that different “disagreement penalty factors” can be studied similarly. For example, a model considered in [61] is based on using $\bar{\rho}$ rather than $\bar{\rho}^2$; i.e.,

$$f(\bar{\rho}, \psi) = e^{-\beta(1-\cos\psi)\bar{\rho}}, \tag{2.11}$$

with the advantage that it scales more consistently for pairwise interactions of multiple crowds. The critical $\hat{\rho}$ values for this alternative model are reported in the last row of Table 2.1.

Table 2.1: Critical densities for two disagreement penalty models and various β values.

β	0.019	0.078	0.178	0.347
ξ	0.037	0.144	0.3	0.5
critical density $\widehat{\rho}$ for the $\beta\bar{\rho}^2$ model	7.25	3.58	2.37	1.70
critical density $\widehat{\rho}$ for the $\beta\bar{\rho}$ model	52.6	12.8	5.62	2.88

2.3.3 Intra-crowd anisotropy

The models in the previous section dealt only with localized interactions of pedestrians, with anisotropy arising due to the presence of multiple crowds. Another source of anisotropy is through nonlocal interactions, where the velocity $\mathbf{V}(\mathbf{x}, t)$ depends on the values of $\rho(\cdot, t)$ in a region containing \mathbf{x} , rather than on a local $\rho(\mathbf{x}, t)$ only. In the paper by Cristiani, Priuli, and Tosin [41], the velocity is no longer chosen directly. Instead, it is the sum of a chosen behavioral velocity $\mathbf{u} \in S^1$, and an interaction velocity perturbation \mathbf{w}_i that is determined based on the density of pedestrians inside some sensory region $S(\mathbf{x}, \mathbf{u})$. This sensory region is usually modeled by a sector (of angle α and radius R) centered at \mathbf{x} and with the orientation determined by \mathbf{u} ; see Figure 2.5. The key idea is that each pedestrian is only influenced by others roughly ahead of them, and not the ones behind. The model developed in [41] defines the interaction velocity \mathbf{w}_i as follows:

$$\mathbf{w}_i = \int_{S(\mathbf{x}, \mathbf{u}) \cap \Omega} \mathcal{F}(\mathbf{y} - \mathbf{x}) \rho(t, \mathbf{x}) d\mathbf{y}; \quad \mathcal{F}(\mathbf{r}) = \begin{cases} \frac{-F\mathbf{r}}{|\mathbf{r}|^2}, & |\mathbf{r}| > \frac{1}{C}, \\ \frac{-FC\mathbf{r}}{|\mathbf{r}|}, & |\mathbf{r}| \leq \frac{1}{C}, \end{cases} \quad (2.12)$$

where F and C are positive constants, with the latter functioning as a ‘‘cutoff’’. Since the interaction velocity perturbation has a non-local dependence on the current ρ , it would be accurate to denote it $\mathbf{w}_i[\rho(\cdot, t)](\mathbf{x}, \mathbf{u})$, but we will at times abbreviate this as $\mathbf{w}_i(\mathbf{x}, \mathbf{u})$ or even \mathbf{w}_i . The overall velocity is $\mathbf{v}[\rho(\cdot, t)](\mathbf{x}, \mathbf{u}) = \mathbf{u} + \mathbf{w}_i[\rho(\cdot, t)](\mathbf{x}, \mathbf{u})$, which is a version of formula (2.3). Modulo this change from $\mathbf{v}(\rho(\mathbf{x}, t), \mathbf{u})$ to $\mathbf{v}[\rho(\cdot, t)](\mathbf{x}, \mathbf{u})$, the functions ρ and φ satisfy the same system of PDEs (2.5).

We will now investigate the convexity of the velocity profiles for the case of intra-crowd anisotropy. Since this model depends non-locally on the density, we will only analyze this under some assumptions on the density function.

Linear Density.

First, let us consider the case of a density distribution that is linear in \mathbf{x} at some fixed time t . WLOG we may assume that this density distribution is of the form:

$$\rho(x, y) = \rho_0 + \rho_x x$$

In this case, we can evaluate the integral found in (2.12), which is done in Appendix A.

We note that for very high densities, the velocity profile may not actually contain the origin, as the interaction velocity can dominate the behavioral velocity. We show in Appendix B that as long as the density is linear in \mathbf{x} and the velocity profile contains the origin, then we have convexity for all α such that $\alpha > 3 \sin(\alpha)$. This condition is satisfied for all $\alpha > \approx 130.57^\circ$, including the value of $\alpha = 170^\circ$ used by Cristiani et al [41]. Otherwise, it may be possible to find a linear density function, and values for the parameters such that the velocity profile will not remain convex. For example, consider the case of $\alpha = 20^\circ$, $F = 1$, $R = 1$, $\rho_0 = 2$, and $\rho_x = -3/2$. A plot of this velocity profile can be found in Figure 2.6a.

This linear density example is also good for illustrating another (general) pitfall in

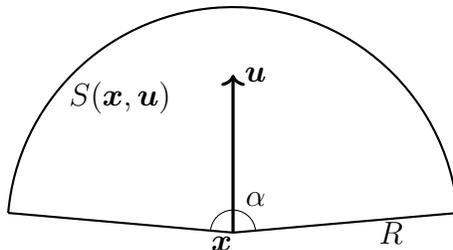


Figure 2.5: An example of sensory region $S(\mathbf{x}, \mathbf{u})$ for intra-crowd anisotropy.

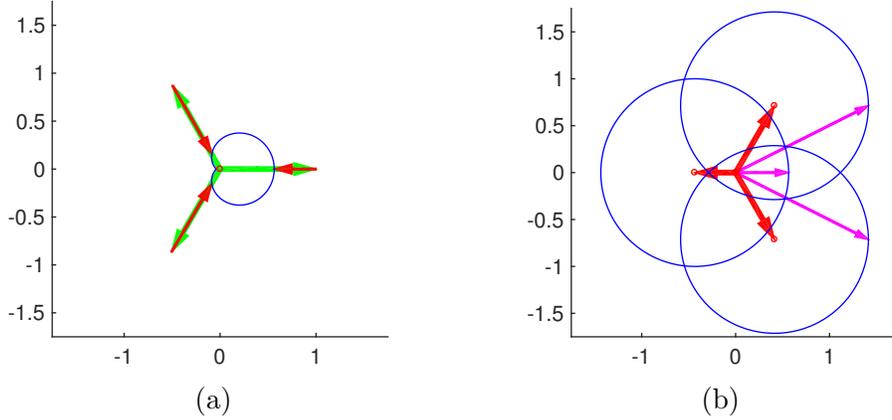


Figure 2.6: Left: Non-convex velocity profile for intra-crowd anisotropy and linear density with $\alpha = 20^\circ$, $F = 1$, $R = 1$, $\rho_0 = 2$, $\rho_x = -3/2$ on the left. Right: three “shifted circle” versions corresponding to different/fixed \mathbf{w}_i choices. Magenta arrows show the resulting velocity corresponding to $\mathbf{u} = (1, 0)$ in each profile. One of them coincides with the correct resulting velocity in subfigure (a).

anisotropic modeling. In [41] the dynamics are given in the form:

$$\mathbf{y}'(t) = \mathbf{u}(t) + \mathbf{w}_i[\rho(\tau, \cdot)](t)$$

However, this misleadingly suggests that \mathbf{w}_i is independent of the choice of \mathbf{u} . To illustrate this point, we show the correctly computed velocity profile in Figure 2.6a with three different versions of \mathbf{u} in green and the corresponding \mathbf{w}_i vectors in red. Choosing any one of these \mathbf{w}_i 's to be used *for all* \mathbf{u} directions will yield a “shifted circle” velocity profile with very different resulting velocities; see Figure 2.6b. Moreover, it is not clear which of these shifts would be a reasonable choice. This limits the validity of the simplified equation proposed in [41]

$$|\nabla\varphi| - \nabla\varphi \cdot \mathbf{w}_i[\rho(\tau, \cdot)] = 1$$

to radially symmetric sensing regions (i.e., with $\alpha = 360^\circ$ only). Based on personal communications with the authors, the actual numerical tests in [41] have used a correct semi-Lagrangian discretization of the general HJB equation (2.5b).

Remark 1. The fact that \mathbf{w}_i is dependent upon the choice of \mathbf{u} also presents a significant computational hurdle in calculating numerical approximations to this model. Since the value of

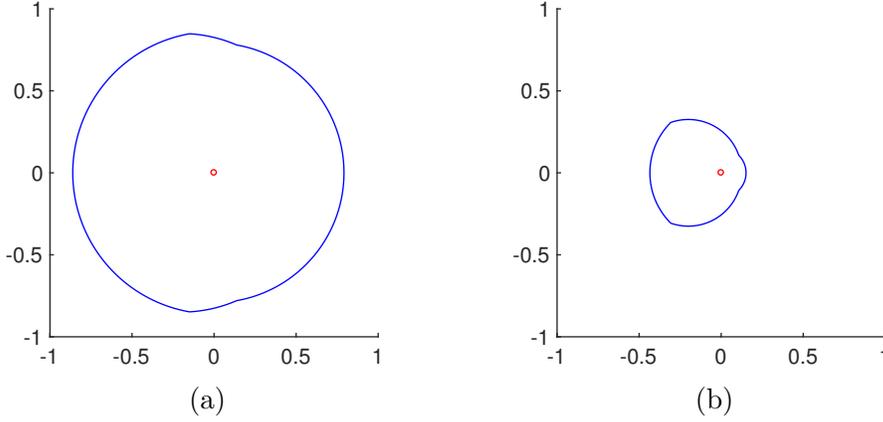


Figure 2.7: Non-convex velocity profiles for discontinuous density: $F = 0.4$, $R = 1$, $\rho_1 = 1.5$, $\rho_2 = 1$, $\alpha = 20^\circ$ in (A) and the same except for $\alpha = 90^\circ$ in (B).

\mathbf{w}_i requires calculating an integral over $S(\mathbf{x}, \mathbf{u})$, this results in frequent numerical integration in order to maximize over all possible choices of \mathbf{u} . However, if the density is approximately linear, our work in Appendix A shows how to evaluate this integral analytically, which could be used to speed up the numerics.

Piecewise Constant Density.

We also considered piecewise constant density functions of the form:

$$\rho(x, y) = \begin{cases} \rho_1, & x > 0 \\ \rho_2, & x \leq 0 \end{cases}$$

Here we find that both convexity and smoothness can also fail at $x = 0$ for some parameter values, see the examples in Figure 2.7.

2.4 Nash equilibria

Returning to the problem of two-crowd interactions with $\rho = (\rho^A, \rho^B)$, let us consider a speed function of the form:

$$v^A(\rho, \mathbf{u}^A) = f(\rho^B, \psi)$$

with the corresponding velocity

$$\mathbf{v}^A(\mathbf{x}, \mathbf{u}^A) = \mathbf{u}^A f(\rho^B(\mathbf{x}), \psi)$$

where \mathbf{u}^A and \mathbf{u}^B are the directions of motion of each crowd and ψ is the angle between them.

In this case, the decision-making of the two crowds is modeled as a non-zero-sum game. We then look for a solution to the following coupled system of Hamilton-Jacobi-Isaacs (HJI) equations:

$$\max_{\mathbf{u}^A \in S^1} \left\{ (\nabla \varphi^A(\mathbf{x}) \cdot (-\mathbf{u}^A) v^A(\rho^A(\mathbf{x}), \mathbf{u}^A, \mathbf{u}^B) \right\} = 1 \quad (2.13)$$

$$\max_{\mathbf{u}^B \in S^1} \left\{ (\nabla \varphi^B(\mathbf{x}) \cdot (-\mathbf{u}^B) v^B(\rho^B(\mathbf{x}), \mathbf{u}^B, \mathbf{u}^A) \right\} = 1 \quad (2.14)$$

This notation is slightly misleading: each of the two PDEs involves maximizing in only one variable (\mathbf{u}^A or \mathbf{u}^B), with the other one participating as a parameter. Both maximizations are assumed to be performed simultaneously, yielding the optimal pair of direction fields $(\mathbf{u}_*^A, \mathbf{u}_*^B)$. A different pair of PDEs could be obtained by assuming that one of the crowds has an advantage of selecting its direction second (once the other crowd's direction is already known). In general, the solution might change depending on which crowd we select to “go first”. This feature is disconcerting from a modeling perspective since the notion of “going first” is harder to motivate when the replanning is done continuously in time (with changing densities ρ^A and ρ^B). So, it is desirable to guarantee that the order of decision making is irrelevant, making the system (2.13, 2.14) meaningful as written above. The sufficient

condition for this is the uniqueness of the two-player *Nash equilibrium* (NE), a concept which we review below for general games.

We assume that player A chooses strategies $a \in \mathcal{A}$ to maximize $J^A(a, b)$ and player B chooses strategies $b \in \mathcal{B}$ to maximize $J^B(a, b)$. In the non-zero-sum setting (when $J^A \neq -J^B$) the well-posedness of the game is non-trivial even when \mathcal{A} and \mathcal{B} are discrete sets. NE is defined as a pair of strategies $(a_*, b_*) \in \mathcal{A} \times \mathcal{B}$ such that there is never an incentive to change unilaterally. More precisely, (a_*, b_*) is said to be NE if both of the following hold:

$$J^A(a_*, b_*) \geq J^A(a, b_*), \quad \forall a \in \mathcal{A},$$

$$J^B(a_*, b_*) \geq J^B(a_*, b), \quad \forall b \in \mathcal{B}.$$

If the game has a unique NE, then we can always assume that both players make their decisions simultaneously – if both players are rational, there is no longer any advantage to “go second.”

The existence of NE for infinite continuous games has been proved under some very general assumptions, at least in the class of mixed policies [57]. For zero-sum-games, the possible non-uniqueness is not a major issue and requires no coordination between the players. If both (a_*, b_*) and $(a_\# , b_\#)$ are NE, then so are $(a_*, b_\#)$ and $(a_\# , b_*)$; moreover, J^A will have the same value for each of these pairs (and $J^B = -J^A$). So, who goes first and which NE they aim for still does not matter. However, none of this holds in the case of non-zero-sum games. In section 2.4.2 we will show examples where the value of multiple NE is different, players disagree about their relative merits, and choosing simultaneously without coordination can easily lead to a non-Nash outcome.

In the case of zero-sum HJI games, the notions of “upper value” and “lower value” are used to capture what happens when one of the players goes first. A separate “Isaacs criterion” on the Hamiltonian ensures that these two values are always equal, enabling simultaneous

decision-making by the players. Additional mild technical conditions guarantee that a NE exists among *pure* strategies, making it unnecessary to consider any relaxed/“chattering” controls [9]. A simple modification of this argument yields the existence of a pure-strategy NE in (2.13, 2.14) provided both velocity profiles are strictly convex. However, to claim that this non-zero-sum system has a well-defined value, we will also need to know that this NE is *unique*.

We will focus on a fixed $\mathbf{x} \in \Omega$, assuming that ρ 's and φ 's are known for both crowds and writing $\mathbf{p} = \nabla\varphi^A(\mathbf{x})$, $\mathbf{q} = \nabla\varphi^B(\mathbf{x})$ to simplify the notation:

$$\max_{\mathbf{u}^A \in S^1} \left\{ \mathbf{p} \cdot (-\mathbf{u}^A) v^A(\rho^B(\mathbf{x}), \mathbf{u}^A, \mathbf{u}^B) \right\} = 1; \quad \max_{\mathbf{u}^B \in S^1} \left\{ \mathbf{q} \cdot (-\mathbf{u}^B) v^B(\rho^A(\mathbf{x}), \mathbf{u}^A, \mathbf{u}^B) \right\} = 1.$$

Furthermore, to focus on the geometric aspects we will use Δ to denote the angle between \mathbf{p} and \mathbf{q} , ψ for the angle between \mathbf{u}^A and \mathbf{u}^B , and (a, b) for the angles that \mathbf{u}^A and \mathbf{u}^B make with $(-\mathbf{p})$ and $(-\mathbf{q})$ respectively.

From the definition (2.9), we note that the anisotropic speeds v^A and v^B have a common isotropic factor which does not affect the choice of optimal direction. Denote $f^A(\psi) = f(\rho^B, \psi)$, $f^B(\psi) = f(\rho^A, \psi)$, recall that $\psi = a - b + \Delta$, and our new task is to show the uniqueness of the NE for all (\mathbf{p}, \mathbf{q}) in the equivalent game

$$\max_{a \in [0, 2\pi]} \left\{ \cos(a) f^A(a - b + \Delta) \right\} = 1, \quad \max_{b \in [0, 2\pi]} \left\{ \cos(b) f^B(a - b + \Delta) \right\} = 1. \quad (2.15)$$

We will assume that the velocity profiles for both crowds are smooth and strictly convex. The strict convexity ensures that the $\arg \max_{a \in [0, 2\pi]}$ is unique for every b , making the “best reply” map $\Phi^A(b)$ well-defined. Since f^A is smooth, the first-order optimality condition

$$0 = \frac{d}{da} \left[\cos(a) f^A(a - b + \Delta) \right] = -\sin(a) f^A(a - b + \Delta) + \cos(a) \frac{\partial}{\partial a} [f^A(a - b + \Delta)] \quad (2.16)$$

should hold for $a = \Phi^A(b)$. If we denote the right hand side as $g(a, b)$, then

$$\frac{\partial g}{\partial a} = -\cos(a) f^A(\psi) - 2 \sin(a) \frac{df^A}{d\psi} + \cos(a) \frac{d^2 f^A}{d\psi^2}; \quad \frac{\partial g}{\partial b} = -\sin(a) \frac{df^A}{d\psi} + \cos(a) \frac{d^2 f^A}{d\psi^2}.$$

Noting that (2.16) implies $\sin(a) = \cos(a) \frac{df^A}{f^A(\psi)}$, we can now simplify the derivative

$$\frac{d\Phi^A}{db} = \frac{-g_b}{g_a} = \frac{-\left(\frac{df^A}{d\psi}\right)^2 + f^A(\psi) \frac{d^2 f^A}{d\psi^2}}{-f^A(\psi)^2 - 2\left(\frac{df^A}{d\psi}\right)^2 + f^A(\psi) \frac{d^2 f^A}{d\psi^2}}. \quad (2.17)$$

A similar derivation can be repeated for the other crowd's best reply $\Phi^B(a)$. If we show that the map $a \mapsto \Phi^A(\Phi^B(a))$ is a contraction mapping, the existence and uniqueness of NE becomes a simple consequence of the Banach fixed-point theorem.

Thus, it is sufficient to show that the derivative is bounded by 1, i.e.

$$\left| \frac{d}{da} \left[\Phi^A(\Phi^B(a)) \right] \right| < 1, \quad \forall a \in [0, 2\pi]$$

It is generally easier to check the slightly stronger condition following from the Chain Rule that

$$\left| \frac{d\Phi^A}{db} \right| \left| \frac{d\Phi^B}{da} \right| < 1, \quad \forall (a, b) \in [0, 2\pi] \times [0, 2\pi]$$

where the derivatives of the best reply maps can be found in Equation (2.17).

2.4.1 NE in intercrowd models

In [75], they assume a speed function for crowd A of the form:

$$v^A(\psi) = v_0 e^{-\alpha((\rho^A)^2 + (\rho^B)^2)} e^{-\beta(\rho^B)^2(1-\cos(\psi))}$$

where ρ is the density of the other crowd. (Crowd B 's speed function is defined analogously.)

Since we are interested only in the shape (rather than scaling) of the velocity profile, we shall just use:

$$f^A(\psi) = e^{-\beta(\rho^B)^2(1-\cos(\psi))}$$

as it is simpler to work with. For ease of notation, we shall also drop the superscripts on ρ .

We calculate that

$$\frac{df^A}{d\psi} = -\beta\rho^2 \sin(\psi) e^{-\beta\rho^2(1-\cos(\psi))},$$

$$\frac{d^2 f^A}{d\psi^2} = \beta^2 \rho^4 \sin(\psi)^2 e^{-\beta \rho^2 (1 - \cos(\psi))} - \beta \rho^2 \cos(\psi) e^{-\beta \rho^2 (1 - \cos(\psi))}.$$

Plugging these into Equation (2.17), we obtain

$$\left| \frac{d\Phi^A}{db} \right| = \left| \frac{\beta \rho^2 \cos(\psi)}{\beta \rho^2 \cos(\psi) + \beta^2 \rho^4 \sin^2(\psi) + 1} \right|. \quad (2.18)$$

We see that the denominator is a quadratic polynomial in $\cos(\psi)$. If we assume the necessary and sufficient condition for strict convexity that $\beta \rho^2 < 1$, then it follows that the absolute value of this polynomial reaches a minimum of $1 - \beta \rho^2$ at $\psi = \pi$. Similarly, the numerator is maximized in absolute value at $\psi = \pi$, so we see that the worst case scenario is that of $\psi = \pi$, where the derivative is bounded by:

$$\left| \frac{d\Phi^A}{db} \right| < \frac{\beta(\rho^B)^2}{1 - \beta(\rho^B)^2}. \quad (2.19)$$

The same process works with the crowds switched. So $\Phi^A(\Phi^B(a))$ is a contraction mapping (and NE is unique) whenever

$$\frac{\beta(\rho^B)^2}{1 - \beta(\rho^B)^2} \frac{\beta(\rho^A)^2}{1 - \beta(\rho^A)^2} < 1 \quad \text{or, equivalently,} \quad \beta(\rho^A)^2 + \beta(\rho^B)^2 < 1. \quad (2.20)$$

The models in [75] assume a value of $\beta = 0.019$, which means that our condition for guaranteeing the uniqueness of NE becomes:

$$(\rho^A)^2 + (\rho^B)^2 < \frac{1}{\beta} \approx 52.632; \quad (2.21)$$

i.e., NE will be unique for all physically relevant crowd densities. However, this criterion becomes much more restrictive for higher (more realistic) values of β . For example, at $\beta = 0.347$, the condition is now $(\rho^A)^2 + (\rho^B)^2 < 2.881$, which is definitely violated when $\rho^A + \rho^B \geq 2.41$ even if $\max(\rho^A, \rho^B) \leq \hat{\rho} \approx 1.7$ and the velocity profiles are convex; see also Table 2.1 and Figure 2.8. Similarly, for the alternative disagreement penalty model based on $(\beta \bar{\rho})$, the corresponding sufficient condition for the uniqueness of NE is $\rho^A + \rho^B < \frac{1}{\beta} = \hat{\rho}$.

For $\beta = 0.347$, this means that the uniqueness might be lost when the total density of both crowds exceeds 2.881.

Our numerical experiments show that the above conditions are actually sharp: all densities from Region 2 in Figure 2.8 seem to yield multiple NE. However, the experiments also show that this only happens when \mathbf{p} and $(-\mathbf{q})$ are sufficiently close; e.g., at $\beta = 0.5$ in the $(\beta\bar{\rho}^2)$ model and $\rho^A = \rho^B = 1.2$, we only observe multiple NE when the angle between \mathbf{p} and \mathbf{q} is greater than 171 degrees.

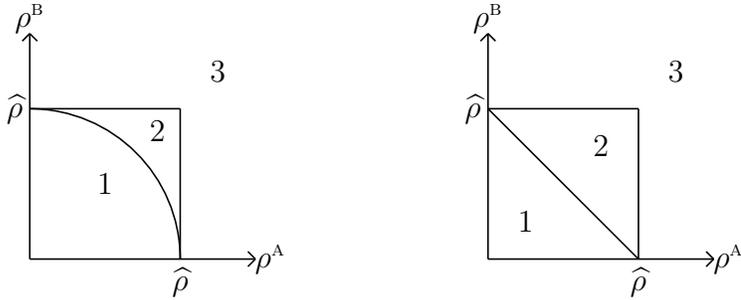


Figure 2.8: Critical density regions. $\beta(\bar{\rho})^2$ model on the left, $\beta\bar{\rho}$ model on the right. Provably unique NE in Region 1; multiple NE observed in Region 2; non-convex velocity profiles in Region 3.

2.4.2 Examples

Example 1

Here we consider an example with $\mathbf{p} = (1, 0)$ and $\mathbf{q} = (-1, 0)$, with $\beta = 0.347$, $\rho^A = 1.68$, and $\rho^B = 0.72$. Numerically, we find three NEs at:

$$\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} 3.1416 \\ 0 \end{pmatrix}, \begin{pmatrix} 3.0366 \\ 0.5208 \end{pmatrix}, \text{ and } \begin{pmatrix} 3.2466 \\ 5.7623 \end{pmatrix}.$$

A plot of $-\mathbf{p}$, $-\mathbf{q}$ and the velocity profiles for each NE can be found in Figure 2.9. The

orange arrow is $-\mathbf{p}$ and the green arrow is $-\mathbf{q}$. The velocity profile and choice of direction for crowd A are plotted in red, while the same is plotted in blue for crowd B .

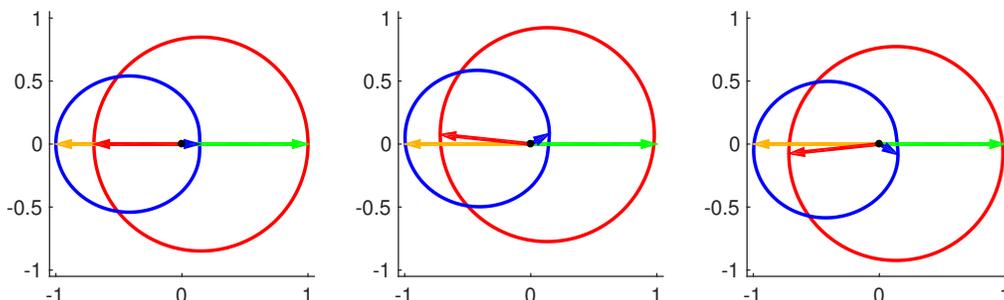


Figure 2.9: $-\mathbf{p}, -\mathbf{q}$, and velocity profiles for each NE in Example 1.

Example 1 is also summarized below in bimatrix form. Both crowds try to maximize their respective outcomes based on (2.15); each entry contains the payoff for the “row player” (crowd A) and the “column player” (crowd B). Of course, both of them have infinitely many options, but we include only those corresponding to Figure 2.9. The NE outcomes are on the diagonal, and their payoffs are different; so, the “value” of this game is not well-defined. Moreover, since the crowds do not coordinate, they cannot know which NE control the other player will choose and this lack of coordination will frequently result in non-Nash (off-diagonal) outcomes.

Nash Equilibria payoffs for Example 1

	0	0.5208	5.7623
3.1416	0.698, 0.141	0.715, 0.139	0.715, 0.139
3.0366	0.695, 0.142	0.718, 0.147	0.705, 0.133
3.2466	0.695, 0.142	0.705, 0.133	0.718, 0.147

Example 2

Now for an example where $\mathbf{p} \neq -\mathbf{q}$. Here we take $\mathbf{p} = (\cos(\pi/20), \sin(\pi/20))$ and $\mathbf{q} = (\cos(39\pi/40), \sin(39\pi/40))$, with $\beta = 0.347$, $\rho^A = 1.68$, and $\rho^B = 1.68$. Numerically, we now find two NEs, this time at:

$$\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} 2.5470 \\ 0.6732 \end{pmatrix}, \text{ and } \begin{pmatrix} 4.0641 \\ 5.4393 \end{pmatrix}.$$

A plot of $-\mathbf{p}, -\mathbf{q}$ and the velocity profiles for each NE can be found in Figure 2.10

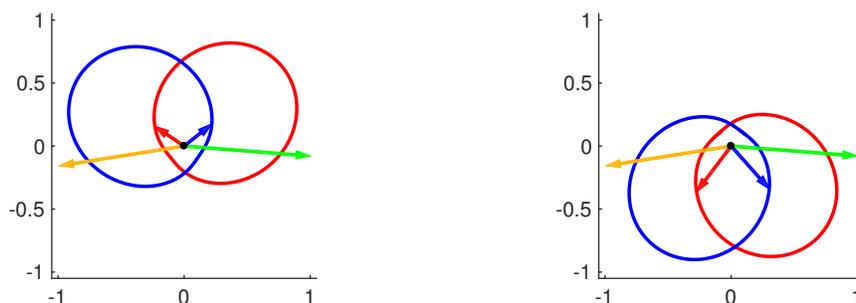


Figure 2.10: $-\mathbf{p}, -\mathbf{q}$, and velocity profiles for each NE in Example 2.

NE payoffs for Example 2

	0.6732	5.4393
2.5470	0.205, 0.205	0.106, 0.105
4.0641	0.105, 0.106	0.328, 0.328

From our numerical experiments, it seems that all cases with $\mathbf{p} \neq -\mathbf{q}$ and multiple NEs are qualitatively similar to Example 2. Since $\mathbf{p} \neq -\mathbf{q}$, we can think of $-\mathbf{p}$ and $-\mathbf{q}$ as splitting the plane into two sections, one where the angle between them is greater than π , and one where the angle is less than π . We always see two NEs, one in the section with angle greater than π , one in the other section. The NE on the side where the angle is less than π then has higher payoffs for both crowds, dominating the other NE.

Example 3

In the previous examples, we showed that multiple NEs are possible even if the velocity profile is strictly convex and smooth. In this last example we aim to illustrate that a strictly convex but non-smooth velocity profile can result in infinitely many NEs. Let us consider a speed function with “disagreement penalty” of the form:

$$f(\bar{\rho}, \psi) = e^{-C\bar{\rho}(\pi^2 - (\text{mod}(\psi, 2\pi) - \pi)^2)}.$$

The resulting velocity profiles form a strictly convex but non-smooth teardrop-like shape as seen in Figure 2.11. In the different subfigures of Figure 2.11, we have consistently chosen $C = 0.1$ and $\rho^A = \rho^B = 1$, keeping \mathbf{p} and \mathbf{q} symmetric relative to the x -axis. We explore how the NEs change as we decrease the angle θ between $(-\mathbf{p})$ and $(-\mathbf{q})$, which are shown by the orange and green arrows respectively.

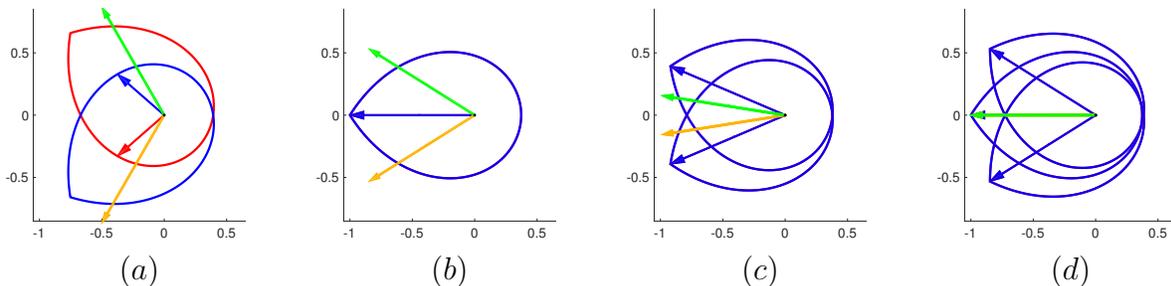


Figure 2.11: Example 3: NEs for non-smooth velocity profiles as $(-\mathbf{p})$ in orange and $(-\mathbf{q})$ in green move closer together. Subfigures (c) and (d) show a few of infinitely many NEs.

Figure 2.11a shows the case $\theta = 2\pi/3$, with a unique NE, both crowds having distinct velocity profiles and different choices of optimal directions (shown in red and blue). As \mathbf{p} and \mathbf{q} move closer together, we still have a unique NE up until the critical angle θ_c , when the velocity profiles of both crowds (and the respective optimal directions) coincide; see Figure 2.11b. A simple geometric argument shows that $\theta_c = \pi/2 - \arctan(1/(2\pi C\bar{\rho}))$. For $\theta \in [0, \theta_c)$, the NE is no longer unique even though in each of them both velocity profiles coincide and

both crowds choose the same “optimal” direction corresponding to the tip of the velocity profile. Note that only some of these NEs are Pareto optimal (the ones where the tip lies in between $(-\mathbf{p})$ and $(-\mathbf{q})$), and many others are not; see the full range of Nash-profiles in Figure 2.11c. It is worth noting that the case $\mathbf{p} = \mathbf{q}$ can also be viewed as one crowd artificially divided into two, whose goals/destinations are actually the same. In order for our model to remain consistent, we would hope that there is a unique NE corresponding to the optimal choice of direction for the isotropic (one-crowd) model. Figure 2.11d shows that this is actually **not** the case and we have infinitely many NEs even with $\theta = 0$. (However, only one of them, corresponding to the vertically symmetric velocity profile and the “isotropically optimal” direction of motion $\frac{-\mathbf{p}}{|\mathbf{p}|}$, is now Pareto-optimal). This suggests that the smoothness of the velocity profile is also a natural consistency condition even for the single crowd models.

2.4.3 Implementation notes for two-crowd models

The previous subsections have discussed uniqueness of Nash Equilibria for the system of coupled Hamilton-Jacobi-Isaacs PDEs (2.13) and (2.14). For numerical simulations, the semi-Lagrangian discretization of these PDEs on the grid results in a pair of coupled optimization problems at each gridpoint (i, j) :

$$\varphi_{ij}^A = \min_{\mathbf{u}_{ij}^A \in S^1} G(\rho_{ij}^A, \rho_{ij}^B, \mathbf{u}_{ij}^B, N\varphi_{ij}^A), \quad (2.22)$$

$$\varphi_{ij}^B = \min_{\mathbf{u}_{ij}^B \in S^1} G(\rho_{ij}^A, \rho_{ij}^B, \mathbf{u}_{ij}^A, N\varphi_{ij}^B), \quad (2.23)$$

where $N\varphi_{ij}^A$ refers to the values of φ^A at the neighboring gridpoints to (i, j) . Ideally, we would like to know that the system (2.22-2.23) has a unique Nash Equilibrium regardless of values of $\rho_{ij}^A, \rho_{ij}^B, N\varphi_{ij}^A$, and $N\varphi_{ij}^B$ whenever the same is true for (2.13-2.14) regardless of the vectors $\nabla\varphi^A$ and $\nabla\varphi^B$. Our numerical experiments confirm this equivalence, but it has not been proven rigorously so far.

These coupled discretized equations have to be solved at every gridpoint. A natural approach for doing this is to mimic the structure of the “best reply” map in Section 2.4. If φ for one of the crowds is assumed to be known, then φ for the other crowd can be solved as in the one-crowd case (e.g., by Fast Sweeping; see Section 2.3.1). So, we can alternate through solving equations (2.22) and (2.23), freezing the current version of φ^A and \mathbf{u}^A to recover φ^B and \mathbf{u}^B , which are then used to update φ^A and \mathbf{u}^A , and so on. When the (discretized) best reply map is a contraction, its fixed point is the unique Nash equilibrium and this iterative process converges to it. Figure 2.12 shows a simulation of two crowds computed in this manner which exhibits qualitative behavior similar to real crowds (e.g. lane formation). If we initialize the discretized best reply map with some initial choice of directions (e.g. from the previous time step), then the convergence of this map is proof that both crowds are in fact using Nash equilibrium strategies. However, this approach is not sufficient to guarantee that this Nash equilibrium is also unique, even if the simulation appears plausible from a phenomenological perspective.

A more careful approach is to iterate over the gridpoints in the outer loop, freezing $(N\varphi_{ij}^A, N\varphi_{ij}^B)$ and updating $(\varphi_{ij}^A, \varphi_{ij}^B)$ simultaneously. We can discretize the control space S^1 (considering a fixed/finite number of directions of motion) and then find the Nash equilibria for the resulting finite bi-matrix game. One advantage is that we can directly verify the number of Nash equilibria, though it might also change as we refine the discretization of S^1 .

Since the coupled system (2.13-2.14) has to be solved in every time-slice, it is desirable to speed up the above procedures as much as possible. One heuristic approach is based on updating the crowd direction fields by measuring the angle Ψ between our **currently** chosen direction and the direction used by the other crowd in the **previous** time-slice. This idea (similar to the approach used in [75] and the subsequent papers) effectively decouples the PDEs, speeding up the numerics and side-stepping all Nash-equilibria issues. However, there

is no proof that the resulting evolution of crowd densities is the same as one would recover from actually solving the coupled system (2.13-2.14).

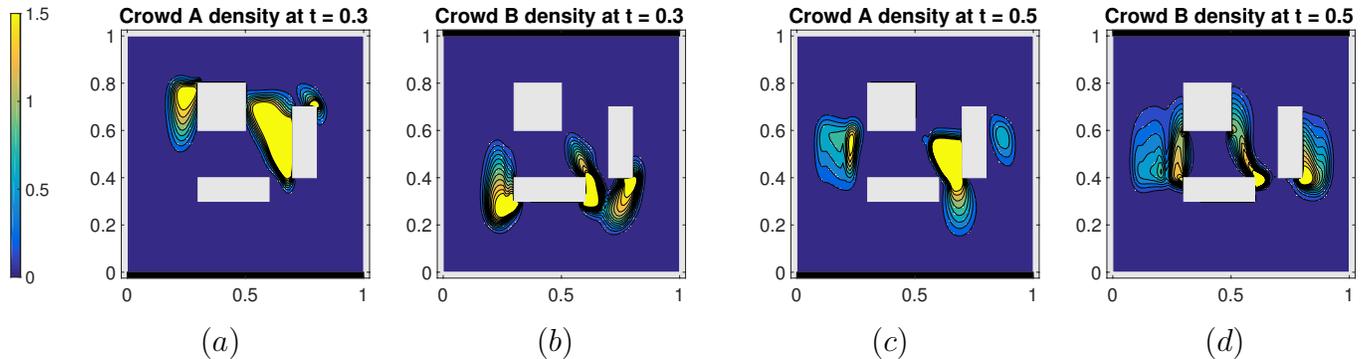


Figure 2.12: A simulation of two crowds, taken at time slices of $t = 0.3$ and $t = 0.5$. Crowd A begins at the top of the domain and is trying to exit at the bottom (shown in black), while Crowd B begins at the bottom of the domain and is trying to exit at the top. Computed on a 100×100 grid with a timestep of $\Delta t = 0.005$. Parameter values are $\alpha = 0.075$, and $\beta = 0.078$.

2.5 Conclusions

We have reviewed several anisotropic crowd flow models, highlighting the geometric interpretation of “optimality” in choosing the pedestrians’ direction field. We have derived internal consistency criteria to guarantee that this direction field is uniquely defined almost everywhere on Ω and showed that the strict convexity and smoothness of the velocity profile are sufficient. Anisotropic interactions of multiple crowds lead to a non-zero-sum-game formulation, and we have developed additional criteria to guarantee the uniqueness of a Nash equilibrium for such games. Up until now these issues were largely ignored in the analysis of some of the more popular models [75, 41], and we showed that they can in principle lead to ill-posedness for a range of physically relevant parameter values. However, we emphasize that our primary goal is not to criticize the prior literature, but to advocate the use of new analytic criteria – either as a pre-processing step or as a run-time “sanity-check” – in future implementations of anisotropic models.

We also proposed an improvement to efficiency of the related numerical methods. For example, the intra-crowd anisotropies may arise due to the lack of radial symmetry in sensing regions [41]. In such cases, the numerical evaluation of the velocity profile becomes costly in itself, and we have showed how this can be done efficiently wherever the crowd density is approximately linear.

We hope that our work will shift the focus of model comparison away from purely phenomenological criteria. E.g., numerical simulations of multiple crowds can produce “lane formation” even if the underlying model is inconsistent or the anisotropic interactions are resolved incorrectly (using gradient descent in u instead of the optimal direction field). However, anisotropic interactions might influence the lane-orientation and are clearly important factors for the behavior of pedestrians in overlapping/interacting “fringes” of such lanes. (E.g., more than 23% of pedestrians in Figure 2.12(c,d) are moving through areas where “the other” crowd’s density is at least 0.3.)

In the future, we would like to investigate several extensions both on the modeling/theoretic side and in numerical techniques. First, the uniqueness of Nash equilibria for general (non-zero-sum) differential games is a challenging question when the players’ dynamics are not separable [23], and we hope that our criteria can be suitably generalized. Second, a semi-Lagrangian discretization of the corresponding system of Hamilton-Jacobi-Isaacs PDEs presents another (discretized) game, and it would be useful to show that similar criteria guarantee the uniqueness of the Nash equilibrium in it. (Otherwise, the results of numerical simulations are rather hard to interpret [25]). Finally, we would like to explore the related issues for Mean Field Games (MFG). Up until now, the MFG-type multi-crowd models have been isotropic (and assumed the presence of stochastic perturbations in pedestrian dynamics, resulting in “viscous” second-order terms in PDEs) [82]. While it is easy to write down the corresponding anisotropic/inviscid equations, we expect them to be sim-

ilarly affected by questions of velocity profile convexity and uniqueness of Nash equilibria. Moreover, the standard MFG framework assumes that each agent's cost/dynamics are only affected by her own state and choice of actions + the aggregate state of all other agents [83, 84, 69, 59]. In contrast, anisotropic multi-crowd models must also account for the direct dependence on other agents' current actions, and it is not obvious whether the standard MFG framework will remain suitable.

3.1 Introduction

While optimal trajectory-planning is a common task in robotics, the notion of “optimality” can be based on many different criteria, such as minimizing time taken to reach the destination, energy required to traverse the path, or minimizing exposure to some sort of threat. In this paper, we focus on the latter in the context of Surveillance-Evasion (SE) games.

We model this adversarial trajectory-planning problem as a semi-infinite zero-sum game between two robotic players: an Observer (O) and an Evader (E) traveling through a domain with occluding obstacles. O chooses a probability distribution over a finite set of predefined surveillance plans. E chooses a probability distribution over an infinite set of trajectories that bring it to a target location before a specified deadline. O’s position and orientation define a time-dependent *pointwise observability* function. E’s goal is to minimize its expected *cumulative observability*, while O aims to maximize this same quantity. Our model is continuous in time and space, relying on numerical methods for partial differential equations (PDEs) to plan E’s trajectories via dynamic programming. We also use techniques from multi-objective and convex optimization to find Nash equilibrium policies for both players.

Classical SE games were introduced in the 1970s and assumed that each player is fully aware of the opponent’s state and can react to any changes in real time [87, 88]. In contrast, our version of the game is built on a different information structure, as each player makes their decisions based only on the probabilistic policy chosen by its opponent, rather than on that opponent’s actual current state. In this sense, we are looking for optimal *open loop* controls, with a built-in uncertainty of the opponent’s position. This information structure

arises in applications where logistical considerations force the players to commit to strategies in advance.

This version of SE games was recently considered in [56] for a model that was rather simplified from the robotics point of view: it lacked any detailed kinematics for E, allowed a set of stationary locations (rather than general surveillance plans) for O, and used the pointwise observability based on omnidirectional sensors at O’s location(s). Here we use the same algorithms for finding Nash equilibrium policies, but extend the approach introduced in [56] to more realistic settings. While we still use a simplified/isotropic model for E’s dynamics, our observer might be non-stationary (choosing among patrol trajectories) and our sensor might be only effective in an angular sector relative to O’s current heading.

In Section 3.2, we review a PDE approach to E’s deterministic trajectory-planning, assuming the pointwise observability is fixed or predetermined. Section 3.3 describes the model and algorithms for SE games, taking adversarial planning into account. Section 3.4 focuses on numerical methods, while Section 3.5 examines the Nash equilibrium policies for several test problems. The limitations of our approach and directions for future work are discussed in Section 3.6.

3.2 Evasive Path-Planning

We begin by considering a standard optimal control problem for the Evader under the assumption that the pointwise observability is fixed and fully known.

E starts at some position $\mathbf{x} \in \Omega$ at the time t , moves with a location-dependent speed $f(\mathbf{x}) > 0$, and can instantaneously change its direction of motion. Its dynamics are given

by:

$$\mathbf{y}'(s) = f(\mathbf{y}(s))\mathbf{a}(s), \quad \mathbf{y}(t) = \mathbf{x}, \quad (3.1)$$

where $\mathbf{a} : \mathbb{R} \rightarrow S^1$ is a measurable *control function* – E's chosen (time-dependent) direction of motion. Its path is constrained to stay within $\Omega \subset \mathbb{R}^2$ (e.g., avoiding any obstacles), and must reach the target \mathbf{x}_T by the time T .

Suppose that the pointwise observability function $K(\mathbf{x}, s)$ is given (i.e., the observer's strategy is fixed). Let $T_{\mathbf{a}} = \min\{s \geq 0 \mid \mathbf{y}(s) = \mathbf{x}_T\}$ be the time it takes the Evader to reach its target using control $\mathbf{a}(\cdot)$. E's goal is to minimize its cumulative observability:

$$\mathcal{J}(\mathbf{x}, t, \mathbf{a}(\cdot)) = \begin{cases} \int_t^{T_{\mathbf{a}}} K(\mathbf{y}(s), s) ds, & T_{\mathbf{a}} \leq T \\ +\infty, & \text{otherwise} \end{cases} \quad (3.2)$$

The value function $u(\mathbf{x}, t)$ is then defined by

$$u(\mathbf{x}, t) = \inf_{\mathbf{a}(\cdot)} \mathcal{J}(\mathbf{x}, t, \mathbf{a}(\cdot)). \quad (3.3)$$

Standard arguments in optimal control theory (e.g., [9, Chapters 3,4]) show that u is the domain-constrained viscosity solution of a time-dependent Hamilton-Jacobi-Bellman (HJB) equation:

$$\begin{aligned} \frac{\partial u}{\partial t} + \min_{|\mathbf{a}|=1} \{f(\mathbf{x})\nabla u(\mathbf{x}, t) \cdot \mathbf{a} + K(\mathbf{x}, t)\} &= 0; \\ u(\mathbf{x}, T) &= +\infty, \quad \forall \mathbf{x} \neq \mathbf{x}_T; \\ u(\mathbf{x}_T, t) &= 0, \quad \forall t \leq T. \end{aligned} \quad (3.4)$$

with an additional condition at $\partial\Omega \setminus \{\mathbf{x}_T\}$ that the minimum is taken over the subset of control values that ensure staying inside $\bar{\Omega}$. Wherever the value function u is smooth, the optimal direction of motion is $\mathbf{a}_* = -\nabla u / |\nabla u|$ and the above equation simplifies to a time-dependent Eikonal PDE

$$\frac{\partial u}{\partial t} - f(\mathbf{x}) |\nabla u(\mathbf{x}, t)| + K(\mathbf{x}, t) = 0. \quad (3.5)$$

In general, (3.5) often does not have a classical solution, but always has a unique viscosity solution, coinciding with the value function of the optimal control problem [9]. The optimal direction of motion \mathbf{a}_* is then uniquely determined almost everywhere on a set $\{(\mathbf{x}, t) \mid u(\mathbf{x}, t) < +\infty\}$ since u is Lipschitz and thus differentiable except on a set of measure zero. However, for those rare starting positions where ∇u is discontinuous, the optimal trajectory is not unique: there can be multiple optimal initial directions \mathbf{a}_* , each of them producing a different optimal trajectory to \mathbf{x}_T . The numerical methods for solving this PDE and tracing the optimal trajectories will be covered in section 3.4.

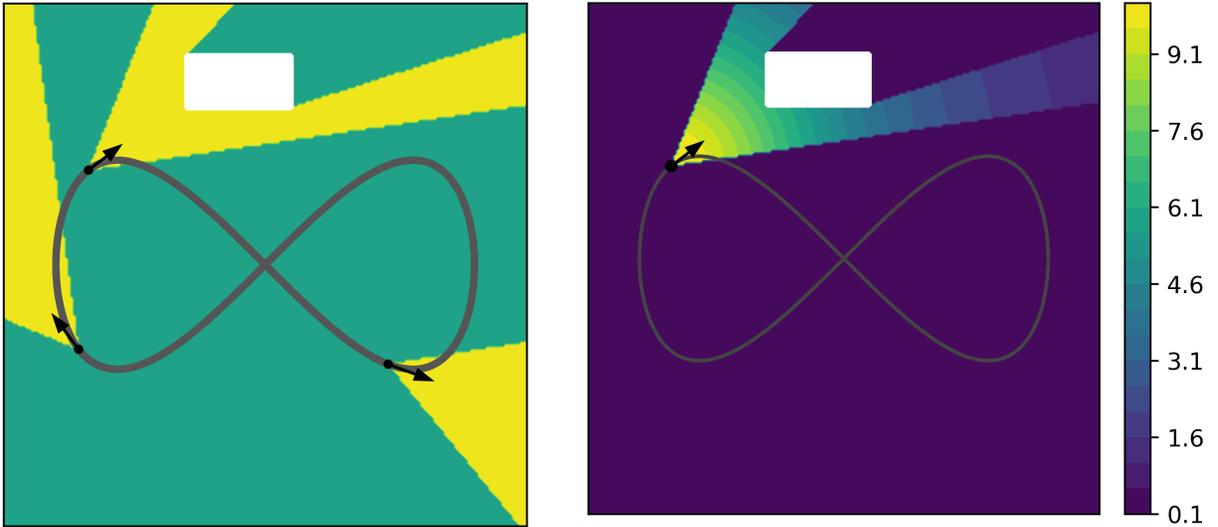


Figure 3.1: LEFT: Visibility field (yellow) for 3 different Observer positions along a patrol trajectory (gray) in a domain with an occluding obstacle (white). RIGHT: The contour plot of the pointwise observability $K(\mathbf{x}, t_1)$, corresponding to O's position $\mathbf{z}(t_1)$.

Before switching to a game-theoretic version of the problem, we remark on the definition of pointwise observability K . Given O's current position $\mathbf{z}(t)$, one can define its *visibility field* $\mathcal{V}(t)$ restricted by obstacles and distance/angle limitations of the sensor. Throughout this paper, we assume that all obstacles are fully occluding and the sensors are only effective in an angular sector of width α centered on observer's current heading $\mathbf{h}(t)$. I.e., $\mathbf{x} \in \mathcal{V}(t)$ if the line segment $(\mathbf{z}(t), \mathbf{x})$ stays in Ω without intersecting any obstacles and the observation angle

(i.e., the angle that $(\mathbf{z}(t), \mathbf{x})$ makes with $\mathbf{h}(t)$) is at most $\alpha/2$. Within $\mathcal{V}(t)$, the pointwise observability should be a decreasing function of the distance $|\mathbf{x} - \mathbf{z}(t)|$. Assuming $\mathbf{z}(t)$ is known, we use

$$K(\mathbf{x}, t) = \begin{cases} \frac{K_0}{|\mathbf{x} - \mathbf{z}(t)|^2 + 0.1} + \sigma, & \mathbf{x} \in \mathcal{V}(t); \\ \sigma, & \mathbf{x} \notin \mathcal{V}(t); \end{cases} \quad (3.6)$$

where K_0 and σ are positive constants. This sector-restricted observability is illustrated in Fig. 3.1.

3.3 Surveillance-Evasion Games

Here we describe the zero-sum game between O and E, following the general framework developed in [56]. We assume that

- O can use any patrol trajectory from the set $\mathcal{Z} = \{\mathbf{z}_1(t), \dots, \mathbf{z}_r(t)\}$ known to both players;
- E can use any of the infinitely many admissible trajectories that lead from \mathbf{x}_s to \mathbf{x}_T by the time T while staying in Ω and avoiding the obstacles.

The payoff of our game is the (expected) cumulative observability, with E as a minimizer and O as a maximizer. Both players are assumed to make their decisions ahead of time and cannot change their mind based on any information gained while traveling through Ω .

3.3.1 The “Evader-goes-second” problem

If O chooses the i -th patrol path, we can simply use \mathbf{z}_i instead of \mathbf{z} in (3.6) to define the corresponding pointwise observability K_i , cumulative observability \mathcal{J}_i , and value function u_i .

The value of $u_i(\mathbf{x}_s, 0)$ yields the cumulative observability along the \mathbf{z}_i -optimal trajectory (i.e., E's best response to O's choice of \mathbf{z}_i .) In Fig. 3.2 we show two circular patrol trajectories (traversed by O with the same angular velocity) and E's optimal trajectory in response to each of them. If O were forced to make its decision deterministically before E, it would simply choose whichever \mathbf{z}_i yields the highest $u_i(\mathbf{x}_s, 0)$.

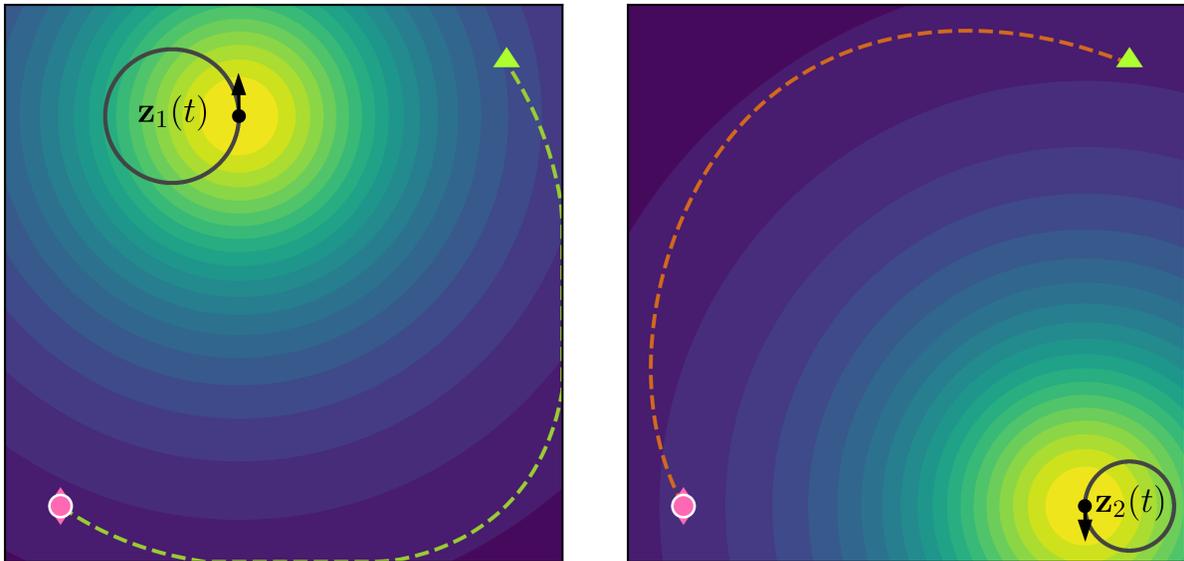


Figure 3.2: E's optimal responses (shown in green and orange) corresponding to O's respective patrol trajectories. The sensor is omnidirectional (i.e., $\alpha = 2\pi$). The pointwise observability is time-dependent, and we show the contour plots for its *initial* version (i.e., $K_i(\mathbf{x}, 0)$) in both cases.

If O prefers to use a probability distribution $\lambda = (\lambda_1, \dots, \lambda_r)$ over the set \mathcal{Z} instead of choosing a specific \mathbf{z}_i , we can similarly define the expected pointwise observability $K^\lambda = \sum_{i=1}^r \lambda_i K_i$ and the corresponding expected cumulative observability $\mathcal{J}^\lambda = \sum_{i=1}^r \lambda_i \mathcal{J}_i$. To find E's best response, the corresponding value function u^λ can be computed by solving (3.5) with K^λ used in place of K . We note that u^λ is *not* a linear combination of u_i 's since this PDE is nonlinear.

This approach is also related to methods for *multi-objective* optimal control. It is generally impossible to minimize the observability with respect to all patrol trajectories simul-

taneously. Instead, one can consider the task of finding *Pareto-optimal* controls. A control $\tilde{\mathbf{a}}(\cdot)$ dominates $\mathbf{a}(\cdot)$ if $\mathcal{J}_i(\mathbf{x}_s, 0, \tilde{\mathbf{a}}(\cdot)) \leq \mathcal{J}_i(\mathbf{x}_s, 0, \mathbf{a}(\cdot))$ for all i , with the inequality strict for at least one i . We will say that $\mathbf{a}(\cdot)$ is Pareto-optimal if it is not dominated by any other control. Plotting the point $(\mathcal{J}_1, \dots, \mathcal{J}_r)$ for each Pareto-optimal control we obtain a Pareto Front (PF), illustrated in Fig. 3.3 for two examples from section 3.5. Each vector λ is normal to a support hyperplane of PF (on which $\mathcal{J}^\lambda = u^\lambda(\mathbf{x}_s, 0)$). If all the elements of λ are positive, it is easy to show that any λ -optimal control is also Pareto-optimal. This is the basis of a *weighted-sum scalarization* approach to general multi-objective trajectory planning introduced in [93]. Unfortunately, it approximates only convex parts of PF [42]. Moreover, it requires imposing a fine grid on the space of λ 's and solving the PDE for each λ -gridpoint. Alternative methods can recover the entire PF [81, 36], but are more computationally expensive. Luckily, for SE games we only need the points on PF corresponding to the players' *mutually* optimal policies; below we explain how this can be done through scalarization with a much smaller number of PDE solves.

O's goal is to find a λ maximizing

$$G(\lambda) = \min_{\mathbf{a}(\cdot)} \sum_{i=1}^r \lambda_i \mathcal{J}_i(\mathbf{x}_s, 0, \mathbf{a}(\cdot)) = u^\lambda(\mathbf{x}_s, 0).$$

Since G is a pointwise minimum of functions linear in λ , we know that $G(\lambda)$ is concave. Thus, O's optimal pdf λ_* can be found by standard methods of convex optimization. Our implementation relies on the projected supergradient ascent method [14, Chap. 8] to maximize G on the r -dimensional simplex of probability densities. During the iterative improvement of λ , we use gradient descent in u^λ starting from $(\mathbf{x}_s, 0)$ to find E's λ -optimal trajectory. Integrating each of K_i 's along that trajectory we obtain the supergradient of $G(\lambda)$, which is then used to update the λ for the next iteration. See Algorithm 4.2 in [56].

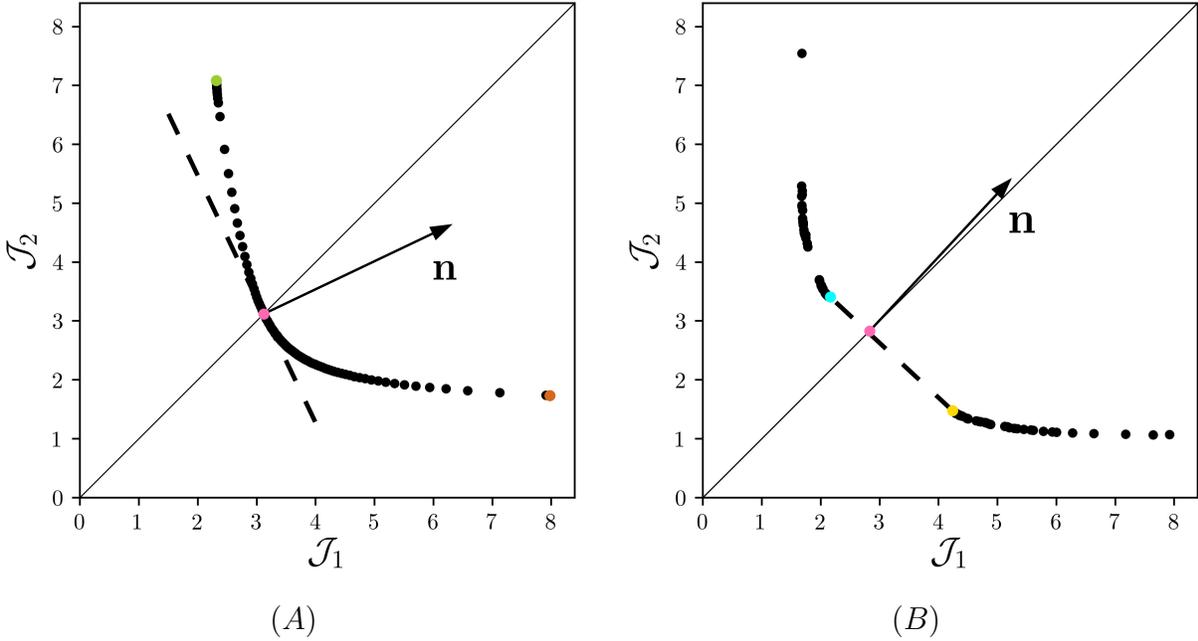


Figure 3.3: The convex part of Pareto Fronts (PFs) corresponding to two examples from section 3.5. PF approximations generated by using 101 λ values, solving for u^λ , and computing $(\mathcal{J}_1, \mathcal{J}_2)$ for λ -optimal trajectories. The dashed line shows a support hyperplane to PF, with \mathbf{n} showing a scaled version of λ_* . Evader’s optimal policy (shown in magenta) corresponds to the intersection of that hyperplane with a “central ray” $\mathcal{J}_1 = \mathcal{J}_2$. LEFT: Example 1 has only one λ_* -optimal trajectory and E’s optimal policy is deterministic; see Fig. 3.5. \mathcal{J}_1 -optimal and \mathcal{J}_2 -optimal trajectories are also represented on PF by the green and orange markers respectively. RIGHT: In Example 2, E’s optimal policy assigns non-zero probabilities to both λ_* -optimal trajectories (cyan and yellow); see Fig. 3.6.

3.3.2 Nash Equilibria

It is restrictive to assume that E is always reacting to O’s decisions. To get a more general interpretation of the game’s value, we need to search for a *Nash equilibrium*. E can also define its policy probabilistically, by choosing a probability distribution θ over the infinite set of admissible controls. Given the players’ policies, the expected payoff is

$$P(\lambda, \theta) = \mathbb{E}_{\lambda, \theta} \left[\mathcal{J}(\mathbf{x}_s, 0, \mathbf{a}(\cdot)) \right] = \sum_{i=1}^r \lambda_i \mathbb{E}_\theta \left[\mathcal{J}_i(\mathbf{x}_s, 0, \mathbf{a}(\cdot)) \right],$$

where $\mathbf{a}(\cdot)$ is a random admissible control chosen according to θ . A pair of policies (λ_*, θ_*) forms a Nash equilibrium if neither player can improve the payoff by changing its policy

unilaterally. I.e.,

$$P(\lambda, \theta_*) \leq P(\lambda_*, \theta_*) \leq P(\lambda_*, \theta), \quad \forall \lambda, \theta.$$

There always exists at least one Nash equilibrium in the set of mixed/probabilistic policies. Moreover, in zero-sum two-player games, the payoff is the same for all Nash equilibria and is thus used to define the *game's value*; e.g., see [97]. Thus, when each player uses a Nash policy, it no longer matters which of them “goes first.”

By the min-max theorem, the value of this game can be found by computing

$$P(\lambda_*, \theta_*) = \min_{\theta} \max_{\lambda} P(\lambda, \theta) = \max_{\lambda} \min_{\theta} P(\lambda, \theta). \quad (3.7)$$

In finite games, this is typically accomplished by linear programming [97], but since E has infinitely many possible paths, our game is *semi-infinite* [117, 101] and this approach is inapplicable. Luckily, it is easy to see that the last expression in (3.7) is equal to $\max_{\lambda} G(\lambda)$ and the previous subsection explains how to find the maximizer λ_* efficiently.

Once the algorithm converges to O's optimal λ_* , we need to find the second half of the Nash equilibrium: E's optimal probability distribution θ_* over the set of admissible trajectories. We accomplish this using the following properties, proven for a stationary Observer in [56], which similarly hold for the time-dependent case considered here. Suppose (λ_*, θ_*) is a Nash equilibrium, $\mathcal{I} = \{i | \lambda_{*,i} > 0\}$, and \mathcal{A} is the set of all controls that have positive probability under θ_* .

1. If λ_* maximizes $G(\lambda)$, then there always exists a probability distribution θ_* over Evader's Pareto-optimal trajectories such that (λ_*, θ_*) is a Nash equilibrium and the set \mathcal{A} has at most $|\mathcal{I}| \leq r$ elements.
2. If $\mathbf{a}(\cdot) \in \mathcal{A}$ then $\mathbf{a}(\cdot)$ is λ_* -optimal.

$$\text{I.e., } \sum_{i=1}^r \lambda_i \mathcal{J}_i(\mathbf{x}_s, 0, \mathbf{a}(\cdot)) = u^{\lambda_*}(\mathbf{x}_s, 0).$$

3. If $i \in \mathcal{I}$, then $\mathbb{E}_{\theta_*} [\mathcal{J}_i(\mathbf{x}_s, 0, \mathbf{a}(\cdot))] = u^{\lambda_*}(\mathbf{x}_s, 0)$.

For all starting positions where u^{λ_*} is differentiable, the optimal direction of motion is $-\nabla u^{\lambda_*}/|\nabla u^{\lambda_*}|$ and the optimal control is unique. Thus, it is natural to expect that, *generically*, there should be only one λ_* -optimal control $\mathbf{a}_*(\cdot)$ leading from \mathbf{x}_s to \mathbf{x}_T by the deadline T . This would imply that $\mathcal{A} = \{\mathbf{a}_*(\cdot)\}$; i.e., E’s half of the Nash Equilibrium is pure/deterministic and $\mathcal{J}_i(\mathbf{x}_s, 0, \mathbf{a}_*(\cdot)) = G(\lambda_*) = u^{\lambda_*}(\mathbf{x}_s, 0)$ for all $i \in \mathcal{I}$. (We say that $\mathbf{a}_*(\cdot)$ is the intersection of the Pareto Front and the “central ray” in the $|\mathcal{I}|$ -dimensional cost space.) This situation, illustrated in Fig. 3.3(A) is indeed quite common. But surprisingly, it is not generic; i.e., for many problems \mathcal{A} must include more than one λ_* -optimal control. This is a counter-intuitive side-effect of changing λ to maximize G : K^λ keeps changing until $(\mathbf{x}_s, 0)$ falls onto a “shockline”, a set of discontinuities of ∇u^{λ_*} . A numerically implemented gradient descent in u^{λ_*} usually yields only one λ_* -optimal trajectory. Approximating *all* of them is much harder. We accomplish this by perturbing λ_* in various directions and computing the resulting unique optimal trajectories corresponding to the perturbed $(\lambda_* + \delta\lambda)$. We refer readers to Alg. 4.3 in [56] for a detailed discussion.

3.4 Numerical Implementation

3.4.1 Numerics for HJB equation

We numerically solve (3.5) using a time-explicit first-order upwind finite-difference scheme on a 9-point stencil. Let $\mathbf{x}_{i,j} = (ih, jh)$, $t_k = k\Delta t$, and $U_{i,j}^k \approx u(\mathbf{x}_{i,j}, t_k)$. We will solve the equation backwards in time, computing $U_{i,j}^{k-1}$ using the values of U at time slice k . More specifically, we will compute an update from each of the eight triangular simplices generated by $\mathbf{x}_{i,j}$ and its eight nearest neighbors (see Fig. 3.4).

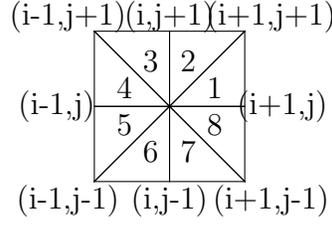


Figure 3.4: Nine-point stencil used to discretize HJB equation at $(\mathbf{x}_{i,j}, t_k)$.

We first compute the first-order approximation $(U_x, U_y) \approx \nabla u(\mathbf{x}_{i,j}, t_k)$ based on a specific simplex. We then check the *upwinding condition*, requiring the approximate gradient to point from that same simplex. E.g., for simplex 1

$$U_x = \frac{U_{i+1,j}^k - U_{i,j}^k}{h}, \quad U_y = \frac{U_{i+1,j+1}^k - U_{i+1,j}^k}{h},$$

and the corresponding upwinding condition is $U_x \leq U_y \leq 0$.

Focusing on a single simplex $l \in \{1, \dots, 8\}$ with vertices $\mathbf{x}_{i,j}$, $\mathbf{x}^{l,1}$, and $\mathbf{x}^{l,2}$, the update from that simplex is

$$U_l = U_{i,j}^k - \Delta t f(\mathbf{x}_{i,j}) D_{i,j}^k + \Delta t K(\mathbf{x}_{i,j}, t_k),$$

where $D_{i,j}^k$ approximates $|\nabla u(\mathbf{x}_{i,j}, t_k)|$. If the upwinding condition is satisfied, we use $D_{i,j}^k = \sqrt{U_x^2 + U_y^2}$. Otherwise, we resort to a “one-sided” semi-Lagrangian update, with

$$D_{i,j}^k = \max \left(\frac{U_{i,j}^k - U(\mathbf{x}^{l,1}, t_k)}{|\mathbf{x}_{i,j} - \mathbf{x}^{l,1}|}, \frac{U_{i,j}^k - U(\mathbf{x}^{l,2}, t_k)}{|\mathbf{x}_{i,j} - \mathbf{x}^{l,2}|}, 0 \right).$$

We then set $U_{i,j}^{k-1}$ to be the smallest of simplex-specific updates $\{U_l\}$. The boundary conditions and terminal conditions are enforced by setting

$$U_{i,j}^k = \begin{cases} 0, & \mathbf{x}_{i,j} = \mathbf{x}_T \text{ and } t_k \leq T; \\ +\infty, & \mathbf{x}_{i,j} \notin \bar{\Omega} \text{ or } (\mathbf{x}_{i,j} \neq \mathbf{x}_T \text{ and } t_k = T). \end{cases}$$

As long as the CFL stability condition $\Delta t \leq h/(\max f(\mathbf{x}_{i,j}))$ is satisfied, this method

is monotone and consistent [51], and therefore converges to the viscosity solution of (3.5) under grid refinement [11].

3.4.2 Time-dependent Visibility

Our definition of $\mathcal{V}_i(t)$ requires computing the set of points that have direct line of sight from $\mathbf{z}_i(t)$. Computational efficiency is very important as this needs to be computed for every time slice. Following [118], we first construct a continuous function $\phi(\mathbf{x})$ such that

$$\phi(\mathbf{x}) \begin{cases} > 0, & \mathbf{x} \in \Omega, \\ = 0, & \mathbf{x} \in \partial\Omega, \\ < 0, & \mathbf{x} \in \mathbb{R}^2 \setminus \bar{\Omega} \quad (\text{e.g., inside obstacles}). \end{cases}$$

Within each time slice t , we numerically solve the quasi-variational inequality

$$\begin{aligned} \max\{\nabla\psi^t(\mathbf{x}) \cdot \mathbf{r}(\mathbf{x}), \psi^t(\mathbf{x}) - \phi(\mathbf{x})\} &= 0, \\ \psi^t(\mathbf{z}(t)) &= \phi(\mathbf{z}(t)), \end{aligned}$$

where $\mathbf{r}(\mathbf{x})$ is a unit vector pointing from $\mathbf{z}(t)$ to \mathbf{x} . Then $\{\psi^t \geq 0\}$ will define the set of points that are not occluded by obstacles. After computing ψ^t , we then enforce any additional distance or angular restrictions to get $\mathcal{V}_i(t)$. Combined with (3.6), this allows us to pre-compute and store $K_i(\mathbf{x}, t)$ for each patrol trajectory $\mathbf{z}_i(t)$ before starting to plan E's optimal responses to various Observer policies λ .

3.4.3 Time-dependent Optimal Path Tracer

Given our numerical solution $U_{i,j}^k$, we now wish to recover the Evader's optimal trajectory $\mathbf{y}(t)$. We approximate the trajectory with a series of points $\mathbf{y}^m \approx \mathbf{y}(m\Delta t)$, where Δt is the

timestep used to numerically solve (3.5).

E’s initial position is $\mathbf{y}^0 = \mathbf{y}(0) = \mathbf{x}_s$. The rest of the path can be found in a semi-Lagrangian manner by computing

$$\mathbf{a}_*^m = \arg \min_{|\mathbf{a}| \leq 1} \{ \tilde{U}(\mathbf{y}^m + \Delta t f(\mathbf{y}^m) \mathbf{a}, m \Delta t) \},$$

$$\mathbf{y}^{m+1} = \mathbf{y}^m + \Delta t \mathbf{a}_*^m.$$

where \tilde{U} is the function obtained from the grid values of U by trilinear interpolation. We terminate the path once we get close enough to the target so that $|\mathbf{y}_m - \mathbf{x}_T| \leq f(\mathbf{y}_m) \Delta t$. In our numerical experiments, we optimize over $\{|\mathbf{a}| = 1\} \cup \{(0, 0)\}$ instead of the entire unit disk $\{|\mathbf{a}| \leq 1\}$.

3.5 Numerical Experiments

In all of our examples¹, we assume that the domain is a unit square, sometimes containing impenetrable and occluding obstacles. This square is discretized on a 201×201 grid ($h = 0.005$). E’s speed is uniform ($f(\mathbf{x}) = 1$) and the timestep is chosen according to the CFL stability condition: $\Delta t = h$. Our examples use observability functions $K_i(\mathbf{x}, t)$ of the form found in (3.6), with $K_0 = 1$ and $\sigma = 0.1$. In all examples, the Evader’s deadline for reaching the target is $T = 4$, but all Nash equilibrium controls lead to a much earlier arrival. (In our setting, the cumulative observability is computed up to the arrival time $T_{\mathbf{a}}$ only.)

Example 1: No obstacles, two patrol trajectories, and omnidirectional sensors (i.e., $\alpha = 2\pi$). Observer uses the same angular speed and moves counterclockwise on both patrol trajectories, which were already introduced in Fig. 3.2. Nash equilibrium policies are shown in Fig. 3.5. See also the Pareto Front in Fig. 3.3(A).

¹ Movies representing time-dependent Nash equilibrium policies for all of these examples can be found at https://eikonal-equation.github.io/TimeDependent_SEG.

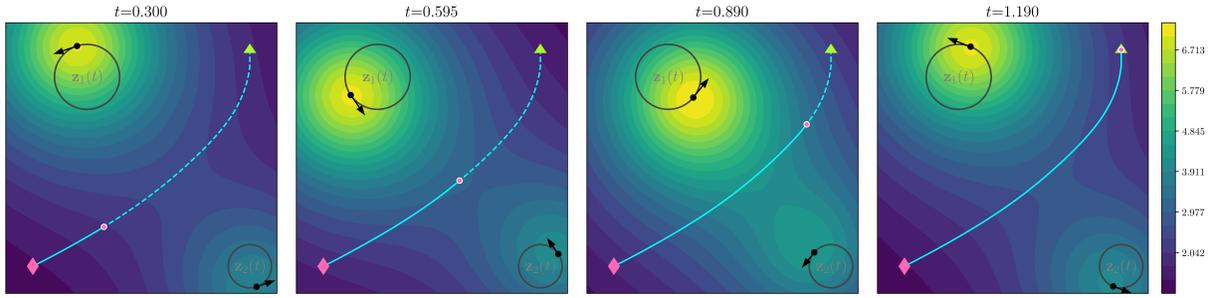


Figure 3.5: Four snapshots of the Nash Equilibrium solution for Example 1. Observer's optimal policy is $\lambda_* = (0.67, 0.33)$. Observer's patrol trajectories (in gray) and Evader's unique λ_* -optimal trajectory (magenta) shown on top of the contour plots of $K^{\lambda_*}(\mathbf{x}, t)$. Observer's possible position on each patrol trajectory is shown by a black circle, with an arrow indicating Observer's current heading. Evader's starting, current, and terminal positions are shown by a magenta diamond, a magenta circle, and a green triangle respectively.

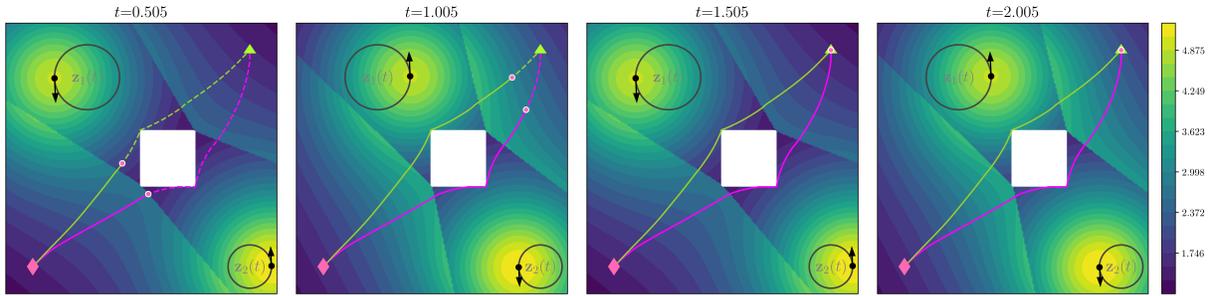


Figure 3.6: Four snapshots of the Nash Equilibrium solution for Example 2. Observer's optimal policy is $\lambda_* = (0.48, 0.52)$, and Evader's optimal policy is $\theta_* = (0.691, 0.309)$, with two different λ_* -optimal trajectories shown, one in yellow and one in blue corresponding to the yellow and blue points in Fig. 3.3B.

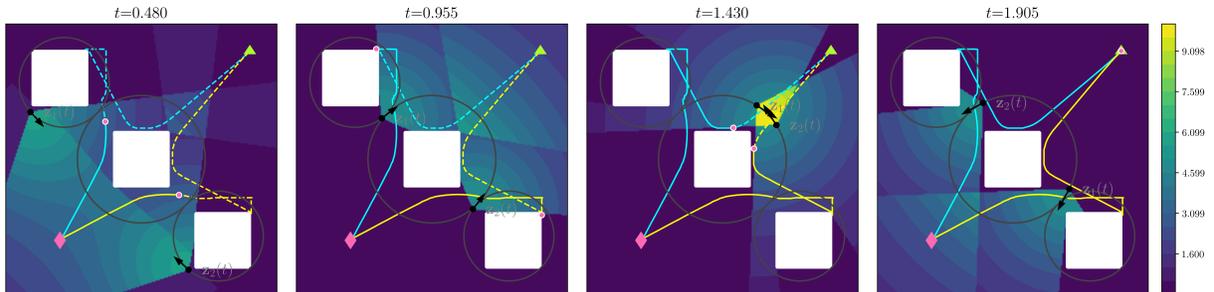


Figure 3.7: Four snapshots of the Nash Equilibrium solution for Example 3. Observer's optimal policy is $\lambda_* = (0.5, 0.5)$, and Evader's optimal policy is $\theta_* = (0.5, 0.5)$, with two different λ_* -optimal trajectories shown in cyan and yellow.

Example 2: One obstacle, with the same sensors and patrol trajectories as Example 1. Nash equilibrium policies are shown in Fig. 3.6. Evader’s optimal policy is probabilistic, relying on two λ_* -optimal trajectories that pass above and below the obstacle. See also the Pareto Front in Fig. 3.3(B).

Example 3: Three obstacles, two patrol trajectories with direction-restricted sensors ($\alpha = 2\pi/3$). The patrol trajectories each follow a figure-eight pattern. Nash equilibrium policies are shown in Fig. 3.7.

Example 4: Many obstacles, with 4 patrol trajectories on the same circle, but at different phases. Observer’s angular speed is $\omega = 1/2\pi$, and with direction-restricted sensors (i.e., $\alpha = 2\pi/3$). Nash equilibrium policies are shown in Fig. 3.8.

3.6 Conclusions

We have developed and tested a numerical method for SE games under uncertainty with isotropic Evader dynamics, a time-dependent Observer, and a direction-limited sensor. The efficiency of our approach stems from combining existing techniques from game theory, convex optimization, and dynamic programming in continuous state/time. The latter requires efficient numerical methods for time-dependent Eikonal PDEs. Incorporating more realistic models of Evader dynamics will require solving more general Hamilton-Jacobi PDEs. While our current implementation relies on an explicit first-order upwind scheme, the use of higher-order accurate discretizations [51, 115] would be desirable in the future. Additional speed improvements might be also attained through a time-dependent version of causal domain restriction [35]. All numerical examples presented above assumed that the Observer is a robotic platform with the direction of sensor/camera fixed relative to the Observer’s heading. However, the same methods would work almost without change for panning/tilting

cameras, Observers with multiple sensors, and more sophisticated observability models in the Observer's field of view [122]. It is also relatively easy to extend the approach to a centralized planner controlling several Evaders, but if all of them have separate targets, the cost of each iteration will increase accordingly. For a large group of Evaders, one will need to rely on distributed optimal control techniques or on Mean Field Game models, if these Evaders engage in selfish/independent trajectory planning.

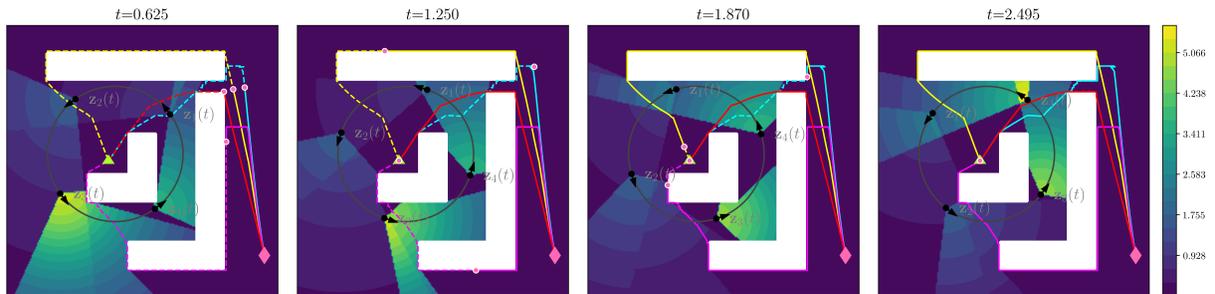


Figure 3.8: Four snapshots of the Nash Equilibrium solution for Example 4. Observer's optimal policy is $\lambda_* = (0.077, 0.127, 0.452, 0.344)$, and Evader's optimal policy is $\theta_* = (0.592, 0.084, 0.145, 0.180)$, with four different λ_* -optimal trajectories shown in cyan, yellow, magenta, and red.

CONTROL-THEORETIC MODELS OF ENVIRONMENTAL CRIME

4.1 Introduction

Mathematical modeling of environmental crimes in national parks has been a growing area of research. Both the authorities and non-governmental conservation organizations strive to prevent *illegal extraction activities*, including wildlife poaching and illegal logging. In 2010, Albers [3] introduced a framework where the government chooses a patrol density (determining the probability of detection at each location) in such a way as to maximize the pristine area of the park. The perpetrators then choose where to extract resources in such a way as to maximize their expected payoff. The problem of choosing patrol densities to maximize the pristine area has also been explored in [77]. In recent years, patrol strategies based on graph-theoretic models (such as PAWS [53] and INTERCEPT [79]) have even been deployed in actual national parks.

The original Albers model [3] was posed with many restrictive/unrealistic assumptions, including the radial symmetry of the protected area, patrol strategies, and resource distribution. In a recent paper by D.J. Arnold et al [8], these restrictions were largely removed, with numerical simulations on arbitrary two-dimensional domains and the terrain directly affecting the speed of perpetrators. The key mathematical tool employed in [8] is the level-set method, developed to track the propagation of interfaces by solving time-dependent Hamilton-Jacobi PDEs [98]. We begin by briefly reviewing both of these models in Section 4.2.

In this paper, we offer a significantly different perspective on how to generalize the Albers model to arbitrary domains. As explained in Section 4.3, our main focus is on a careful

treatment of the path-planning problem faced by the extractors. Capturing their rational behavior yields a more accurate prediction of affected areas and allows for a better planning of optimal patrol strategies. We show that the perpetrators optimization problem changes significantly based on the consequences of being spotted while transporting the protected resources.

In our first “Model G”, where authorities use “ground patrols” to spot the extractors, any detection leads to an immediate confiscation, with perpetrators subsequently switching to the fastest path to leave the protected area. The probability of being detected along any originally chosen path is not a sufficient description here, since it does not reflect the expected duration of that post-confiscation trip. We handle this challenge by finding the optimal controls for a suitable *randomly-terminated process*. In our second “Model A”, where authorities run “aerial patrols,” it is assumed that the perpetrators remain oblivious of when a detection happens and are only apprehended later, once they reach the boundary of the protected area. As a result, they always use their pre-selected trajectories, which are chosen to balance the probability of detection against the time or (cumulative difficulty) of the post-extraction trip. In standard path-planning applications, the goal is usually to minimize the integral of some “running cost” along a chosen trajectory. The key difficulty of this second model is that no such running cost can be explicitly defined to recover trajectories maximizing the extractor’s expected profit. Nevertheless, in section 4.4 we show that this problem can be solved using the tools of *multi-objective* optimal control theory.

We discuss efficient numerical methods for the associated stationary Hamilton-Jacobi-Bellman PDEs in Section 4.5. Our algorithms are illustrated on a number of examples in Section 4.6, including on realistic terrain maps from Yosemite National Park in California and Kangaroo Island in South Australia. We show that Models A and G often yield very different perpetrator trajectories and also differ in their predictions of the subdomain unaffected by

extraction activities. We conclude by discussing possible extensions and directions for future work in section 4.7.

4.2 Prior Work

The model proposed by Albers in [3] is based on a *one-period Stackelberg game* between the protected area managers (PAMs) and the extractors targeting a protected domain $\Omega \subset \mathbb{R}^2$. PAMs decide on their patrol strategy, defining the pointwise rate of extractor-detection $\psi : \Omega \rightarrow \mathbb{R}_{+,0}$ and respecting the known budget constraints for patrolling efforts:

$$\int_{\Omega} \psi^{\gamma}(\mathbf{x}) d\mathbf{x} \leq E, \quad (4.1)$$

with a constant $\gamma \geq 1$ modeling a possible increase in marginal costs of maintaining a higher detection rate.

Once ψ is chosen, it becomes known to extractors, who select which parts of Ω to target, trying to maximize their expected profit (defined below). Correspondingly, PAMs' goal is to select ψ that will minimize the impact of the extractors' best/rational response. This can be interpreted as either maximizing the “pristine” area unaffected by the extractors or maximizing the amount of non-extracted resources. In Albers' original model both of these interpretations are equivalent since she assumes a homogeneous distribution of protected resources.

Even more restrictively, the domain Ω is assumed to be a unit disk, with extractors evenly distributed along its boundary. This radial symmetry dramatically simplifies the problem and allows for analytical solutions for the optimal behavior of all participants. E.g., it is easy to see that only radially symmetric $\psi(\mathbf{x})$ would need to be considered – Albers further restricts this to piecewise-constant radial functions, with positive rate of detection in an

annulus between radial distances d_e and d_i from $\partial\Omega$ and $\psi = 0$ everywhere else. The same radial symmetry ensures that extractors travel along straight lines from $\partial\Omega$ toward the center of Ω and their entire strategy can be encoded by the chosen depth of penetration $d \in [0, 1]$. Albers assumes that extractors travel to their chosen location unimpeded (since they have not broken any law yet), then turn around and start extracting resources as they retrace their steps to $\partial\Omega$. If they are detected by PAMs on their way back, the extracted resources are confiscated and they have to return to $\partial\Omega$ empty-handed, incurring the time/opportunity cost without any reward. Thus, each extractor chooses d to maximize their expected profit

$$P(d) = (1 - \Psi(d))B(d) - C(d) = B(d) - (\Psi(d)B(d) + C(d)), \quad (4.2)$$

where Ψ , B , and C are respectively the cumulative probability of being detected, the value of extracted resources, and the cost of traversing the path – all corresponding to going distance d into Ω before starting to extract resources. While this is not explicitly specified in [3], C can also include the cost of getting to the point from which the extraction starts. A more subtle point is that B includes the value of *all* resources extracted along the way back. In the general setting, this continuous-along-the-path extraction would induce a competition among the extractors searching for the best paths through Ω , but the radial symmetry mitigates this effect. However, this makes it harder to justify the *one-period* nature of the model: as the extractors’ actions deplete the resources in parts of the protected area, that would obviously change the future optimizations problems (faced by both PAMs and the extractors) even in a radially symmetric setting. Such dynamic consequences of “incomplete enforcement” are acknowledged in [3] and also well-known in practice; e.g., [102, 103].

A recent paper by D.J. Arnold et al in [8] extended the above general approach to more realistic protected areas. Their model allows for general domain geometries, resource distributions, terrain effects, and PAMs’ strategies. One important distinction is that, unlike Albers, they model the extraction as occurring at isolated points. An advantage of this approach is that this reduces competition among the extractors (since the paths-to- $\partial\Omega$ tra-

versed by others don't affect the value of what you are currently trying to extract). For an extractor targeting a location $\mathbf{x} \in \Omega$, they define the expected profit as

$$P(\mathbf{x}) = B(\mathbf{x}) - \tilde{C}, \quad (4.3)$$

where B is the value of resources available for extraction in a small neighborhood of \mathbf{x} , while \tilde{C} is the effective cost of a post-extraction trajectory, accounting both for the time/difficulty of travel from \mathbf{x} to $\partial\Omega$ and for the expected losses from a possible capture. A perpetrator chooses this trajectory to minimize \tilde{C} (and thus maximize P). In [8] this optimization is performed implicitly by using a level set method originally developed for tracking the propagation of interfaces [98]. The authors show that, in an Albers-style radially symmetric setting with unit walking speed of perpetrators, their approach produces the same $P(\mathbf{x})$ as predicted by formula (4.2). A comparison with (4.3) thus suggests an explicit interpretation $\tilde{C}(\mathbf{x}) = \Psi B(\mathbf{x}) + C$.

For any chosen constant $b > 0$, Arnold et al compute $\tilde{C}(\mathbf{x}_0)$ simultaneously for all extraction positions \mathbf{x}_0 such that $B(\mathbf{x}_0) = b$. The idea is to find equal-effective-cost curves by using the level sets of an auxiliary time-dependent function $\phi(\mathbf{x}, t)$ satisfying a Hamilton-Jacobi PDE

$$\phi_t + F(\mathbf{x}, b)|\nabla\phi| = 0, \quad t > 0, \mathbf{x} \in \Omega, \quad (4.4)$$

with initial conditions $\phi(\mathbf{x}, 0) = \text{distance from } \mathbf{x} \text{ to } \partial\Omega$. Assuming that $\Gamma(t)$ is some interface starting from $\Gamma(0) = \partial\Omega$ and monotonically advancing into Ω with the normal speed F , its evolution is encoded by the zeroth level-set of ϕ solving the above PDE; i.e., $\phi(\mathbf{x}, t) = 0 \iff \mathbf{x} \in \Gamma(t)$. For the purposes of this model, the authors choose their normal speed of the interface to be

$$F(\mathbf{x}, b) = \frac{1}{1/f(\mathbf{x}) + \alpha\psi(\mathbf{x})b}, \quad (4.5)$$

where $f(\mathbf{x})$ is the extractor's local speed of motion defined by the terrain at \mathbf{x} , and a parameter $\alpha > 0$ encodes the degree of extractor's risk-aversion. This formula is motivated

phenomenologically: $1/f(\mathbf{x})$ represents the extractor’s “slowness of motion” and $\psi(\mathbf{x})b$ represents their expected losses (per unit time, while traveling through \mathbf{x}). An increase in either of these terms should make the effective cost higher. Once ϕ is computed, the authors define \tilde{C} implicitly through

$$\phi(\mathbf{x}_0, \tilde{C}(\mathbf{x}_0)) = 0, \quad \forall \mathbf{x}_0 \text{ such that } B(\mathbf{x}_0) = b,$$

with an optimal path to $\partial\Omega$ from any such \mathbf{x}_0 recovered via gradient descent in ϕ . The process is then repeated for multiple values of b , until \tilde{C} (and thus also P) are defined for all gridpoints of some grid imposed on Ω .

In another departure from Albers’ original model, the authors assume that extractors will target all locations where P is within $(1 - \varepsilon)$ factor from its maximal value; i.e., the extraction happens on the set

$$\Omega_e = \left\{ \bar{x} \in \Omega \mid P(\bar{x}) \geq (1 - \varepsilon) \max_{\mathbf{x} \in \Omega} P(\mathbf{x}) \right\}.$$

(One drawback of this assumption is that extraction will always happen on some part of Ω , regardless of how high the detection rate $\phi(\mathbf{x})$ might be.) The authors then define the pristine area by excluding not only the above but also the union of narrow corridors around optimal trajectories from Ω_e . The latter are approximated by starting from a finite number of points selected uniformly at random in Ω_e . Since there is no extraction along the path, the motivation for this exclusion might perhaps be based on extraction-unrelated damage to the protected area. While [8] does not present a systematic method for optimizing PAMs’ choice of ψ , the authors use the above interpretation to compare the effectiveness of several (ad hoc) patrol strategies staying within the specified patrol budget. Their simulations (using $\gamma = 1, \alpha = 1$, and $\varepsilon = 0.085$) are based on the terrain data from Yosemite National Park in California and Kangaroo Island in South Australia, which they have also kindly shared with us to conduct the numerical experiments presented in section 4.6.

We end this section by discussing the difference between path planning to avoid detection vs capture/interception. Even if the former always leads to the latter, there is still a very subtle modeling question of whether the capture happens immediately upon detection and, if there is a significant delay, whether the extractors learn that they have been detected before the capture. These issues are not important in [3] since the extractors’ paths are always radial in Albers’ setting. But Arnold et al describe path-planning that strikes a balance between the time of travel and risk of detection all the way to the boundary – with no provision for switching to quicker/easier paths if authorities manage to intercept the extractors en route and confiscate their haul. Thus, we believe that the model in [8] is more suitable for the setting where the extractors remain oblivious of their detection (perhaps in the case of aerial surveillance) and are only intercepted upon reaching $\partial\Omega$.

4.3 Extractor’s path-planning

This section defines the optimization problem faced by a perpetrator deciding whether to extract resources from a fixed location $\mathbf{x} \in \Omega$. As in [8], our approach is based on “point-wise” extraction decisions, but since these resources are typically continuously distributed in the protected area Ω , this requires further clarification. We will consider a fine grid \mathcal{X} of possible extraction sites, with each $\mathbf{x} \in \mathcal{X}$ representing a center of its extraction cell $N_{\mathbf{x}}$, and $B(\mathbf{x})$ representing the value of all resources that can be extracted from that cell. The same grid \mathcal{X} will be also used to solve the PDEs numerically in section 4.5.

We will further assume that

1. The extractor can enter Ω through any part of $\partial\Omega$ and has no reason to avoid surveillance from PAMs until they reach the extraction site \mathbf{x} .

2. While traveling through Ω , the extractor may freely choose any measurable “control function” $\mathbf{a} : \mathbb{R} \rightarrow S^1$, specifying their chosen direction of motion for all times t . This defines the corresponding trajectory since their (isotropic) speed of motion $f : \Omega \rightarrow \mathbb{R}_+$ reflects the local terrain and is a part of the input data.
3. The time/difficulty cost of traveling through Ω is reflected by integrating some known running cost $K : \Omega \rightarrow \mathbb{R}_+$ along their chosen trajectory. One natural choice is $K(\mathbf{x}) = \kappa$, reflecting the monetary value the extractor associates with each hour of their time spent on a trip. (To normalize the units, all of our numerical examples use $\kappa = 1$ for all pre-extraction and post-extraction travel through Ω .) But a non-constant K can be also used to reflect the extractor’s distaste for some parts of the protected area (e.g., due to vegetation).
4. Once they reach \mathbf{x} , the extraction in $N_{\mathbf{x}}$ is instantaneous and the extractor immediately starts along their (possibly different) path to the boundary.
5. On the way back to $\partial\Omega$, they are trying to minimize both the time/cost of travel and the probability of being spotted by PAMs through aerial surveillance.
6. The extractor has full prior knowledge of PAMs’ location-dependent detection rate $\psi : \Omega \rightarrow \mathbb{R}_{+,0}$.
7. Rational extractors will target all sites $\mathbf{x} \in \mathcal{X}$ for which the expected profit $P(\mathbf{x})$ of their optimal roundtrip is above some threshold level \tilde{p} . This leaves a pristine area

$$\Omega_p = \{\mathbf{x} \mid P(\mathbf{x}) \leq \tilde{p}\}.$$

(Our numerical experiments use $\tilde{p} = 0$.) Note that this is different from the approach in [8] not only in providing an absolute threshold, but also because we don’t exclude from Ω_p any corridors around post-extraction trajectories. We believe that this is more consistent with the notion of localized extractions.

Computing the expected profit $P(\mathbf{x})$ requires solving two different optimization problems: (a) finding the best (PAMs-ignoring) pre-extraction trajectory from $\partial\Omega$ to \mathbf{x} and then (b) finding the best post-extraction trajectory from \mathbf{x} to $\partial\Omega$, balancing the cost of travel and risk of detection. We will start by addressing (a) in section 4.3.1 since it is easy to solve by standard tools of *single-criterion deterministic optimal control theory*. Below we provide only a very brief introduction, referring to standard references (e.g., [22] and [9]) for technical details.

We will then use two models to address (b) under very different assumptions about the nature of PAMs' enforcement activities and the information available to perpetrators:

Model G (ground patrols):

If the extractor is detected while traveling with protected resources, they are immediately apprehended and the resources are confiscated. From there on, the perpetrator does not attempt extracting anything on this trip and simply seeks to minimize the integral of K on their way to $\partial\Omega$.

Model A (aerial patrols):

The extractor has no way of learning if or when the detection occurs. But if it happens, they are later apprehended with probability one upon reaching $\partial\Omega$ and his entire haul is confiscated.

We will show that these two interpretation require fairly different mathematical tools: Model G is best recast as an optimal control of a randomly-terminated process (subsection 4.3.2), while Model A requires the techniques from multi-objective dynamic programming

(subsection 4.3.3). The numerical methods for both versions are covered in section 4.5.

4.3.1 Getting to the extraction point

We will use $R(\mathbf{x})$ to denote the cost of the optimal pre-extraction trajectory from $\partial\Omega$ to \mathbf{x} . But to make the discussion closer to the canonical optimal control setting, we will reverse the direction of the trajectory in (a), looking instead for a K -optimal way to reach $\partial\Omega$ from \mathbf{x} . (Based on our isotropy assumptions, this direction reversal will change neither the time of travel nor the integral of K along any considered trajectory.) The added benefit is that the same discussion will be a useful starting point for solving (b).

The extractor's time-dependent position in the domain will be specified by

$$\mathbf{y}'(t) = f(\mathbf{y}(t)) \mathbf{a}(t), \quad \mathbf{y}(0) = \mathbf{x}, \quad (4.6)$$

Once $\mathbf{a}(\cdot)$ is chosen, this defines the exit time $T_{\mathbf{x},\mathbf{a}(\cdot)} = \min\{t \geq 0 \mid \mathbf{y}(t) \in \partial\Omega\}$ and the cumulative cost of starting at \mathbf{x} and using $\mathbf{a}(\cdot)$:

$$\mathcal{J}(\mathbf{x}, \mathbf{a}(\cdot)) = \int_0^{T_{\mathbf{x},\mathbf{a}(\cdot)}} K(\mathbf{y}(t)) dt.$$

If the extractor only cares about selecting a path to minimize this \mathcal{J} , it is easy to accomplish by traditional tools of single-criterion dynamic programming introduced by Richard Bellman in the 1950s.

The *value function* $u(\mathbf{x})$ is defined as the minimum cost one has to pay starting from \mathbf{x} ; i.e., $u(\mathbf{x}) = \inf_{\mathbf{a}(\cdot)} \mathcal{J}(\mathbf{x}, \mathbf{a}(\cdot))$. A Taylor series expansion along the optimal trajectory can be used to derive a Hamilton-Jacobi-Bellman (HJB) PDE that u must satisfy if it is sufficiently smooth. For our isotropic cost and dynamics, this PDE is merely an Eikonal equation

$$|\nabla u(\mathbf{x})| f(\mathbf{x}) = K(\mathbf{x}) \quad (4.7)$$

solved with Dirichlet boundary conditions $u = 0$ on $\partial\Omega$. Unfortunately, in general (4.7) does not have a classical/smooth solution, and there are infinitely many weak (Lipschitz-continuous) solutions. Additional test conditions were introduced by Crandall and Lions [38] to select among them the unique *viscosity solution* coinciding with the value function of the above control problem. Several highly efficient methods for solving Eikonal PDEs numerically were developed in the past 25 years; see a brief overview at the end of section 4.5.

We note that taking $K = 1$ results in $\mathcal{J}(\mathbf{x}, \mathbf{a}(\cdot)) = T_{\mathbf{x}, \mathbf{a}(\cdot)}$. So, both the distance-from-boundary $d(\mathbf{x})$ and the min-time-from-boundary $\tau(\mathbf{x})$ can be also found by solving the Eikonal equations $|\nabla d| = 1$ and $|\nabla \tau|f = 1$ with zero boundary conditions. If $K = \kappa$, then the minimum cost of reaching \mathbf{x} from $\partial\Omega$ is simply

$$R(\mathbf{x}) = \kappa\tau(\mathbf{x}).$$

The characteristic curves of HJB equations are the optimal trajectories, which in the Eikonal case coincide with the gradient lines of the viscosity solution. I.e., once the value function is computed, an optimal trajectory can be found via gradient descent in u . Since this function is differentiable almost everywhere, the optimal trajectories are unique for almost all $\mathbf{x} \in \Omega$.

Remark 2. We note that the function \tilde{C} used in [8] can also be found directly (without using a level-set formulation) by solving an Eikonal equation $|\nabla \tilde{C}|F = 1$. While the level set method is a much more general interface tracking approach, the fast Eikonal solvers are likely to be more efficient for the current application.

4.3.2 Returning with the loot (Model G: ground patrols)

Here we assume that a detection leads to an immediate apprehension and confiscation of extracted resources. Thus, it is natural to split the post-extraction trajectory into two parts: a pre-detection trajectory (leading all the way to $\partial\Omega$ if the detection was avoided) and a post-detection trajectory (chosen by a perpetrator to simply minimize the integral of K up to $\partial\Omega$ after the confiscation). Suppose the perpetrator starts from an extraction point \mathbf{x}_0 and manages to reach some $\mathbf{x} \in \Omega$ undetected. From there on, they use a control $\mathbf{a}(\cdot)$, moving from $\mathbf{x} = \mathbf{y}(0)$ along a trajectory $\mathbf{y}(t)$ with a possible detection at some time T before reaching the boundary. Their remaining cost is

$$\mathcal{J}(\mathbf{x}, \mathbf{a}(\cdot), T) = \begin{cases} \int_0^T K(\mathbf{y}(t)) dt + B(\mathbf{x}_0) + R(\mathbf{y}(T)), & \text{if } T < T_{\mathbf{x}, \mathbf{a}(\cdot)} \\ & \text{(i.e., if detected)} \\ \int_0^{T_{\mathbf{x}, \mathbf{a}(\cdot)}} K(\mathbf{y}(t)) dt, & \text{otherwise.} \end{cases}$$

Of course, the time of capture T is random and its distribution depends on the chosen pre-detection trajectory. So, the perpetrator's value function is defined to minimize the expected remaining cost (from \mathbf{x} to $\partial\Omega$):

$$\bar{u}(\mathbf{x}) = \inf_{\mathbf{a}(\cdot)} \mathbb{E}_T [\mathcal{J}(\mathbf{x}, \mathbf{a}(\cdot), T)].$$

We note that it also depends on the value of extracted resources, $b = B(\mathbf{x}_0)$, but we leave this implicit to simplify the notation.

To obtain the Bellman optimality condition, we assume that the detection will not happen for the next s seconds, but might occur after that period with probability $s\psi(\mathbf{x})$. This yields

$$\bar{u}(\mathbf{x}) = \inf_{\mathbf{a}(\cdot)} \left\{ \int_0^s K(\mathbf{y}(t)) dt + s\psi(\mathbf{x}) [b + R(\mathbf{y}(s))] + (1 - s\psi(\mathbf{x})) \bar{u}(\mathbf{y}(s)) \right\} + o(s).$$

Assuming that \bar{u} is sufficiently smooth, a Taylor expansion of the above yields a Hamilton-Jacobi PDE

$$\begin{aligned} |\nabla \bar{u}(\mathbf{x})| f(\mathbf{x}) &= K(\mathbf{x}) + \psi(\mathbf{x}) (b + R(\mathbf{x}) - \bar{u}(\mathbf{x})), & \text{on } \Omega; \\ \bar{u}(\mathbf{x}) &= 0, & \text{on } \partial\Omega. \end{aligned} \quad (4.8)$$

Since this PDE generally does not have a classical/smooth solution, that derivation is only formal, but viscosity solution theory [38, 9] allows us to pick the weak solution coinciding with the value function. The above derivation is similar to what is used in optimal control of *randomly-terminated process*. Such processes arise in production/maintenance planning [20], economic growth and global climate change modeling [64], multi-generational games [65], and optimizing the routing of emergency response vehicles [6]. In the current context, “termination” is either “detection/capture” (since we already know the cost of optimal actions after that point) or “reaching $\partial\Omega$ undetected”. The latter possibility yields Dirichlet boundary conditions in (4.8) and avoids the issues related to “free boundary” and quasi-variational inequalities considered in [6].

Once \bar{u} is found, the optimal pre-detection trajectories can be found by gradient descent. The extractor’s expected profit can be also computed as

$$P^g(\mathbf{x}_0) = b - \bar{u}(\mathbf{x}_0) - R(\mathbf{x}_0), \quad \forall \mathbf{x}_0 \in X \text{ such that } B(\mathbf{x}_0) = b,$$

where $R(\mathbf{x}_0)$ accounts for the cost of a pre-extraction trajectory.

We note that it is easy to find an upper bound on \bar{u} by committing to use some $\mathbf{a}(\cdot)$ up until $\partial\Omega$ even if the detection occurs earlier. If that $\mathbf{a}(\cdot)$ is selected to minimize the integral of K , this yields $\bar{u}(\mathbf{x}_0) \leq b + R(\mathbf{x}_0)$, but a sharper bound is obtained by choosing a control optimal for Model A considered below. This is why $P^g \geq P^a$ and $\Omega_p^g \subset \Omega_p^a$.

4.3.3 Returning with the loot (Model A: aerial patrols)

On the way back to $\partial\Omega$, the extractor needs to balance the cumulative cost of the path with expected losses resulting from a possible detection by PAMs. This is even more important under the current model, where the extractor does not learn whether he was detected until reaching $\partial\Omega$. Here we formally derive the probability that an extractor is not detected while traversing a path $\mathbf{y} : [0, T] \rightarrow \Omega$. Suppose the probability of detection at a location \mathbf{x} over a small time interval Δt is $\psi(\mathbf{y})\Delta t$. Then the probability \mathcal{P}_N of making it through N such consecutive intervals of length $\Delta t = T/N$ without ever being detected is:

$$\mathcal{P}_N = (1 - \psi(\mathbf{y}_1)\Delta t)(1 - \psi(\mathbf{y}_2)\Delta t) \cdots (1 - \psi(\mathbf{y}_N)\Delta t) = \prod_{i=1}^N (1 - \psi(\mathbf{y}_i)\Delta t).$$

Taking the logarithm of both sides,

$$\log(\mathcal{P}_N) = \sum_{i=1}^N \log(1 - \psi(\mathbf{y}_i)\Delta t) = -\sum_{i=1}^N (\psi(\mathbf{y}_i)\Delta t + \mathcal{O}(\Delta t^2)).$$

Taking the limit as $N \rightarrow \infty$, the probability of not being detected along this path until at least the time T is

$$\mathcal{P} = e^{-\int_0^T \psi(\mathbf{y}(t))dt}. \quad (4.9)$$

Our emphasis on careful modeling of the cumulative probability of detection $\Psi = 1 - \mathcal{P}$ is perhaps the main distinction from earlier models. Both [3] and [8] implicitly assume $\Psi \approx \int_0^T \psi(\mathbf{y}(t))dt$. As a first-order approximation of a concave function ($1 - e^{-\xi} = \xi + \mathcal{O}(\xi^2)$), this overestimates Ψ for every trajectory. It also yields a simplified optimization problem, but with significant limitations described in Remark 3. Instead, we find the extractor's

expected profit by using the accurate \mathcal{P} to optimize his post-extraction trajectory:

$$\begin{aligned}
\mathbf{y}'(t) &= f(\mathbf{y})\mathbf{a}(t), & \mathbf{y}(0) &= \mathbf{x} \\
T_{\mathbf{x},\mathbf{a}(\cdot)} &= \min\{t \geq 0 \mid \mathbf{y}(t) \in \partial\Omega\} \\
\mathcal{J}_1(\mathbf{x}, \mathbf{a}(\cdot)) &= \int_0^{T_{\mathbf{x},\mathbf{a}(\cdot)}} \psi(\mathbf{y}(t)) dt \\
\mathcal{J}_2(\mathbf{x}, \mathbf{a}(\cdot)) &= \int_0^{T_{\mathbf{x},\mathbf{a}(\cdot)}} K(\mathbf{y}(t)) dt \\
P^a(\mathbf{x}) &= \sup_{\mathbf{a}(\cdot) \in \mathcal{A}} \{B(\mathbf{x})e^{-\mathcal{J}_1(\mathbf{x},\mathbf{a}(\cdot))} - \mathcal{J}_2(\mathbf{x}, \mathbf{a}(\cdot))\} - R(\mathbf{x}),
\end{aligned} \tag{4.10}$$

where \mathcal{A} is the set of *admissible* controls, i.e., the set of measurable functions from \mathbb{R} to S^1 , and $R(\mathbf{x})$ is the optimal cost of a pre-extraction path defined in subsection 4.3.1.

Remark 3. If $\mathcal{J}_1(\mathbf{x}_0, \mathbf{a}(\cdot)) \ll 1$ for all \mathbf{x}_0 and all controls $\mathbf{a}(\cdot)$ close to optimizing (4.10), a linear approximation of the detection probability $\Psi \approx \mathcal{J}_1$ is reasonable and leads to a significantly simpler optimization problem similar to those considered in [3] and [8]. In that case,

$$P^a(\mathbf{x}) \approx B(\mathbf{x}) - \inf_{\mathbf{a}(\cdot) \in \mathcal{A}} \left\{ B(\mathbf{x})\mathcal{J}_1(\mathbf{x}, \mathbf{a}(\cdot)) + \mathcal{J}_2(\mathbf{x}, \mathbf{a}(\cdot)) \right\} - R(\mathbf{x}). \tag{4.11}$$

Since the linearized approach exaggerates Ψ , we know that this estimate is actually a lower bound on $P^a(\mathbf{x})$. The minimizing control in (4.11) can be found by solving the Eikonal PDE

$$|\nabla u|f(\mathbf{x}) = B(\mathbf{x}_0)\psi(\mathbf{x}) + K(\mathbf{x}) \tag{4.12}$$

with zero boundary conditions. If one prefers the level set formulation, this is equivalent to using the interface speed of

$$F(\mathbf{x}, b) = f(\mathbf{x}) / (K(\mathbf{x}) + b\psi(\mathbf{x})),$$

which is similar to but not quite the same as the formula (4.5) used in [8] – even in the case of risk neutral extractors (i.e., $\alpha = 1$) concerned with their time along the trajectory (i.e., $K(\mathbf{x}) = 1$).

Unfortunately, \mathcal{J}_1 can be arbitrarily large, which makes this “ Ψ -linearization” unsuitable for most realistic situations. (Indeed, in our numerical examples of section 4.6, $\mathcal{J}_1 > 1$ is often observed along many optimal paths.) This is the key reason why the more complicated multi-objective path-planning methods considered below are actually needed.

Remark 4. The non-linear dependence on \mathcal{J}_1 is the reason why P^a cannot be computed by standard tools of single-objective optimal control on Ω . But this actually can be accomplished if we are willing to increase the dimension of our planning space. We note that \mathcal{P} satisfies an ODE $\mathcal{P}'(t) = -\psi(\mathbf{y}(t))\mathcal{P}(t)$ along the chosen trajectory, with $\mathcal{P}(0) = 1$, $\mathbf{y}(0) = \mathbf{x}_0$. One could define a new value function $w(\mathbf{x}, p)$ as the minimum expected cost of the remaining path to $\partial\Omega$ if the extractor has already moved from \mathbf{x}_0 to \mathbf{x} and the probability that they have not been detected so far is p . A formal control-theoretic argument shows that w should satisfy the following Hamilton-Jacobi PDE on $\Omega \times (0, 1]$:

$$\psi(\mathbf{x})p \frac{\partial w}{\partial p} + |\nabla_{\mathbf{x}} w| f(\mathbf{x}) = K(\mathbf{x}) \quad (4.13)$$

with boundary conditions $w(\mathbf{x}, p) = (1-p)b$ on $\partial\Omega \times (0, 1]$. Once w is computed, the expected profit can be found as $P^a(\mathbf{x}_0) = b - w(\mathbf{x}_0, 1) - R(\mathbf{x}_0)$ for every \mathbf{x}_0 such that $B(\mathbf{x}_0) = b$. The need to solve this PDE on a higher dimensional domain for many b values would make this approach computationally expensive, with additional numerical difficulties due to that fact that the coefficient $(\psi(\mathbf{x})p)$ can be arbitrarily small or even zero. This is why we opt instead to use multi-objective dynamic programming on the original planning space Ω .

4.4 Multi-Objective Approach (Model A)

For convenience, we will define

$$\mathcal{F}(\mathcal{J}_1, \mathcal{J}_2) = Be^{-\mathcal{J}_1} - \mathcal{J}_2,$$

and in a slight abuse of notation, drop the first argument from both \mathcal{J}_1 and \mathcal{J}_2 (since \mathbf{x} is fixed throughout this section), and also drop the argument from $\mathbf{a}(\cdot)$.

Ideally, one would prefer to find a control minimizing both \mathcal{J}_1 and \mathcal{J}_2 simultaneously. Somewhat surprisingly this is actually possible when K is constant and PAMs use “ τ -banded” patrol densities; i.e., when ψ is really a function of the min-time-from- $\partial\Omega$ $\tau(\mathbf{x})$ defined in section 4.3. It is not hard to show that in this case the trajectories recovered from the Eikonal (4.7) will also minimize the probability of detection. When f is also constant, such trajectories will be simply straight lines (from the extraction point \mathbf{x} to the closest point on $\partial\Omega$), which will also minimize the probability of detection for all “distance-to- $\partial\Omega$ -banded” patrol strategies $\psi(d)$ (including the homogeneous $\psi(\mathbf{x}) = \psi_0$) considered both in [3] and [8]. This is also the setting where there is no difference between the predictions based on Model A and Model G – even if the loot is confiscated, there is no incentive to switch to another trajectory.

For the above scenarios the extractor’s path-planning is fairly trivial. But in the general case, these two minimization criteria are in conflict, and one has to switch to a notion of *Pareto optimality*.

4.4.1 Pareto Front

We will say that a control \mathbf{a}_1 *dominates* another control \mathbf{a}_2 if $\mathcal{J}_1(\mathbf{a}_1) \leq \mathcal{J}_1(\mathbf{a}_2)$ and $\mathcal{J}_2(\mathbf{a}_1) \leq \mathcal{J}_2(\mathbf{a}_2)$ with at least one of those inequalities strict. All \mathbf{a} ’s not dominated by any other control are called *Pareto optimal*. They correspond to trajectories that cannot be improved with respect to both criteria simultaneously. It will be useful to consider the *Pareto Front*

$$PF = \{(\mathcal{J}_1(\mathbf{a}), \mathcal{J}_2(\mathbf{a})) \mid \mathbf{a} \text{ is Pareto optimal} \}.$$

In addition to control functions, the extractor could in principle also consider mixed/probabilistic strategies, which specify a probability distribution over a set of available controls. A strategy selecting any control with probability one is usually called pure or deterministic. Since \mathcal{F} is decreasing in both arguments, it follows that $\mathcal{F}(\mathcal{J}_1(\mathbf{a}_1), \mathcal{J}_2(\mathbf{a}_1)) > \mathcal{F}(\mathcal{J}_1(\mathbf{a}_2), \mathcal{J}_2(\mathbf{a}_2))$ for any control \mathbf{a}_1 dominating \mathbf{a}_2 . From this we see that any optimizer of (4.10) over the set of pure strategies must belong to PF. We also note that the extractor has no incentive to consider mixed strategies, since the expected payoff for using strategy \mathbf{a}_1 with probability θ and \mathbf{a}_2 with probability $1 - \theta$ is

$$\theta \mathcal{F}(\mathcal{J}_1(\mathbf{a}_1), \mathcal{J}_2(\mathbf{a}_1)) + (1 - \theta) \mathcal{F}(\mathcal{J}_1(\mathbf{a}_2), \mathcal{J}_2(\mathbf{a}_2)) \leq \max_{i=1,2} \mathcal{F}(\mathcal{J}_1(\mathbf{a}_i), \mathcal{J}_2(\mathbf{a}_i))$$

So in practice, we will compute $P^a(\mathbf{x})$ by maximizing $\mathcal{F}(\mathcal{J}_1, \mathcal{J}_2)$ over the $(\mathcal{J}_1, \mathcal{J}_2) \in PF$.

There are a number of techniques for finding Pareto-optimal controls including *scalarization* on Ω [93] and other methods relying on augmented PDEs on an expanded state space [81, 46]. To scalarize, one chooses some $\lambda \in [0, 1]$ and optimizes a convex combination $\mathcal{J}^\lambda = \lambda \mathcal{J}_1 + (1 - \lambda) \mathcal{J}_2$. The resulting λ -optimal trajectory is also Pareto-optimal (see Figure 4.1), and the procedure is applied repeatedly for different λ values. Unfortunately, such scalarization only finds points on the convex part of the Pareto Front [42]. This is a serious disadvantage since non-convexities of PF are both common and important in many control applications including robotic navigation [81, 36]. In other cases, such as our recent work on surveillance-evasion games [56, 27], special features of the problem might make scalarization sufficient. Below we explain why this is also the case for the post-extraction path-planning.

Observation 5. Any global maximizer of (4.10) will correspond to a point on the convex part of PF.

Proof. Suppose a control \mathbf{a} corresponds to a point $(\mathcal{J}_1(\mathbf{a}), \mathcal{J}_2(\mathbf{a}))$ on a non-convex section

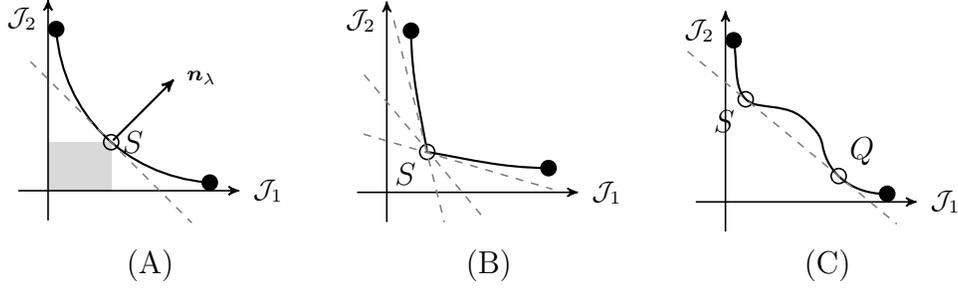


Figure 4.1: Common Pareto Front scenarios. (A) Convex smooth PF with a point S corresponding to some specific λ . The dashed line perpendicular to $\mathbf{n}_\lambda = (\lambda, 1 - \lambda)$ is tangent to PF at S . If any path yielded a $(\mathcal{J}_1, \mathcal{J}_2)$ point below this line, the path corresponding to S would not be λ -optimal. Since the grey rectangle (comprised of points dominating S) lies below that line, λ -optimality implies Pareto optimality $\forall \lambda \in (0, 1)$. (B) Convex non-smooth PF with a ‘kink’ at the point S makes the corresponding path λ -optimal for a range of λ 's, with a different “support hyperplane” corresponding to each of them. (C) Non-convex smooth PF. Points S and Q correspond to 2 different λ -optimal paths. The portion of PF between S and Q cannot be found by scalarization.

of PF. Then there exist controls $\mathbf{a}_1, \mathbf{a}_2$ and a constant $0 \leq \theta \leq 1$ such that

$$\theta \mathcal{J}_1(\mathbf{a}_1) + (1 - \theta) \mathcal{J}_1(\mathbf{a}_2) \leq \mathcal{J}_1(\mathbf{a})$$

$$\theta \mathcal{J}_2(\mathbf{a}_1) + (1 - \theta) \mathcal{J}_2(\mathbf{a}_2) \leq \mathcal{J}_2(\mathbf{a})$$

with at least one of the inequalities strict. Since \mathcal{F} is decreasing and convex in both arguments,

$$\begin{aligned} \mathcal{F}(\mathcal{J}_1(\mathbf{a}), \mathcal{J}_2(\mathbf{a})) &< \mathcal{F}\left(\theta \mathcal{J}_1(\mathbf{a}_1) + (1 - \theta) \mathcal{J}_1(\mathbf{a}_2), \theta \mathcal{J}_2(\mathbf{a}_1) + (1 - \theta) \mathcal{J}_2(\mathbf{a}_2)\right) \\ &\leq \theta \mathcal{F}\left(\mathcal{J}_1(\mathbf{a}_1), \mathcal{J}_2(\mathbf{a}_1)\right) + (1 - \theta) \mathcal{F}\left(\mathcal{J}_1(\mathbf{a}_2), \mathcal{J}_2(\mathbf{a}_2)\right) \\ &\leq \max_{i=1,2} \{\mathcal{F}(\mathcal{J}_1(\mathbf{a}_i), \mathcal{J}_2(\mathbf{a}_i))\} \end{aligned}$$

Thus, such \mathbf{a} cannot be a maximizer in (4.10). □

4.4.2 Scalarized Eikonal

For any $\lambda \in [0, 1]$, we define the scalarized running cost K^λ as

$$K^\lambda(\mathbf{x}) = \lambda\psi(\mathbf{x}) + (1 - \lambda)K(\mathbf{x}). \quad (4.14)$$

A standard optimal control argument [9] shows that the value function

$$u^\lambda(\mathbf{x}) = \inf_{\mathbf{a}(\cdot) \in \mathcal{A}} \left\{ \lambda \mathcal{J}_1(\mathbf{x}, \mathbf{a}(\cdot)) + (1 - \lambda) \mathcal{J}_2(\mathbf{x}, \mathbf{a}(\cdot)) \right\} \quad (4.15)$$

is the unique viscosity solution [38] of the scalarized Eikonal equation:

$$\begin{aligned} f(\mathbf{x}) |\nabla u^\lambda(\mathbf{x})| &= K^\lambda(\mathbf{x}), & \mathbf{x} \in \Omega \\ u^\lambda(\mathbf{x}) &= 0, & \mathbf{x} \in \partial\Omega. \end{aligned} \quad (4.16)$$

For every specific λ , we can recover the corresponding λ -optimal paths by gradient descent in u^λ . We note that taking $\lambda = 0$ corresponds to detection-indifferent optimization, yielding the PDE (4.7), which we already needed to solve to find R along with the optimal pre-extraction path from $\partial\Omega$ to each site \mathbf{x}_0 . The case $\lambda = 1$ yields the PDE for recovering the time-indifferent least-detectable paths.

Remark 6. We note that a proper choice of λ can also represent the PDE model based on the “ $\Psi \approx \mathcal{J}_1$ ” approximation discussed in Remark 3. In particular, for $\lambda_\sharp = B(\mathbf{x}_0)/(B(\mathbf{x}_0) + 1)$, the equation (4.15) can be rewritten as

$$(B(\mathbf{x}_0) + 1) f(\mathbf{x}) |\nabla u^{\lambda_\sharp}(\mathbf{x})| = B(\mathbf{x}_0)\psi(\mathbf{x}) + K(\mathbf{x}).$$

As a result, $(B(\mathbf{x}_0) + 1) u^{\lambda_\sharp}(\mathbf{x})$ is the viscosity solution of (4.12). When $\mathcal{J}_1 \ll 1$, this yields “nearly optimal” post-extraction paths. Based on (4.11), we also know that

$$P^a(\mathbf{x}_0) \geq P_\sharp(\mathbf{x}_0) = B(\mathbf{x}_0) - (B(\mathbf{x}_0) + 1) u^{\lambda_\sharp}(\mathbf{x}_0) - R(\mathbf{x}_0). \quad (4.17)$$

4.4.3 Auxiliary PDEs

However, we are ultimately interested in the expected payoff (4.10), not the value function $u^\lambda(\mathbf{x})$. Under suitable regularity assumptions on the speed f and running cost K , it can be shown that an optimal control $\mathbf{a}(\cdot)$ exists for every $\mathbf{x} \in \Omega$, and the infimum in formula (4.15) can be replaced with a minimum [9]. We define the set of λ -optimal controls $\mathcal{A}_\mathbf{x}^\lambda$ starting from $\mathbf{x} \in \Omega$ as:

$$\mathcal{A}_\mathbf{x}^\lambda = \arg \min_{\mathbf{a}(\cdot) \in \mathcal{A}} \left\{ \lambda \mathcal{J}_1(\mathbf{x}, \mathbf{a}(\cdot)) + (1 - \lambda) \mathcal{J}_2(\mathbf{x}, \mathbf{a}(\cdot)) \right\} \quad (4.18)$$

We note that the optimal control $\mathbf{a}(\cdot)$ will be unique at every point \mathbf{x} where $u^\lambda(\mathbf{x})$ is differentiable. Given our conditions on f and K , $u^\lambda(\mathbf{x})$ is Lipschitz-continuous, and so $\mathcal{A}^\lambda(\mathbf{x})$ will be a singleton for almost every $\mathbf{x} \in \Omega$.

We will now define the λ -optimal-restricted value functions v_1^λ and v_2^λ as:

$$v_1^\lambda(\mathbf{x}) = \inf_{\mathbf{a}(\cdot) \in \mathcal{A}_\mathbf{x}^\lambda} \left\{ \mathcal{J}_1(\mathbf{x}, \mathbf{a}(\cdot)) \right\} \quad (4.19)$$

$$v_2^\lambda(\mathbf{x}) = \inf_{\mathbf{a}(\cdot) \in \mathcal{A}_\mathbf{x}^\lambda} \left\{ \mathcal{J}_2(\mathbf{x}, \mathbf{a}(\cdot)) \right\} \quad (4.20)$$

Given the solution $u^\lambda(\mathbf{x})$ of (4.16), we can recover v_1^λ and v_2^λ by solving the following system of auxiliary linear PDEs introduced in [93]:

$$\begin{aligned} \nabla v_1^\lambda(\mathbf{x}) \cdot \nabla u^\lambda(\mathbf{x}) &= \frac{\psi(\mathbf{x})K^\lambda(\mathbf{x})}{f(\mathbf{x})^2} \\ \nabla v_2^\lambda(\mathbf{x}) \cdot \nabla u^\lambda(\mathbf{x}) &= \frac{K(\mathbf{x})K^\lambda(\mathbf{x})}{f(\mathbf{x})^2} \end{aligned} \quad (4.21)$$

with $v_1^\lambda = v_2^\lambda = 0$ on $\partial\Omega$.

Since we have shown in Section 4.4.1 that the global optimizer of P lies on the convex part of the $(\mathcal{J}_1, \mathcal{J}_2)$ Pareto front, it can be recovered through scalarization. More precisely, the optimal expected payoff satisfies

$$P^a(\mathbf{x}) = \sup_{\lambda \in [0,1]} \left\{ B(\mathbf{x})e^{-v_1^\lambda(\mathbf{x})} - v_2^\lambda(\mathbf{x}) \right\} - R(\mathbf{x}),$$

where the optimal λ will be usually \mathbf{x} -dependent.

4.5 Numerical Methods

Since the domain Ω is meant to represent some protected area such as a national park, it will in general be irregularly shaped. One way of handling this would be to discretize Ω directly through the use of $\partial\Omega$ -conforming triangulated meshes. Efficient methods for numerically solving equation (4.16) on such meshes are already known [12, 108, 100]. For this paper, we will instead use a Cartesian grid \mathcal{X} to discretize a rectangular region in \mathbb{R}^2 containing Ω . Without loss of generality, suppose that Ω is contained within a rectangle $[0, X_{max}] \times [0, Y_{max}]$. Our grid spacing will be $\Delta x = X_{max}/N_x$ in the x -direction, and $\Delta y = Y_{max}/N_y$ in the y -direction. The coordinates of our gridpoints will be $x_i = i\Delta x$ for $i = 0, 1, \dots, N_x$ and $y_j = j\Delta y$ for $j = 0, 1, \dots, N_y$. For Model A, we will also discretize λ as $\lambda_k = k/N_\lambda$ for $k = 0, 1, \dots, N_\lambda$. For Model G, we will also discretize b as $b_m = \hat{B} + m(\check{B} - \hat{B})/N_b$, with $\hat{B} = \min_{\mathbf{x}} B(\mathbf{x})$, $\check{B} = \max_{\mathbf{x}} B(\mathbf{x})$, and $m = 0, 1, \dots, N_b$.

4.5.1 Scalarized Eikonal Equation (Model A)

We will approximate our value function with

$$U_{i,j}^k \approx u^{\lambda_k}(x_i, y_j)$$

and our λ -optimal-restricted value functions with

$$V_{i,j}^{k,1} \approx v_1^{\lambda_k}(x_i, y_j), \quad V_{i,j}^{k,2} \approx v_2^{\lambda_k}(x_i, y_j).$$

For each λ_k value, we will need to solve the scalarized Eikonal equation (4.16) and the coupled system of auxiliary linear PDEs (4.21) simultaneously. To approximate the gradients

of the value functions, we will make heavy use of the following first-order difference operators:

$$\begin{aligned} D_{i,j}^{+x}[W] &= \frac{W_{i+1,j} - W_{i,j}}{\Delta x}, & D_{i,j}^{-x}[W] &= \frac{W_{i,j} - W_{i-1,j}}{\Delta x}, \\ D_{i,j}^{+y}[W] &= \frac{W_{i,j+1} - W_{i,j}}{\Delta y}, & D_{i,j}^{-y}[W] &= \frac{W_{i,j} - W_{i,j-1}}{\Delta y}. \end{aligned}$$

where W could represent any of the value function approximations U^k , $V^{k,1}$, or $V^{k,2}$.

Since the scalarized Eikonal equation (4.16) and the auxiliary PDEs (4.21) are solved along the same characteristics, we will make sure that our approximations of the gradients are always using the same stencils. The upwind difference operator \mathcal{D}_{ij}^x in the x -direction will be:

$$\mathcal{D}_{i,j}^x[W] = \begin{cases} D_{i,j}^{+x}[W], & \text{if } D_{i,j}^{+x}[U^k] \leq \min(-D_{i,j}^{-x}[U^k], 0); \\ D_{i,j}^{-x}[W], & \text{if } -D_{i,j}^{-x}[U^k] < \min(D_{i,j}^{+x}[U^k], 0); \\ 0, & \text{otherwise;} \end{cases}$$

with $\mathcal{D}_{i,j}^y$ defined accordingly for the y -direction. Note that the stencil being used will always be determined by U^k regardless of whether \mathcal{D} is being applied to U^k , $V^{k,1}$, or $V^{k,2}$.

Using these first-order difference operators, we can write down the first-order upwind discretization of our system of PDEs as:

$$\begin{aligned} f_{i,j} \left[(\mathcal{D}_{i,j}^x[U^k])^2 + (\mathcal{D}_{i,j}^y[U^k])^2 \right]^{1/2} &= K_{i,j}^{\lambda_k} \\ \mathcal{D}_{i,j}^x[U^k] \cdot \mathcal{D}_{i,j}^x[V^{k,1}] + \mathcal{D}_{i,j}^y[U^k] \cdot \mathcal{D}_{i,j}^y[V^{k,1}] &= \frac{\psi_{i,j} K_{i,j}^{\lambda_k}}{f_{i,j}^2} \\ \mathcal{D}_{i,j}^x[U^k] \cdot \mathcal{D}_{i,j}^x[V^{k,2}] + \mathcal{D}_{i,j}^y[U^k] \cdot \mathcal{D}_{i,j}^y[V^{k,2}] &= \frac{K_{i,j} K_{i,j}^{\lambda_k}}{f_{i,j}^2} \end{aligned} \tag{4.22}$$

with the boundary conditions

$$U_{i,j}^k = V_{i,j}^{k,1} = V_{i,j}^{k,2} = 0, \quad \text{for } (x_i, y_j) \notin \Omega.$$

Since this is a monotone and consistent finite difference approximation of (4.16), U^k will converge under grid refinement to the viscosity solution u^{λ_k} [11]. But on any fixed grid,

finding U^k values is not straightforward since the above discretized system is coupled and nonlinear. On a grid with M gridpoints, straightforward Gauss-Jacobi iterations result in a high computational cost of $O(M^2)$, and Gauss-Seidel iterations with any fixed ordering of gridpoints are similarly inefficient.

A significant speed up is gained by exploiting the causal properties of upwind difference operators. Based on the definition of \mathcal{D}^x and \mathcal{D}^y , each $U_{i,j}^k$ depends only on the *smaller* neighboring U^k values. This *monotone causality* allows for the use of non-iterative Dijkstra-like “Fast Marching” methods [120, 109, 110], which compute U^k in $O(M \log M)$ operations. They mimic the logic of a classical shortest path algorithm on graphs [47] and effectively de-couple this discretized system by determining the correct ordering of gridpoints/equations on the fly.

Another popular approach for efficient Eikonal solvers is to use Gauss-Seidel iterations, but alternating through a list of direction-aligned grid orderings (from SW, SE, NE, and NW). Such “Fast Sweeping” techniques [19, 127, 119] aim to guess the direction of characteristics for a part of the domain, recovering many correct U^k values in each iteration. These methods converge in $O(\omega M)$ operations, but the factor ω is not known a priori and depends on the number of times the characteristics change direction from quadrant to quadrant. Both classes of methods have their preferred sets of problems. E.g., Fast Sweeping is clearly better when characteristics are largely straight lines, while Fast Marching is typically preferable on domains with complex geometry or with rapidly changing PDE coefficients. More recent hybrid (two-scale) methods aim to combine the best features of Marching and Sweeping. We refer to [31, 32] for a detailed description, benchmarking, and comparison with other sequential and parallel Eikonal solvers.

While our own implementation is based on Fast Marching, other methods mentioned above can be similarly adapted to the current application. (Since the directions of derivative

approximations for $V^{k,1}$ and $V^{k,2}$ are determined by U^k values, these two grid functions are also easily computed simultaneously with U^k .) Which method would be more efficient will depend on the geometry of Ω and the properties of ψ , K , and f .

Remark 7. The above grid functions are needed to predict the decisions of a rational extractor. For any single site $\mathbf{x} = (x_i, y_j)$, whether it will be extracted under Model A can be answered as soon as we find any k such that

$$B_{i,j}e^{-V_{i,j}^{k,1}} - V_{i,j}^{k,2} - R_{i,j} > \tilde{p}.$$

The exact $P^a(x_i, y_j)$ could be computed by any continuous optimization in λ . But to approximate P^a on the entire spatial grid \mathcal{X} , it is more efficient to compute $V^{k,1}$ and $V^{k,2}$ for all $k = 0, 1, \dots, N_\lambda$ and then use a λ -grid search:

$$P_{i,j}^a \approx \max_k \left\{ B_{i,j}e^{-V_{i,j}^{k,1}} - V_{i,j}^{k,2} \right\} - R_{i,j}. \quad (4.23)$$

4.5.2 Randomly-terminated Eikonal equation (Model G)

We will define \bar{U} as a grid function approximating the value function \bar{u} of section 4.3.2. The upwind difference operators can be similarly used to approximate the gradient and then used to obtain a consistent, monotone discretization of PDE (4.8) :

$$\begin{aligned} & [\min(D_{i,j}^{+x}[\bar{U}^m], -D_{i,j}^{-x}[\bar{U}^m], 0)]^2 + [\min(D_{i,j}^{+y}[\bar{U}^m], -D_{i,j}^{-y}[\bar{U}^m], 0)]^2 \\ & = [K_{i,j} + \psi_{ij}(b_m + R_{i,j} - \bar{U}_{i,j}^m)]^2 / f_{i,j}^2, \end{aligned} \quad (4.24)$$

with the boundary conditions $\bar{U}_{i,j}^m = 0$ for $(x_i, y_j) \notin \Omega$.

This is essentially the same discretization previously used in [6], where it was shown that its causal properties allow for a non-iterative Dijkstra-like algorithm with $O(M \log M)$ complexity. We refer readers to [6, Appendix B] for a quadrant-by-quadrant interpretations of (4.24) and a detailed discussion of efficient implementation. This approach is arguably

even better suited in the current context: the “free boundary” present in [6] made it necessary to use a special initialization, but our Dirichlet boundary conditions make it possible to use the original Fast Marching.

Remark 8. For any single site $\mathbf{x} = (x_i, y_j)$, the expected profit under Model G could be computed by using the exact $B_{i,j}$ instead of b_m in equation (4.24). But to approximate P^g on the entire spatial grid \mathcal{X} , it is more efficient to compute \bar{U}^m for all $m = 0, 1, \dots, N_b$. If m is such that $B_{i,j} \in (b_m, b_{m+1}]$, we can define

$$P_{i,j}^{g-} = B_{i,j} - \bar{U}_{i,j}^m - R_{i,j} \quad \text{and} \quad P_{i,j}^{g+} = B_{i,j} - \bar{U}_{i,j}^{m+1} - R_{i,j}.$$

Either one of these can be used as a good approximation since $P_{i,j}^g \in [P_{i,j}^{g-}, P_{i,j}^{g+}]$ and $P_{i,j}^{g+} - P_{i,j}^{g-} \leq (\check{B} - \hat{B})/N_b$. We know that (x_i, y_j) will be extracted if $P_{i,j}^{g-} > \tilde{p}$, and will not be extracted if $P_{i,j}^{g+} \leq \tilde{p}$. In our numerical tests we use the \tilde{p} level set of $\frac{1}{2}(P^{g-} + P^{g+})$ to approximate $\partial\Omega_p^g$. A more conservative discrete approximation of that boundary could be also obtained from the set $\{(x_i, y_j) \mid P_{i,j}^{g-} \leq \tilde{p} \leq P_{i,j}^{g+}\}$.

4.6 Examples

In all of our numerical examples, we use a homogeneous running cost $K(\mathbf{x}) = \kappa = 1$. As a result, the pre-extraction cost $R(\mathbf{x})$ coincides with $\tau(\mathbf{x})$, the minimum time required starting from $\partial\Omega$ to reach the extraction point \mathbf{x} . Similarly, $\mathcal{J}_2(\mathbf{x}, \mathbf{a}(\cdot))$ represents the time to $\partial\Omega$ starting from \mathbf{x} and using the post-extraction control $\mathbf{a}(\cdot)$. In Examples 1-5 the perpetrators can move with speed $f = 1$ (and thus, $\tau(\mathbf{x}) = d(\mathbf{x})$), while in Examples 6 and 7, their speed is terrain dependent.

To save space, we primarily focus on the expected profit P^a corresponding to Model A (the aerial patrols). Our Figure captions report the maximum attainable expected profit $\bar{P}^a = \max_{\mathbf{x} \in \Omega} P^a(\mathbf{x})$, the proportion of protected area $A_p^a = |\Omega_p^a|/|\Omega|$, and the proportion of

protected value $V_p^a = \int_{\Omega_p^a} B(\mathbf{x})d\mathbf{x} / \int_{\Omega} B(\mathbf{x})d\mathbf{x}$. Implementing the same patrol density ψ in both Model A and Model G (the ground patrols) might be hard in practice, but for the sake of quantitative comparison, we assume that this can be done. Since in Model G the costs resulting from a detection are typically lower, this results in $P^g \geq P^a$ and $\Omega_p^g \subset \Omega_p^a$. We demonstrate this in Examples 5-7, where we compare $\partial\Omega_p^a$ with $\partial\Omega_p^g$ (using $N_\lambda = N_b$) and provide the related statistics.

In most cases we consider ψ to be a part of the problem statement, but in Examples 3 and 4 we also perform a straightforward optimization of ψ to maximize A_p^a while respecting the specified patrol density budget E , using (4.1) with $\gamma = 1$.

Our numerical approach is quite efficient. Even though the implementation is not highly optimized, it was easy to perform all simulations on a mid-range laptop computer (MacBook Pro with a 2GHz Intel Core i5 processor and 8GB of RAM). For a fixed ψ and any specific λ , the execution times varied from 120 milliseconds on smaller grids (Examples 3-4) to 50 seconds on larger grids (Examples 6-7). The overall time depended on the desired λ -resolution. The source code for our implementation will be made available at <https://github.com/eikonal-equation/Environmental-Crime>.

4.6.1 Example 1: a circle

We start with a radially symmetric example close to the original setting of Albers [3]. The domain Ω is a disk of diameter 1, and we use a constant benefit function $B = 2$. But we assume a more general pointwise detection rate based on the distance d from the boundary:

$$\psi(d) = \frac{\mu}{50(d - 0.3)^2 + 0.5}, \quad d \in [0, 0.5], \quad (4.25)$$

where μ is a constant chosen to enforce the budget constraint $\int_{\Omega} \psi(\mathbf{x})d\mathbf{x} = E$ with $E = 2.5$. A contour plot of ψ can be found in Figure 4.2b. The expected profit P^a is shown in Figure

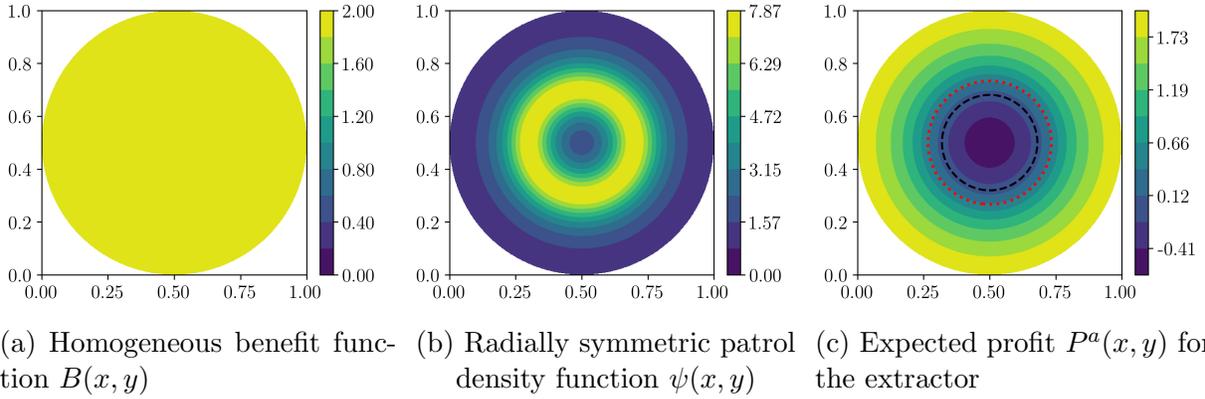


Figure 4.2: A disk-like domain. The maximum profit $\bar{P}^a = 2$ is attained at $\partial\Omega$, the pristine area proportion is $A_p^a = 13.02\%$ and, since $B(\mathbf{x})$ is constant, the same $V_p^a = 13.02\%$ of the value is protected. If $P_{\#}$ was used instead of P^a , it would overestimate the pristine proportion and value protected as $A_p^a = V_p^a = 21.77\%$.

4.2c, with the dashed line indicating the boundary of the pristine region Ω_p . The dotted line indicates the zero level set of $P_{\#}$, the approximation to P^a discussed in Remark 6.

We use a uniform rectangular grid with dimensions $N_x = 501$, $N_y = 501$. Since f is uniformly one, with ψ and K depending only on the distance from the boundary, this example falls into the category of “ τ -banded” scenarios discussed in Section 4.4. The optimal trajectories are simply straight lines to the closest point on the boundary. Since the same trajectory minimizes both \mathcal{J}_1 and \mathcal{J}_2 , the Pareto Front consists of a single point, which can be recovered using any λ with a single PDE solve.

4.6.2 Example 2: banded patrol density on a square

We use a sum of two Gaussians to specify the benefit function

$$B(x, y) = e^{-10((x-0.25)^2+(y-0.5)^2)} + e^{-10((x-0.75)^2+(y-0.5)^2)}$$

on a unit square Ω ; see Figure 4.3a. We assume the same “banded” detection rate defined in (4.25), but with μ selected based on $E = 2$. A contour plot of ψ is shown in Figure 4.3b.

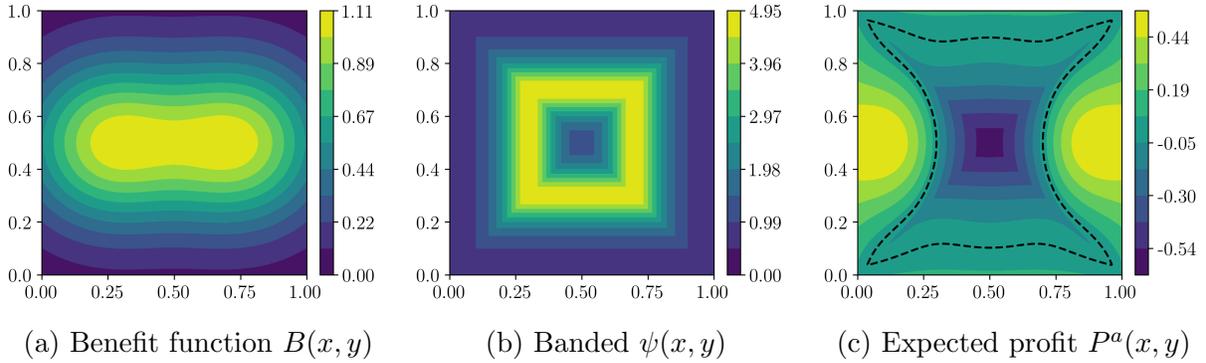


Figure 4.3: Banded patrol density on a square. $\bar{P}^a = 0.56$, $A_p^a = 45.67\%$, and $V_p^a = 52.99\%$.

The expected profit P^a can be found in Figure 4.3c, with the dashed line showing $\partial\Omega_p^a$.

This example was computed on a uniform rectangular grid with dimensions $N_x = 501$, $N_y = 501$. As in Example 1, this is another case of a “ τ -banded” patrol density, so optimal trajectories are straight lines to the closest point on $\partial\Omega$, and any single u^λ is sufficient to compute $P^a(\mathbf{x})$. (Note that this property holds regardless of a particular benefit function B . It also implies that both Models A and G will yield exactly the same predictions.)

4.6.3 Example 3: placing a patrol station

As noted in [3], in realistic situations PAMs face additional constraints and the detection rate typically decreases with the distance to patrol stations (in our case – the locations where the surveillance drones are based). We will suppose that the detection rate $\psi(x, y)$ is now a single Gaussian “centered” at a patrol station $(\bar{x}, \bar{y}) \in \Omega$:

$$\psi_{\bar{x}, \bar{y}}(x, y) = \mu e^{-30((x-\bar{x})^2 + (y-\bar{y})^2)}$$

with μ selected based on $E = 2$. Using the same domain, speed, and benefit functions defined in Example 2, we will now search for the patrol station location that maximizes the pristine area proportion A_p^a . In principle, this could be done with any black-box derivative-

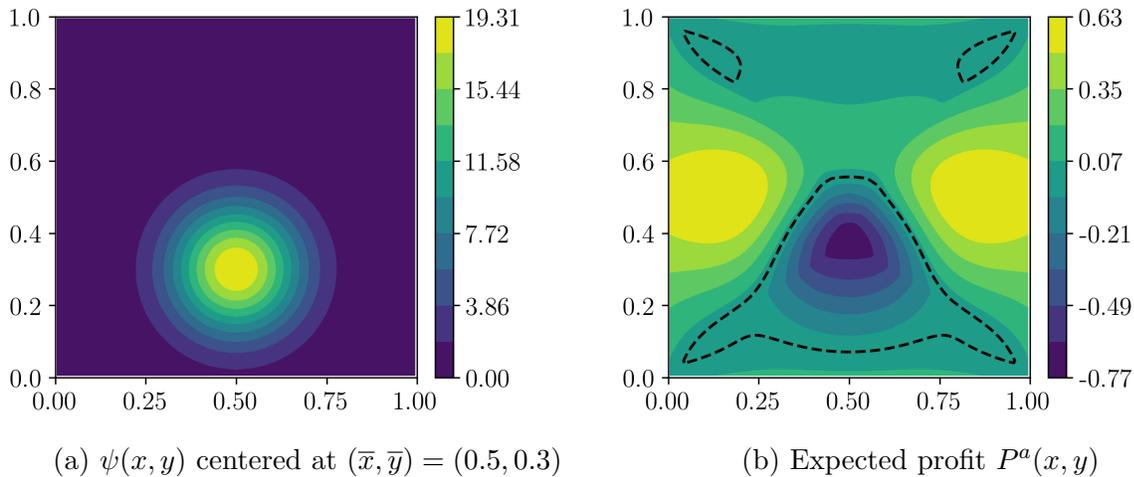


Figure 4.4: Optimal patrol station position. $\bar{P}^a = 0.63$, $A_p^a = 23.53\%$, and $V_p^a = 24.54\%$

free optimization method on Ω , but for the sake of simplicity we have implemented a grid search on a coarse (11×11) grid. For each candidate location, the pristine region Ω_p is computed on a uniform rectangular grid with $N_x = 201$, $N_y = 201$, and $N_\lambda = 101$.

Due to the symmetry, there are actually two A_p^a -maximizing locations: $(\bar{x}, \bar{y}) = (0.5, 0.3)$ and $(\bar{x}, \bar{y}) = (0.5, 0.7)$. Figure 4.4a shows ψ centered at the former. The corresponding expected profit P^a can be found in Figure 4.4b, with the dashed line showing $\partial\Omega_p^a$.

4.6.4 Example 4: allocating surveillance resources

Given the non-unique maxima (\bar{x}, \bar{y}) in Example 3, we now suppose that PAMs decide to build a separate drone station at each of these two locations. The next question is how to share the surveillance resources between them. Assuming that the overall number of drones is fixed and that the detection rate generated by each station is proportional to the number of drones stationed there, any allocation corresponds to choosing $\mathbf{w} = (w_1, w_2)$ with

$w_1, w_2 \geq 0$, $w_1 + w_2 = 1$, and defining

$$\begin{aligned}\psi_{\mathbf{w}}(x, y) &= \mu_{\mathbf{w}} (w_1 G_1(x, y) + w_2 G_2(x, y)), \\ G_1(x, y) &= e^{-30((x-0.5)^2 + (y-0.3)^2)}, \\ G_2(x, y) &= e^{-30((x-0.5)^2 + (y-0.7)^2)},\end{aligned}$$

with $\mu_{\mathbf{w}}$ chosen to ensure $\int_{\Omega} \psi_{\mathbf{w}}(x, y) dx dy = E = 2$. For each \mathbf{w} , we compute the A_p^a on a uniform rectangular grid with $N_x = 201$, $N_y = 201$, and $N_{\lambda} = 101$. A grid search over 101 values of w_1 is used to find the A_p^a -maximizing allocation. As in the previous example, the answer is not unique: both $\mathbf{w} = (0.43, 0.57)$ and $\mathbf{w} = (0.57, 0.43)$ are optimal.

For the first of these, Figure 4.5a shows a contour plot of $\psi_{\mathbf{w}}$ along with four different Pareto-optimal paths starting from an extraction point $\mathbf{x}_0 = (0.555, 0.315)$. The path maximizing the expected profit P^a is shown in magenta, the path minimizing the probability of detection Ψ is shown in green, and the path minimizing the time of travel is shown in black. The red path corresponds to the “nearly optimal” $\lambda_{\sharp} = \frac{B}{B+1}$ of the linearized- Ψ problem, as discussed in Remark 6. The corresponding expected profit P^a can be found in Figure 4.5b, with the dashed line showing $\partial\Omega_p^a$.

In Figure 4.6, we consider the λ -optimization problem (4.23) for the same extraction point \mathbf{x}_0 in detail. Figure 4.6a shows the λ_k -optimal payoffs $B(x, y)e^{-V_{i,j}^{k,1}} - V_{i,j}^{k,2} - R(x, y)$ in blue, with the maximum profit $P^a(x, y)$ as a black dashed line. The choice of path based on λ_{\sharp} is clearly far from optimal, which is not surprising since \mathcal{J}_1 is not small. The linearized- Ψ approach significantly overestimates the probability of detection along all trajectories. In fact, if we were to base the estimated “profit” on $u^{\lambda_{\sharp}}$, the formula (4.17) would yield an even more pessimistic $P_{\sharp}(\mathbf{x}_0) = -0.699$.

Figure 4.6b shows the $(\mathcal{J}_1, \mathcal{J}_2)$ Pareto Front for the same \mathbf{x}_0 . The color of points in in 4.6a and 4.6b matches the color of the corresponding paths in 4.5a.

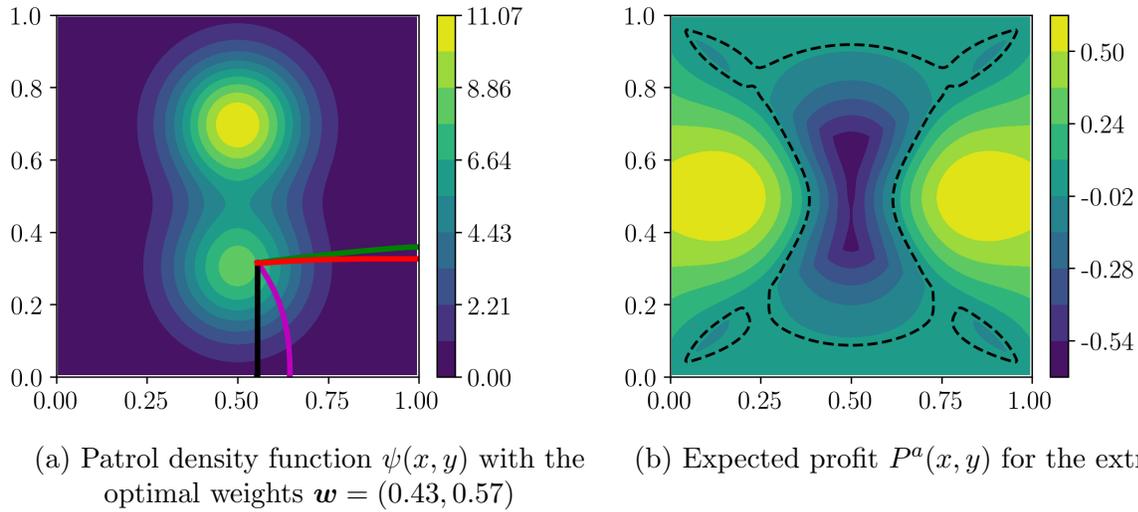


Figure 4.5: Allocating surveillance resources. $\bar{P}^a = 0.63$, A_p^a has increased to 35.26%, and V_p^a has increased to 38.48%. We note that from the PAMs' perspective, the banded patrol from Example 2 still performs better while using the same budget ($E = 2$).

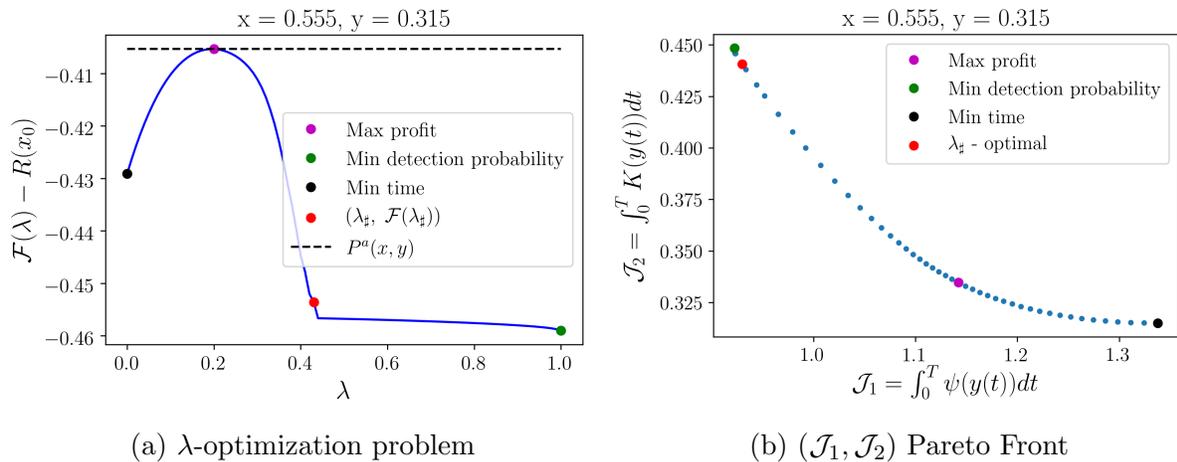


Figure 4.6: Lambda Optimization and Pareto Front at $\mathbf{x}_0 = (0.555, 0.315)$ in Example 4.

4.6.5 Example 5: exploiting patrol gaps

The main purpose of this example is to highlight the differences in post-extraction trajectories between Models A and G. The domain Ω is again the unit square $[0, 1] \times [0, 1]$, with a uniform speed of $f = 1$. The benefit function B is

$$B(x, y) = 3 + 7.5 \exp \left[-10 \left((x - 0.7)^2 + (y - 0.5)^2 \right) \right]$$

and can be seen in Figure 4.7a. The patrol density $\psi(x, y)$ is a sum of eight Gaussians with different means and standard deviations and can be seen in Figure 4.7b.

In Figure 4.7c we see the expected profit P^a from Model A computed on a uniform rectangular grid with $N_x = N_y = 501$, and $N_\lambda = 401$. The optimal post-extraction trajectory corresponding to $\hat{\mathbf{x}} = (0.824, 0.58)$ is shown in magenta, and $\partial\Omega_p^a$ is shown in orange.

In Figure 4.7d we see the expected profit P^g from Model G computed on the same grid, and $\partial\Omega_p^g$ is shown in orange. While P^g is only slightly larger than P^a , a significant difference can be seen in the post-extraction trajectories. The solid magenta line represents the optimal post-extraction (pre-detection) trajectory starting from $\hat{\mathbf{x}}$. The dotted magenta lines show possible post-detection trajectories from three different positions. Since $f = 1$, every post-detection trajectory is just a straight line to the closest point on $\partial\Omega$.

4.6.6 Example 6: Yosemite National Park

Here we use the domain and terrain data from Yosemite National Park. To facilitate the comparison, we use the same B , ψ , and f defined in [8].

The benefit function shown in Figure 4.8a is $B(d) = 8d(2d_m - d)/(d_m)$, where $d_m =$

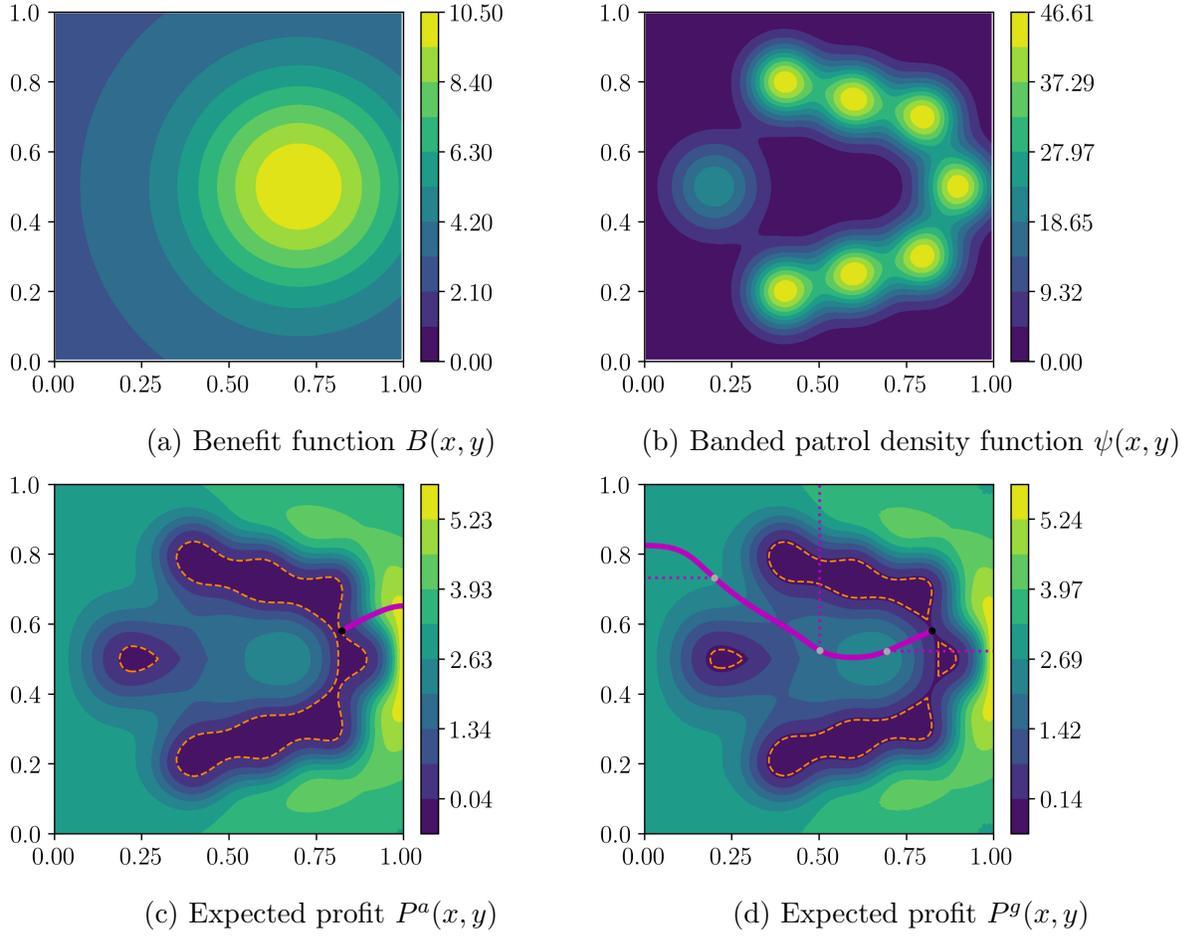


Figure 4.7: Differences between path-planning in Model A ($\bar{P}^a = 5.87$, $A_p^a = 11.06\%$, $V_p^a = 14.66\%$.) and Model G ($\bar{P}^g = 5.88$, $A_p^g = 8.63\%$, $V_p^g = 11.02\%$).

$\max_{\mathbf{x} \in \Omega} d(\mathbf{x})$. The banded patrol density shown in Figure 4.8b is defined by

$$\psi(d) = \begin{cases} \frac{\mu(0.7d_m - d)}{d_m}, & 0.3d_m < d < 0.7d_m \\ 0, & \text{otherwise} \end{cases}$$

where μ is selected based on $E = 3 \times 10^4$.

The extractors' isotropic speed of motion is shown in Figure 4.8c and depends on the slope of the elevation map $z(\mathbf{x})$. Namely, $s = |\nabla z(\mathbf{x})|$ is the grade in the direction of $\nabla z(\mathbf{x})$

and

$$f(s) = 1.11 \exp\left(-\frac{(100s + 2)^2}{2345}\right). \quad (4.26)$$

This example was computed on a uniform rectangular grid with $N_x = 5406$, $N_y = 4325$, and $N_\lambda = 21$. The expected profit P^a can be seen in Figure 4.8d. The red dashed line corresponds to the boundary of the “high value region”

$$\Omega_e = \left\{ \bar{\mathbf{x}} \in \Omega \mid P^a(\bar{\mathbf{x}}) + R(\bar{\mathbf{x}}) \geq (1 - \varepsilon) \max_{\mathbf{x} \in \Omega} (P^a(\mathbf{x}) + R(\mathbf{x})) \right\}, \quad (4.27)$$

with $\varepsilon = 0.85$. This definition is chosen for the sake of direct comparison with Figure 9 in [8] since that paper does not consider the cost of pre-extraction trajectories and their definition of pristine region is also different.

Under our interpretation of the extraction criteria, the extractors will target more than 98% of the park, and the patrol budget E is clearly insufficient. We repeat this experiment with a higher $E = 2 \times 10^5$; see Figure 4.9.

This example presents a non-trivial Pareto Front since the detection rate ψ is d -banded (rather than τ -banded) and the time-optimal paths are generally not detection-minimizing. As a result, the predicted profit in Model A is different from Model G, but the differences are not very large here; e.g., $A_p^a - A_p^g \approx 2.16\%$ even with the higher patrol budget used in Figure 4.9.

It is also worth noting that the definition (4.26) results in very small f (and occasionally even zero up to machine precision) on all steep parts of Ω . As a result, the cost of targeting those points becomes extremely high even if everything around them is extracted. This explains the “noisy” look of $\partial\Omega_e$ in Figure 4.8d and $\partial\Omega_p$ in Figure 4.9. Such un-extractable points are not identified in Figure 9b of [8] since they are masked by the b -interpolation procedure used in that paper.

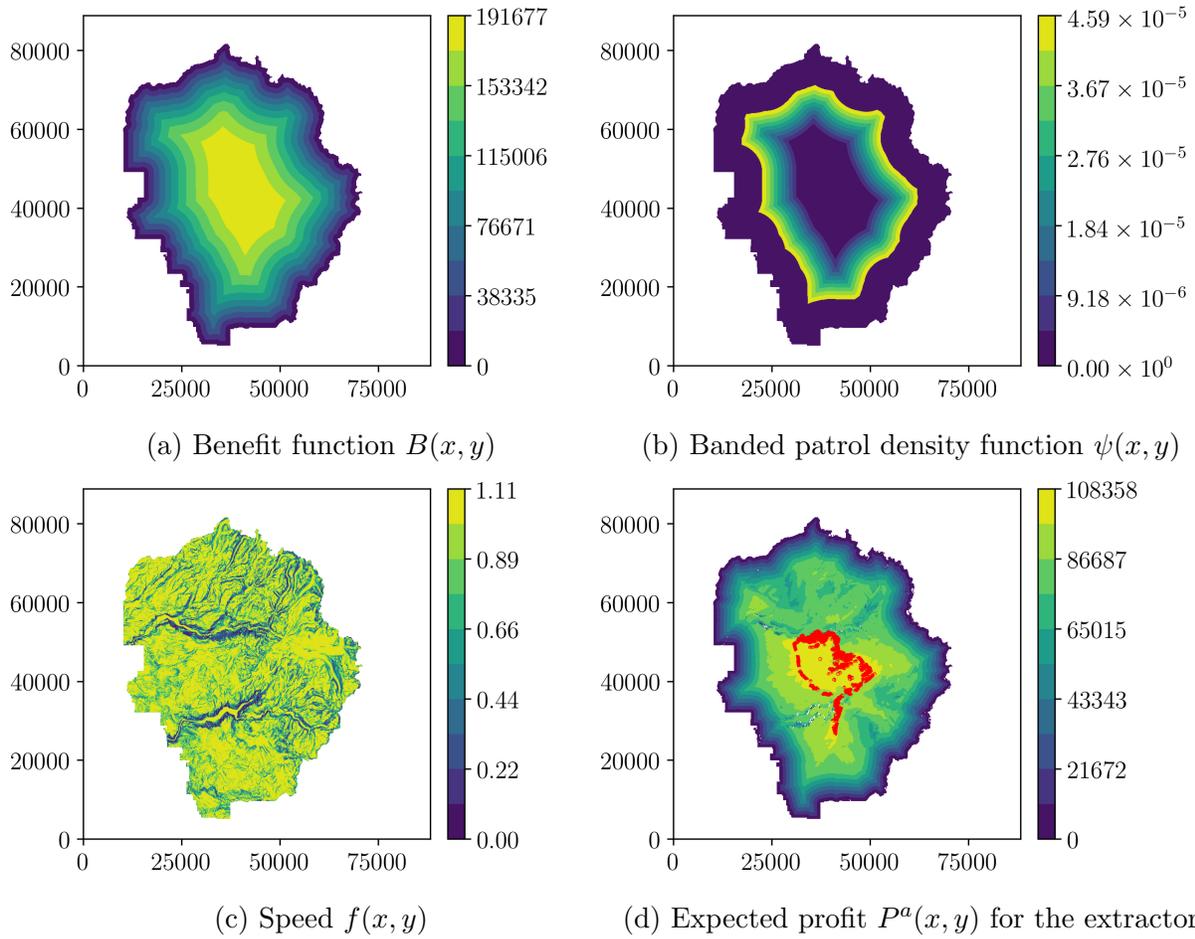


Figure 4.8: Yosemite National Park with $E = 3 \times 10^4$, corresponding to Figure 9b in [8]. $\bar{P}^a = 108358$, $A_p^a = 1.90\%$, and $V_p^a = 1.86\%$. Dashed red line indicates $\partial\Omega_e$.

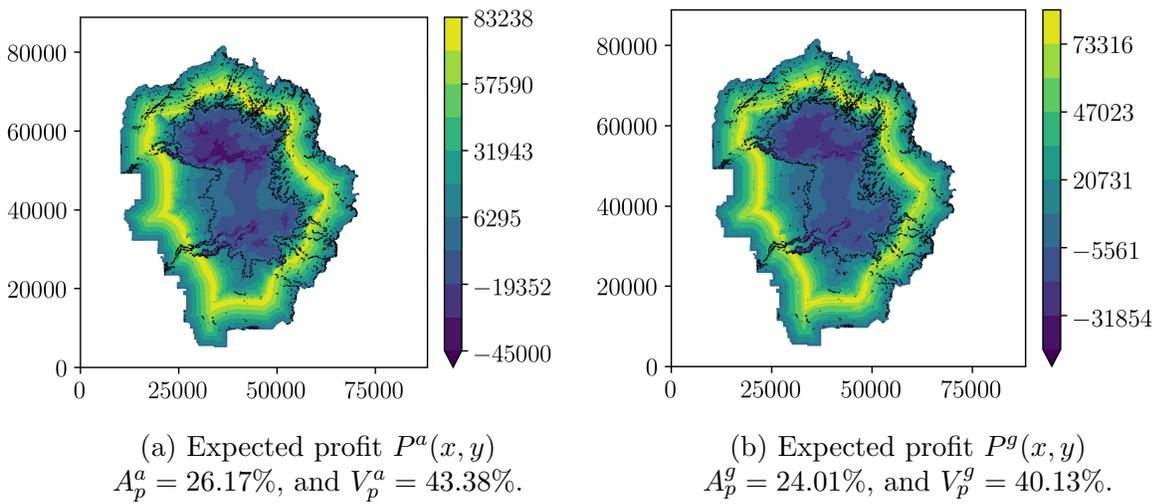


Figure 4.9: Yosemite National Park with $E = 2 \times 10^5$. Black line indicates $\partial\Omega_p$.

4.6.7 Example 7: Kangaroo Island

We use the domain and terrain data corresponding to Kangaroo Island with the same B , ψ , and f considered in [8].

The functions B and f are defined exactly as in Example 6; see Figures 4.10a and 4.10c. The detection rate ψ is a positive constant from $x = 30,000$ to $x = 90,000$ and zero elsewhere, scaled to match the budget $E = 3 \times 10^4$; see Figure 4.10b.

This example was computed on a uniform rectangular grid with $N_x = 7159$, $N_y = 3111$, and $N_\lambda = 21$. The expected profit P^a can be seen in Figure 4.10d. The red dashed line corresponds to the boundary of the “high value region” defined in (4.27) and can be compared with Figure 12b in [8]. Again, almost the entire Ω is targeted by the extractors. Since the terrain is mostly flat, this remains true even with a much higher budget $E = 2.5 \times 10^5$; see Figure 4.11a.

In Figure 4.11b we examine Model G with the same parameters. Since a detection is much less costly in Model G, the extractors are even more willing to enter the patrolled area, and the pristine area (shown in orange) is about two times smaller than in Figure 4.11a.

4.7 Conclusions

We have presented two control-theoretic models of illegal resource extraction counteracting the detection efforts of Protected Area Managers (PAMs). The perpetrators are assumed to choose potential extraction sites based on their expected profit. They take into account the distribution of resources, the geometry and terrain of the protected area Ω , and the localized detection rate $\psi : \Omega \rightarrow \mathbb{R}_{+,0}$ due to PAMs’ patrol activities. The cost of extracting at $\mathbf{x} \in \Omega$

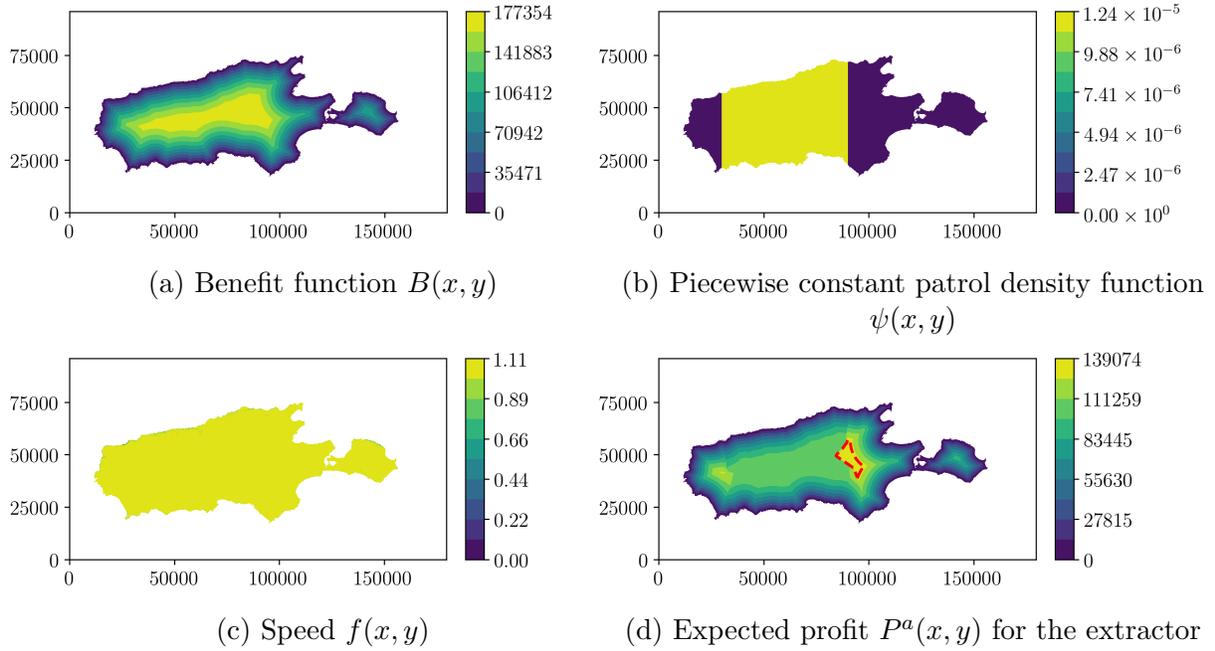


Figure 4.10: Kangaroo Island with $E = 3 \times 10^4$; corresponding to Figure 9b in [8]. $\bar{P}^a = 139074$, $A_p^a = 0.16\%$, and $V_p^a = 0.0042\%$. Dashed red line indicates $\partial\Omega_e$.

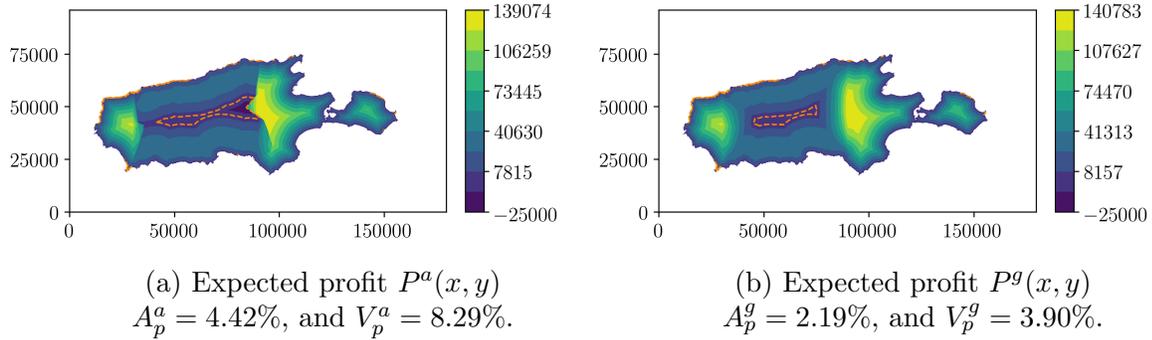


Figure 4.11: Kangaroo Island with $E = 2.5 \times 10^5$. Dashed orange line indicates $\partial\Omega_p$.

is based on a roundtrip $\partial\Omega \rightarrow \mathbf{x} \rightarrow \partial\Omega$. The two portions of the trip usually follow different paths. The $\partial\Omega \rightarrow \mathbf{x}$ path is selected to minimize the time and hardship of travel, while on the way back ($\mathbf{x} \rightarrow \partial\Omega$) these are balanced against the expected losses resulting from a possible detection by PAMs.

A detection is always followed by confiscation of resources carried by the extractor. But the details of how and when a confiscation happens significantly influence the perpetra-

tors’ decision making process. Our “Model G” assumes that the extractors are spotted by ground patrols with an immediate capture/confiscation after their detection. It is further assumed that after the confiscation the extractors switch to the quickest path to leave the protected area. We find their best pre-detection trajectories by recasting this as an optimal control of a suitable randomly-terminated process. Our “Model A” assumes that the extractors are spotted by aerial patrols but remain oblivious when this happens, continuing along their originally chosen trajectory until they are captured at $\partial\Omega$. The key challenge of this application is the impossibility of writing a single running cost to be integrated along the post-extraction path unless we resort to a linearizing approximation for the probability of detection (see Remark 3) or agree to increase the dimension of our planning space (see Remark 4). We show that the former often leads to very inaccurate predictions while the latter leads to expensive numerical methods. Instead, we opt to use the techniques of *multiobjective* dynamic programming on the original domain Ω , resulting in an accurate and efficient numerical implementation. Our method is illustrated on a range of examples, with two of them based on real terrain maps from Yosemite National Park in California and Kangaroo Island in South Australia. In each case, we compute the perpetrators’ expected profit from all possible extraction sites as well as the “pristine” region $\Omega_p \subset \Omega$ not affected by extraction. We also show that Model G generally results in higher expected profits, smaller Ω_p , and different post-extraction trajectories.

Many of our modeling assumptions were made primarily for the simplicity of exposition and could be easily relaxed in the future. For example, it would not be hard to incorporate extractor’s preferences for starting and terminal positions on $\partial\Omega$. One could also easily model the time needed to extract at each site and the resulting increase in the risk of detection. It would be similarly trivial to include additional fines in case of detection/capture. Following [8], we have assumed that the extractors’ speed of motion f is isotropic. I.e., if $z(\mathbf{x})$ is the elevation map, the speed f was based on the grade in the direction of $\nabla z(\mathbf{x})$ rather than

on the grade in the chosen direction of motion \mathbf{a} . In reality, a pedestrian’s speed should depend on both, particularly for moderate $|\nabla z|$. Any such model would replace Eikonal PDEs with anisotropic HJB equations. Efficient numerical methods for the latter have also been developed both in Fast Marching [108, 4, 92] and Fast Sweeping [119, 78] frameworks.

While our current implementation is sequential, we note that it would be easy to build a parallel algorithm with excellent scalability for computing $P(\mathbf{x})$: for each specific λ_k or b_m , the value function could be computed by a separate CPU core. The resulting speed-up will be very useful in optimizing the choice of detection rate ψ to maximize the area of Ω_p while staying within PAMs’ enforcement/surveillance budget. In the current paper we perform some of this by straightforward grid searches, but a more general ψ -optimization will require higher computational efficiency. The problem of optimally allocating surveillance resources across several patrol stations (e.g., Example 4) will become particularly expensive as the number of stations grows. We believe that an efficient solution will be based on generalizing our recent work on surveillance-evasion games [56, 27].

We conclude by listing two significant challenges not tackled in the current paper. Firstly, a “single period” assumption for this Stackelberg game is rather limiting. As extractors continue degrading the protected area Ω , the distribution of resources changes and PAMs should accordingly re-allocate the surveillance resources, adjusting ψ to increase the pristine region Ω_p . Secondly, in practice the detection rate ψ is not chosen directly but results from implementing specific surveillance strategies (e.g., choices of particular patrol trajectories and patrol frequencies). It would be far more realistic and more difficult to optimize those strategies directly.

**UNCERTAINTY QUANTIFICATION AND MANAGEMENT IN
PIECEWISE DETERMINISTIC MARKOV PROCESSES**

5.1 Introduction

Piecewise-deterministic Markov processes (PDMP) provide a powerful formalism for modeling situations with discrete random change in the global environment. That formalism is particularly useful when the number of *deterministic modes* of the global environment is relatively small and there is a high fidelity statistical characterization of mode-to-mode *switching* rates. Such processes arise in a broad range of applications, especially in the biological sciences [104]. For example, they can be used to model keratin network formation [15], SIRS epidemic spread [89], and genetic networks [105]. In other disciplines, applications of PDMPs include fatigue crack growth [33] and financial contagion [43], manufacturing processes [2, 18, 96, 107], sustainable development, economic growth & climate change [65, 64], and path-planning under uncertainty [6, 113, 30].

In this paper we focus on a computational framework for quantifying the uncertainty in outcomes of PDMPs inherent due to the random switching times and possible uncertainty in switching rates. If the PDMP system is controlled in real time, we also show that this uncertainty of outcomes can be actively *managed*.

In PDMPs, the full state of the system is described by a continuous component $\mathbf{x} \in \Omega \subset \mathbb{R}^d$ and a discrete component $i \in \mathcal{M} = \{1, \dots, M\}$ that represents the current deterministic “mode”. Starting from the initial configuration (\mathbf{x}, i) , the evolution of continuous component

$\mathbf{y}(t)$ is defined by a (mode-dependent) ODE

$$\begin{aligned} \mathbf{y}'(t) &= \mathbf{f}(\mathbf{y}(t), m(t)) = \mathbf{f}_{m(t)}(\mathbf{y}(t)), \\ \mathbf{y}(0) &= \mathbf{x} \in \Omega, \\ m(0) &= i \in \mathcal{M}, \end{aligned} \tag{5.1}$$

while the switches in mode $m(t)$ are based on a continuous-time Markov process on \mathcal{M} .

Using λ_{ij} to denote the rate of $(i \rightarrow j)$ switching, we can write

$$\lim_{\tau \rightarrow 0} \frac{\mathbb{P}(m(t+\tau) = j \mid m(t) = i)}{\tau} = \lambda_{ij}, \quad \forall t \geq 0, i \in \mathcal{M}, j \in \mathcal{M} \setminus \{i\}. \tag{5.2}$$

Here, we focus on *exit-time* problems, in which the process stops as soon as the system reaches a compact exit set $Q \subset \Omega$. Due to the random mode-switches, the exit-time $T_{\mathbf{x},i} = \min\{t > 0 \mid \mathbf{y}(t) \in Q\}$ is also random, which makes it somewhat harder to approximate the distribution for our main object of study – the *cumulative cost* of the PDMP $\mathcal{J}(\mathbf{x}, i)$.

In addition to mode-dependent dynamics $\mathbf{f} : \Omega \times \mathcal{M} \rightarrow \mathbb{R}^d$, we also include a mode-dependent *running cost* $C : \Omega \times \mathcal{M} \rightarrow \mathbb{R}^+$ and *exit cost* $q : Q \times \mathcal{M} \rightarrow \mathbb{R}^{+,0}$. To simplify the notation, we will also sometimes use the mode as a subscript:

$$C_i(\mathbf{x}) = C(\mathbf{x}, i), \quad \mathbf{f}_i(\mathbf{x}) = \mathbf{f}(\mathbf{x}, i), \quad q_i(\mathbf{x}) = q(\mathbf{x}, i), \quad \text{etc.}$$

We will assume that q_i 's are continuous in \mathbf{x} , while C_i 's and \mathbf{f}_i 's are bounded and piecewise Lipschitz continuous. The cumulative cost is then formally defined as

$$\mathcal{J}_i(\mathbf{x}) = \mathcal{J}(\mathbf{x}, i) = \int_0^{T_{\mathbf{x},i}} C(\mathbf{y}(t), m(t)) dt + q(\mathbf{y}(T_{\mathbf{x},i}), m(T_{\mathbf{x},i})). \tag{5.3}$$

We will generally assume that Ω is a closed set and the process can continue on $\partial\Omega \setminus Q$, but if the dynamics forces us to leave Ω before reaching Q , this will result in $\mathcal{J} = +\infty$. We note that the notion of cumulative cost is much more common in *controlled* PDMPs, where it is used to select criteria for control optimization. But we also consider \mathcal{J} in this simpler uncontrolled case to focus on a single measurable outcome of the process.

We develop our approach in this general setting, but our numerical experiments highlight that studying \mathcal{J} is far from trivial even if $C \equiv 1$, $q \equiv 0$, and $Q = \partial\Omega$, yielding $\mathcal{J}(\mathbf{x}, i) = T_{\mathbf{x}, i}$, the time until we reach the boundary. For a motivating example, consider a “sailboat” traveling with unit speed on an interval $\Omega = [0, 1]$ and subject to random mode/wind switches. We will assume that it is moving rightward in mode 1 and leftward in mode 2, the time intervals between mode switches are independent exponentially distributed random variables with rate λ , and the process terminates as soon as the boat reaches $Q = \{0, 1\}$.

Throughout the paper, we take an unapologetically exploratory approach, focusing on derivation of equations and numerical methods as well as instructive test problems rather than proofs of convergence or realistic applications. In section 5.2, we explain how the CDF for \mathcal{J} can be computed by solving a system of coupled linear PDEs. Our equations can be interpreted as a PDMP-adapted version of the Kolmogorov Backward Equation generalized to handle arbitrary running costs rather than just time. Another related approach is the previous development of numerical methods for the Liouville-Master Equation in [7]. We also derive simpler recursive difference equations to compute the CDF for a discrete analog of our setting – a random route-switching process on a graph.

In most real world applications, all switching rates (λ_{ij}) will be known only approximately and it is necessary to bound the results of this modeling uncertainty. In section 5.3, we show how bounds on these switching rates can be used to bound the CDF of \mathcal{J} . Interestingly, it turns out that it is easier to compute tight bounds if the switching rates are not assumed to be constant in time.

In many applications, the focus is on optimally *controlling* PDMP processes (affecting the dynamics in each deterministic mode), with the notion of optimality typically based on the average-case outcomes (e.g., minimizing the expected total cost). Once a control is fixed, the same uncertainty quantification tools covered in sections 5.2 and 5.3 become

relevant. Moreover, the control can also be selected to *manage* the uncertainty, providing some robustness guarantees or minimizing the probability of undesirable outcomes. Following the latter idea, we introduce a method for optimizing the CDF of controlled PDMP models in section 5.4. We conclude by discussing further extensions and limitations of our approach in section 5.5.

5.2 Computing the CDF

Before discussing the methods for approximating the CDF for the randomly switching process described in section 5.1, we first consider the same challenge for Markov-style switching on a graph in section 5.2.1, then switching to the continuous version in section 5.2.2. Numerical methods for the latter are then described in section 5.2.3 and illustrated on examples in section 5.2.4.

5.2.1 Discrete PDMPs

We start by reviewing a simple model of deterministic routing on a directed graph with a node set $X = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ and a target set $Q \subset X$. We will assume that $K : X \times X \rightarrow (0, +\infty]$ specifies the known cost of possible “steps” (i.e., node-to-node transitions) with $K(\mathbf{x}, \mathbf{x}') = +\infty$ iff $(\mathbf{x}, \mathbf{x}') \notin E$. A *route* on this graph can be specified in *feedback form* by a mapping $F : X \rightarrow X$ such that $(\mathbf{x}, F(\mathbf{x})) \in E \forall \mathbf{x} \in X$. Given a starting position $\mathbf{y}_0 = \mathbf{x} \in X$, a path can be defined by a sequence $\mathbf{y}_{n+1} = F(\mathbf{y}_n)$, terminating as soon as $\mathbf{y}_n \in Q$. We will further assume that the terminal cost charged at that point is specified by $q : Q \rightarrow [0, +\infty)$. If the path enters Q after $\bar{n}(\mathbf{x})$ steps, its cumulative cost can be expressed

as

$$\mathcal{J}(\mathbf{x}) = \sum_{n=0}^{\bar{n}(\mathbf{x})-1} K(\mathbf{y}_n, \mathbf{y}_{n+1}) + q(\mathbf{y}_{\bar{n}(\mathbf{x})}),$$

with $\mathcal{J}(\mathbf{x}) = +\infty$ if the path remains forever in $X \setminus Q$. The recursive relationship among \mathcal{J} values makes it easy to recover all of them by solving a linear system

$$\begin{aligned} \mathcal{J}(\mathbf{x}) &= K(\mathbf{x}, F(\mathbf{x})) + \mathcal{J}(F(\mathbf{x})), & \forall \mathbf{x} \in X \setminus Q; \\ \mathcal{J}(\mathbf{x}) &= q(\mathbf{x}), & \forall \mathbf{x} \in Q. \end{aligned} \quad (5.4)$$

We will now consider a version of the problem with a total of M different routes F_1, \dots, F_M , each of them with its own pair of running and terminal costs (K_i, q_i) defined on the same graph. To simplify the notation, we will use $K_i(\mathbf{x})$ as a shorthand for $K_i(\mathbf{x}, F_i(\mathbf{x}))$. We define a random route-switching process by assuming that there is a chance of switching to another route after each transition. I.e., if the current route/mode is F_i , the probability p_{ij} of switching to F_j after the next transition is known a priori for all $i, j \in \mathcal{M} = \{1, \dots, M\}$. The number of transitions is now a random variable, along with the cost paid for each future transition. In defining the new random cumulative cost $\mathcal{J}_i(\mathbf{x})$, we note that the subscript only encodes the initial route used in the first transition as we depart from \mathbf{x} . It is easy to see that $u_i(\mathbf{x}) = \mathbb{E}[\mathcal{J}_i(\mathbf{x})]$ should satisfy a recursive relation

$$\begin{aligned} u_i(\mathbf{x}) &= K_i(\mathbf{x}) + \sum_{j=1}^M p_{ij} u_j(F_i(\mathbf{x})), & \forall \mathbf{x} \in X \setminus Q, i \in \mathcal{M}; \\ u_i(\mathbf{x}) &= q_i(\mathbf{x}), & \forall \mathbf{x} \in Q, i \in \mathcal{M}. \end{aligned} \quad (5.5)$$

It is worth noting that this system of MN linear equations lacks the nice causal property that we enjoyed in the deterministic case. Assuming that $\mathcal{J}(\mathbf{x})$ is finite for every $\mathbf{x} \in X$, we know that the system eq. (5.4) must be triangular up to permutation (since there would be no loops in paths specified by F on $X \setminus Q$). The same is clearly not true for eq. (5.5) and loops can easily arise as a result of random route-switching.

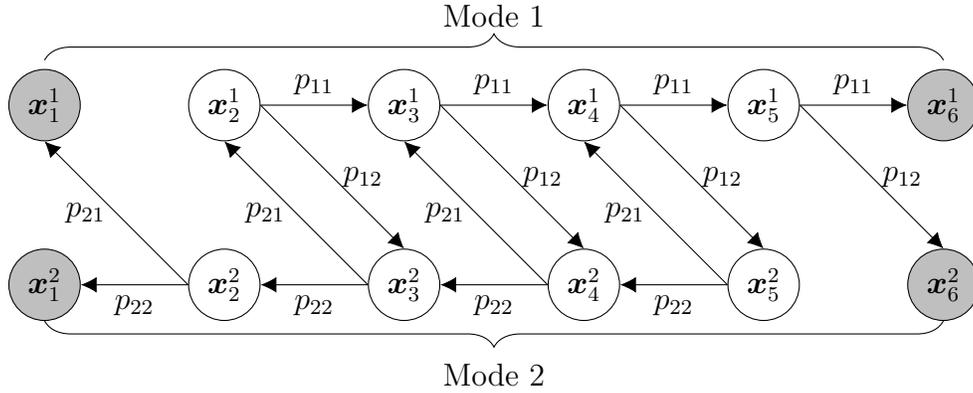


Figure 5.1: Fully discrete PDMP with $M = 2$ modes and $N = 6$ nodes. In mode 1 the motion is always to the right; in mode 2 the motion is always to the left. The exit set is $Q = \{\mathbf{x}_1, \mathbf{x}_6\}$.

We note that this process can be also interpreted as a Markov chain on an extended graph. One would create M copies of the original graph (on the nodes \mathbf{x}_n^i) with each mode/route F_i represented as a separate “layer” and inter-layer transitions governed by p_{ij} ’s. Figure 5.1 illustrates one such example with two modes and associated probabilities p_{11}, p_{12}, p_{21} , and p_{22} . In the special case of $K_i \equiv 1$ and $q_i \equiv 0$ for all $i \in \mathcal{M}$, the above equations for u_i ’s are simply describing the *mean hitting time* for the set $Q \times \mathcal{M}$. However, we are interested in more general costs and would also like to compute the full CDFs $w_i(\mathbf{x}, s) = \mathbb{P}(\mathcal{J}_i(\mathbf{x}) \leq s)$ for each \mathcal{J}_i . It is easy to show that these functions must satisfy a recursive relationship

$$w_i(\mathbf{x}, s) = \sum_{j=1}^M p_{ij} w_j(F_i(\mathbf{x}), s - K_i(\mathbf{x})), \quad \forall \mathbf{x} \notin Q, i \in \mathcal{M}, s > 0; \quad (5.6)$$

with the initial and boundary conditions

$$w_i(\mathbf{x}, s) = \begin{cases} 0, & \text{if } (\mathbf{x} \notin Q, s \leq 0) \text{ or } (\mathbf{x} \in Q, s < q_i(\mathbf{x})); \\ 1, & \text{if } \mathbf{x} \in Q, s \geq q_i(\mathbf{x}). \end{cases} \quad (5.7)$$

We will assume that the range of s values of interest is $\mathcal{S} = [0, S]$, where S is some constant specified in advance.

Based on the general properties of CDFs, all w_i ’s are monotone non-decreasing and upper-

semicontinuous in s . Moreover, the positivity of K_i 's ensures the explicit causality of this system: in eq. (5.6) each $w_i(\mathbf{x}, s)$ can only depend on $w_i(\mathbf{x}', s')$ if $s' < s$. Thus, the system can be solved in a single sweep (from the initial conditions at $s = 0$, “upward” in s).

Still, it can be useful to precompute $s_i^0(\mathbf{x}) = \inf\{s \mid w_i(\mathbf{x}, s) > 0\}$ and $w_i^0(\mathbf{x}) = w_i(\mathbf{x}, s_i^0(\mathbf{x}))$ by computations on X alone. Intuitively, $s_i^0(\mathbf{x})$ can be thought of as the minimum attainable cost starting in mode i at position \mathbf{x} , and $w_i^0(\mathbf{x})$ is the probability of attaining said cost. It is easy to see that s_i^0 satisfies the recursive system:

$$\begin{aligned} s_i^0(\mathbf{x}) &= K_i(\mathbf{x}) + \min_{\substack{j \in \mathcal{M} \\ \text{s.t. } p_{ij} > 0}} \{s_j^0(F_i(\mathbf{x}))\}, & \forall \mathbf{x} \in X \setminus Q, i \in \mathcal{M}; \\ s_i^0(\mathbf{x}) &= q_i(\mathbf{x}), & \forall \mathbf{x} \in Q, i \in \mathcal{M}. \end{aligned} \quad (5.8)$$

solvable by the standard Dijkstra's method in $O(MN \log(MN))$ operations.

The values of $w_i^0(\mathbf{x})$ can also be found in the process of computing $s_i^0(\mathbf{x})$. If $\mathcal{I}(\mathbf{x}) \subset \mathcal{M}$ is the arg min set in eq. (5.8), then

$$\begin{aligned} w_i^0(\mathbf{x}) &= \sum_{j \in \mathcal{I}(\mathbf{x})} p_{ij} w_j^0(F_i(\mathbf{x})), & \forall \mathbf{x} \in X \setminus Q, i \in \mathcal{M}; \\ w_i^0(\mathbf{x}) &= 1, & \forall \mathbf{x} \in Q, i \in \mathcal{M}. \end{aligned} \quad (5.9)$$

Numerically solving Equations (5.8) and (5.9) can be advantageous because they are computed on the lower-dimensional domain $X \times M$ instead of $X \times M \times \mathcal{S}$. This information can then be used as initial/boundary conditions to solve eq. (5.6) on a smaller subset of $X \times \mathcal{S}$.

5.2.2 Continuous PDMPs

Now we are interested in extending our results from the discrete case to continuous settings. First, we will consider a PDMP that is continuous in space (as in section 5.1), but with

discrete timesteps of length τ . We will assume that during a timestep started in mode $i \in \mathcal{M}$, the state first evolves from $\mathbf{y}(0) = \mathbf{x} \in \Omega \setminus Q$ according to an ODE $\mathbf{y}'(t) = \mathbf{f}_i(\mathbf{y}(t))$ with no random switches until the time

$$\tau_{\mathbf{x},i} = \min(\tau, \min\{t \mid \mathbf{y}(t) \in Q\}).$$

We define natural analogs for operators used to pose the graph routing problem in the previous subsection:

$$F_i(\mathbf{x}) = \mathbf{x} + \int_0^{\tau_{\mathbf{x},i}} \mathbf{f}_i(\mathbf{y}(t)) dt = \mathbf{y}(\tau_{\mathbf{x},i}), \quad (5.10)$$

$$K_i(\mathbf{x}) = \int_0^{\tau_{\mathbf{x},i}} C_i(\mathbf{y}(t)) dt, \quad (5.11)$$

where C_i is the running cost for that mode.

To stay consistent with the continuous in time Markov process described in (5.2), we define the mode-switching-over-the-time- $\tau_{\mathbf{x},i}$ probabilities $p_{ij}(\tau_{\mathbf{x},i}) = \mathbb{P}(m(\tau_{\mathbf{x},i}) = j \mid m(0) = i)$ by using the transition rate matrix $\Lambda = (\lambda_{ij})$. For $i \neq j$, λ_{ij} 's encode the rate of switching while the diagonal elements are defined by $\lambda_{ii} = -\sum_{j \neq i} \lambda_{ij}$. The evolution of the probability matrix $P(t) = (p_{ij}(t))$ is then given by an ODE

$$\frac{d}{dt}P(t) = P(t)\Lambda, \quad P(0) = I,$$

and it follows that $P(\tau_{\mathbf{x},i}) = \exp(\Lambda\tau_{\mathbf{x},i})$. Finally, if $\mathbf{y}(\tau_{\mathbf{x},i}) \in Q$, we assume that the PDMP will immediately terminate with an exit cost of $q_j(\mathbf{y}(\tau_{\mathbf{x},i}))$, where j is the final mode after a possible last transition.

With this notation in hand, we can define the same functions characterizing the random cumulative cost: u_i , w_i , d , s_i^0 , and w_i^0 will all satisfy the same recursive formulas already defined on a graph in the previous subsection. The only caveat is that p_{ij} 's will need to be replaced by $p_{ij}(\tau_{\mathbf{x},i})$. Since τ and $\tau_{\mathbf{x},i}$ are equivalent except on a small neighborhood of Q ,

in the following sections we will slightly abuse the notation by referring to τ to simplify the formulas.

A standard argument based on Taylor series expansion [45, 121] shows that the expected costs $u_i(\mathbf{x}) = \mathbb{E}[\mathcal{J}_i(\mathbf{x})]$ formally satisfy a system of weakly-coupled linear PDEs:

$$\nabla u_i(\mathbf{x}) \cdot \mathbf{f}_i(\mathbf{x}) + C_i(\mathbf{x}) + \sum_{j \neq i} \left[\lambda_{ij} (u_j(\mathbf{x}) - u_i(\mathbf{x})) \right] = 0 \quad (5.12)$$

with boundary conditions given by:

$$u_i(\mathbf{x}) = q(\mathbf{x}, i); \quad \forall i \in \mathcal{M}, \quad \forall \mathbf{x} \in Q.$$

Here we use a similar approach to derive a system of PDEs satisfied by the cumulative distribution functions $w_i(\mathbf{x}, s)$. More specifically, we will be taking the timestep τ to be 0 in the limit. The first order approximations of the transition probabilities are:

$$\begin{aligned} p_{ij}(\tau) &= 1 - e^{-\lambda_{ij}\tau} + o(\tau) = \lambda_{ij}\tau + o(\tau), \quad j \neq i \\ p_{ii}(\tau) &= 1 - \sum_{j \neq i} \lambda_{ij}\tau + o(\tau). \end{aligned} \quad (5.13)$$

The first-order approximation of the dynamics in eq. (5.10) is

$$F_i(\mathbf{x}) = \mathbf{x} + \tau \mathbf{f}_i(\mathbf{x}) + o(\tau), \quad (5.14)$$

and the first-order approximation of the running cost in eq. (5.11) is

$$K_i(\mathbf{x}) = \int_0^\tau C_i(\mathbf{y}(t)) dt = \tau C_i(\mathbf{x}) + o(\tau), \quad (5.15)$$

Plugging in our approximations (5.13), (5.14), and (5.15) into the recursive relationship in Equation (5.6) and then Taylor expanding w_i ,

$$w_i(\mathbf{x}, s) = \left(1 - \sum_{j \neq i} \lambda_{ij}\tau\right) w_i(F_i(\mathbf{x}), s - \tau C_i(\mathbf{x})) + \sum_{j \neq i} \lambda_{ij}\tau w_j(F_i(\mathbf{x}), s - \tau C_i(\mathbf{x})) + o(\tau)$$

$$w_i(\mathbf{x}, s) = w_i\left(F_i(\mathbf{x}), s - \tau C_i(\mathbf{x})\right) + \tau \sum_{j \neq i} \lambda_{ij} [w_j(\mathbf{x}, s) - w_i(\mathbf{x}, s)] + o(\tau)$$

$$0 = \tau \nabla w_i(\mathbf{x}, s) \cdot \mathbf{f}_i(\mathbf{x}) - \tau C_i(\mathbf{x}) \frac{\partial w}{\partial s}(\mathbf{x}, s) + \tau \sum_{j \neq i} \lambda_{ij} [w_j(\mathbf{x}, s) - w_i(\mathbf{x}, s)] + o(\tau),$$

where $\nabla = \left(\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \dots, \frac{\partial}{\partial x_d}\right)$ denotes the gradient in the spatial coordinates. Dividing both sides by τ and then taking the limit as $\tau \rightarrow 0$, we obtain a linear PDE for each mode i :

$$\nabla w_i(\mathbf{x}, s) \cdot \mathbf{f}_i(\mathbf{x}) - C_i(\mathbf{x}) \frac{\partial w_i}{\partial s}(\mathbf{x}, s) + \sum_{j \neq i} \lambda_{ij} [w_j(\mathbf{x}, s) - w_i(\mathbf{x}, s)] = 0. \quad (5.16)$$

This system of weakly coupled PDEs satisfies the initial conditions

$$w_i(\mathbf{x}, 0) = \begin{cases} 1, & \forall \mathbf{x} \in Q \text{ s.t. } q(\mathbf{x}, i) \leq 0 \\ 0, & \text{otherwise} \end{cases} \quad (5.17)$$

and the boundary conditions

$$w_i(\mathbf{x}, s) = \begin{cases} 1, & \forall \mathbf{x} \in Q \text{ s.t. } q(\mathbf{x}, i) \leq s, \\ 0, & \forall \mathbf{x} \in Q \text{ s.t. } q(\mathbf{x}, i) > s. \end{cases} \quad (5.18)$$

The above initial/boundary conditions are sufficient when $Q = \partial\Omega$ or if Ω is invariant under all vector fields \mathbf{f}_i . All of our examples considered in the next sections fall in this category. But more generally, if vector fields are such that a trajectory might leave Ω prior to reaching Q , one could treat this event as an immediate failure, essentially imposing $w_i(\mathbf{x}, s) = 0$ for all $\mathbf{x} \notin \Omega$ and all $s \in \mathbb{R}$.

As in the discrete case in section 5.2.1, it can be useful to precompute the minimum attainable cost to use as initial/boundary conditions when solving eq. (5.16). From the discrete case we recall that $s_i^0(\mathbf{x}) = \inf\{s \mid w_i(\mathbf{x}, s) > 0\}$ denotes the minimum cost possible when starting from position \mathbf{x} in mode i assuming that transitions between modes can occur whenever desired. In the continuous case these transitions can occur without delay, and therefore $s_i^0(\mathbf{x}) = s_j^0(\mathbf{x})$ for all i and j in \mathcal{M} , so we will replace all of these with $s^0(\mathbf{x})$. (Also,

unlike in the discrete case, it is entirely possible that $w_i(\mathbf{x}, s^0(\mathbf{x})) = 0$ for all i . The cost of $s^0(\mathbf{x})$ might be attainable only through perfectly timed transitions, which in the continuous case would happen with probability zero.) A formal Taylor series expansion of eq. (5.8) yields the following differential equation and boundary conditions for $s^0(\mathbf{x})$:

$$\begin{aligned} \min_i \{C_i(\mathbf{x}) + \nabla s^0(\mathbf{x}) \cdot \mathbf{f}_i(\mathbf{x})\} &= 0, & \mathbf{x} \in \Omega \setminus Q; \\ s^0(\mathbf{x}) &= \min_i \{q_i(\mathbf{x})\}, & \mathbf{x} \in Q. \end{aligned} \quad (5.19)$$

We are also interested in the probability $w_i^0(\mathbf{x})$ of attaining the minimum attainable cost $s_0(\mathbf{x})$ when starting from mode i and position \mathbf{x} . If we denote the argmin set of eq. (5.19) as $\mathcal{I}(\mathbf{x})$, then $w_i^0(\mathbf{x})$ formally satisfies the following system:

$$\begin{aligned} 0 &= \nabla w_i^0(\mathbf{x}) \cdot \mathbf{f}_i(\mathbf{x}) + \sum_{j \neq i} \lambda_{ij} [w_j^0(\mathbf{x}) - w_i^0(\mathbf{x})], & \mathbf{x} \in \Omega \setminus Q, i \in \mathcal{I}(\mathbf{x}); \\ w_i^0(\mathbf{x}) &= 1, & \mathbf{x} \in Q, i \in \mathcal{I}(\mathbf{x}); \\ w_i^0(\mathbf{x}) &= 0, & \mathbf{x} \in \Omega, i \notin \mathcal{I}(\mathbf{x}). \end{aligned} \quad (5.20)$$

Once $s^0(\mathbf{x})$ and $w_i^0(\mathbf{x})$'s are known, the computation of w_i 's can be restricted to $\{(\mathbf{x}, s) \mid s \in (s_0(\mathbf{x}), S]\}$, solving PDEs eq. (5.16) with “initial” conditions $w_i(\mathbf{x}, s^0(\mathbf{x})) = w_i^0(\mathbf{x})$.

5.2.3 Numerics for CDF computation

We will approximate the domain Ω with a rectangular grid of points $\{\mathbf{x}_{\mathbf{k}}\}$ with grid spacing Δx , where $\mathbf{k} = (k_1, \dots, k_d)$ is a multi-index and $\mathbf{x}_{\mathbf{k}} = (k_1 \Delta x, \dots, k_d \Delta x)$. We will also approximate the second argument of the CDF with regularly spaced points $s_n = n \Delta s$.

We will derive equations for a grid-function $W_{i,\mathbf{k}}^n \approx w_i(\mathbf{x}_{\mathbf{k}}, s_n)$, with $W_{i,\mathbf{k}}^0$ values determined by the initial conditions eq. (5.17). To simplify the discussion, we assume that both $\partial\Omega$ and Q are grid-aligned, with boundary values prescribed by eq. (5.18).

Equation eq. (5.6) is then naturally interpreted as a recipe for a semi-Lagrangian discretization using a pseudo-timestep of length τ . To obtain the first-order scheme, we can use the linear approximations eq. (5.15) and eq. (5.14), yielding the following equation at each gridpoint $\mathbf{x}_k \in \Omega$, mode $i \in \mathcal{M}$, and cost threshold s_n :

$$W_{i,\mathbf{k}}^n = \sum_{j=1}^M p_{ij}(\tau) W_j(\mathbf{x}_k + \tau \mathbf{f}_i(\mathbf{x}_k), s_n - \tau C_i(\mathbf{x}_k)), \quad n = 1, 2, \dots, N \quad (5.21)$$

where $W_j : \Omega \times \mathbb{R} \rightarrow \mathbb{R}$ is the result of interpolating the grid-function $W_{j,\mathbf{k}}^n$ in both \mathbf{x} and s variables, and the p_{ij} 's are defined as in eq. (5.13). In our implementation, all W_j 's are defined by multi-linear interpolation, but more sophisticated interpolation techniques (e.g., based on ENO/WENO [114]) may be used instead to decrease the numerical viscosity. More accurate approximations of F_i and K_i could be also employed to increase the formal order of accuracy of the discretization. For fully deterministic processes, similar semi-Lagrangian schemes have been proven to converge under the grid refinement to a piecewise-continuous solution on all compact sets not containing the discontinuity [10]. While we do not attempt to prove this here, our numerical experiments indicate that the same holds true in piecewise-deterministic problems.

Our update formula eq. (5.21) is only valid when $\mathbf{x}_k + \tau \mathbf{f}_i(\mathbf{x}_k)$ remains in Ω . With grid-aligned $\partial\Omega$, a rather conservative sufficient condition for this is

$$\tau \cdot \max_i \left\{ \max_{\mathbf{x}} \{ |\mathbf{f}(\mathbf{x}, i)| \} \right\} \leq \Delta x. \quad (5.22)$$

Furthermore, we would like to ensure that our updates are *causal*, that is the right hand side of eq. (5.21) depends only upon the values of W at lower s -slices, and not on any of W^n values. While not strictly necessary, this ensures that the updates for each mode are uncoupled, speeding up the computation. A sufficient condition for this is

$$\tau \cdot \min_i \left\{ \min_{\mathbf{x}} \{ C(\mathbf{x}, i) \} \right\} \geq \Delta s. \quad (5.23)$$

The inequality eq. (5.22) is only needed if we want to use the same τ at all grid points instead of selecting a smaller time step near $\partial\Omega$ only. But if this τ -uniformity is desired, satisfying both eq. (5.22) and eq. (5.23) requires

$$\frac{\Delta s}{\min\{C\}} \leq \frac{\Delta x}{\max\{|\mathbf{f}|\}}. \quad (5.24)$$

We note that, even though the above looks like a CFL-type restriction, it is not needed to guarantee the stability (semi-Lagrangian discretizations are unconditionally stable), but simply to ensure the causality (and hence the efficiency) of our discretization.

Under certain conditions, eq. (5.21) may also be re-interpreted as a finite differences discretization of the PDE eq. (5.16). To give a concrete example, suppose that $d = 1$, and the domain $\Omega = [0, 1]$ is approximated by a grid of regularly spaced points denoted $x_k = k\Delta x$. Furthermore, suppose that there is a mode i where $C_i = 1$, and $f_i(x_k) = f_{ik} > 0$. If we choose $\tau = \Delta s$, then eq. (5.21) for $n + 1$ becomes:

$$\begin{aligned} W_{i,k}^{n+1} &= \sum_{j=1}^M p_{ij}(\Delta s) W_j(x_k + f_{ik}\Delta s, s_n) \\ &= W_i(x_k + f_{ik}\Delta s, s_n) + \sum_{j \neq i} \lambda_{ij} \Delta s [W_j(x_k + f_{ik}\Delta s, s_n) - W_i(x_k + f_{ik}\Delta s, s_n)] \\ &= W_{i,k}^n + \frac{f_{ik}\Delta s}{\Delta x} (W_{i,k+1}^n - W_{i,k}^n) + \sum_{j \neq i} \lambda_{ij} \Delta s (W_j - W_i)(x_k + f_{ik}\Delta s, s_n); \\ f_{ik} \left[\frac{W_{i,k+1}^n - W_{i,k}^n}{\Delta x} \right] - \left[\frac{W_{i,k}^{n+1} - W_{i,k}^n}{\Delta s} \right] + \sum_{j \neq i} \lambda_{ij} (W_j - W_i)(x_k + f_{ik}\Delta s, s_n) &= 0, \end{aligned} \quad (5.25)$$

which is a consistent first-order finite differences discretization of eq. (5.16). Furthermore, in this 1D example, the CFL condition for this discretization is exactly eq. (5.24). The scheme eq. (5.25) is monotone (and thus stable [39]) whenever this CFL condition is satisfied. It is important to note that the summands in eq. (5.25) are evaluated at $x_k + f_{ik}\Delta s$ (and therefore are convex combinations of W^n values at x_k and x_{k+1}). Evaluating those terms at the naive choice of x_k would result in a non-monotone discretization, which is in fact unstable.

To compute the minimum attainable cost $s^0(\mathbf{x})$ and the probability of attaining it $w^0(\mathbf{x})$, we use first-order semi-Lagrangian discretizations of eq. (5.19) and eq. (5.20). For $d = 1$, the discretized equations for $s^0(x)$ are

$$\begin{aligned} s^0(x_k) &= \min_i \left\{ C_i(x_k) \frac{\Delta x}{|f_i(x_k)|} + s^0(x_{k'}) \right\}, & x_k \notin Q; \\ s^0(x_k) &= \min_i \{q_i(x_k)\}, & x_k \in Q; \end{aligned} \quad (5.26)$$

where

$$k' = \begin{cases} k + 1, & f_i(x_k) > 0; \\ k - 1, & f_i(x_k) < 0. \end{cases}$$

In 1D, this system of equations can be solved efficiently with two iterative “sweeps” – first increasing and then decreasing in k . In higher space dimensions, it can be solved in $O(MN \log(N))$ time using a Dijkstra-like method.

In the process of solving for $s^0(x)$, we also solve eq. (5.20) using a first-order semi-Lagrangian scheme. Using $\mathcal{I}(x_k)$ to denote the argmin set of (5.26), the values of w_i^0 are initialized according to

$$w_i^0(x_k) = \begin{cases} 1, & x_k \in Q, i \in \mathcal{I}(x_k); \\ 0, & \text{otherwise.} \end{cases}$$

Whenever the value of $s^0(x_k)$ is updated, we simultaneously update w_i^0 according to

$$w_i^0(x_k) = \begin{cases} w_i^0(x_{k'}) + \frac{\Delta x}{|f_i(x_{k'})|} \sum_{j \neq i} \lambda_{ij} [w_j^0(x_{k'}) - w_i^0(x_{k'})], & i \in \mathcal{I}(x_k); \\ 0, & i \notin \mathcal{I}(x_k). \end{cases} \quad (5.27)$$

These values of $s^0(x)$ and $w_i^0(x)$ are then used as initial/boundary conditions¹ for computing $w_i(x)$. This provides a speed improvement and also reduces the smearing of w_i 's discontinuities due to numerical viscosity.

¹ Since the graph of $s^0(x)$ is generally not grid aligned in $\Omega \times \mathcal{S}$, such a domain restriction requires either a use of “cut cells” just above $s = s^0(x)$ or a conservative “rounding up” of s^0 values. Our implementation relies on the latter, which introduces additional $O(\Delta s)$ errors.

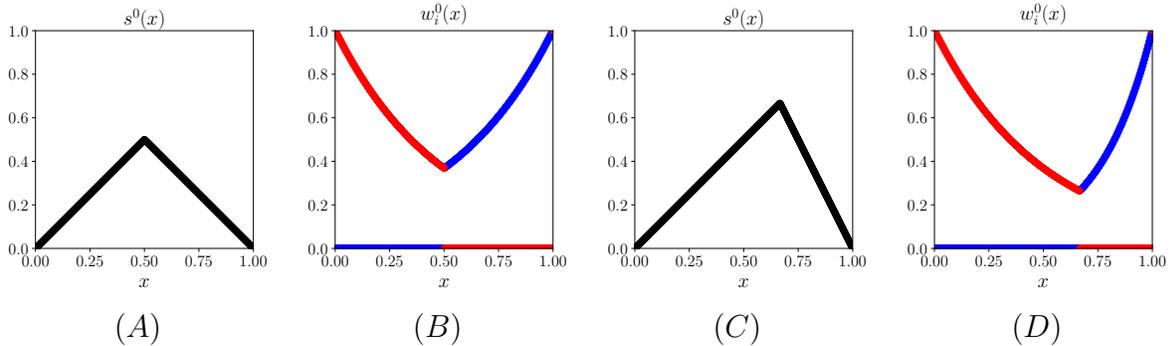


Figure 5.2: Minimum cost $s^0(x)$ and probability $w_i^0(x)$ of attaining that minimum cost. Subfigures (A) and (B) are for Example 1 and (C) and (D) are for Example 2. Graphs of w_i^0 are shown in blue for $i = 1$ and in red for $i = 2$.

5.2.4 Experimental Results

We illustrate our approach with three examples of uncontrolled PDMPs on \mathbb{R} and \mathbb{R}^2 . In all of these, we assume $Q = \partial\Omega$, $C \equiv 1$, and $q \equiv 0$, ensuring that the cumulative cost \mathcal{J} corresponds to the time to $\partial\Omega$. For simplicity, we will also assume uniform transition rates; i.e., $\lambda_{ij} = \lambda > 0$ for all $i \neq j$.

Example 1: We start by considering a “sailboat” test problem described in the introduction with $\Omega = [0, 1]$, $Q = \{0, 1\}$, $M = 2$, $f_i(x) = (-1)^{i+1}$, and symmetric transition rates $\lambda_{12} = \lambda_{21} = 2$. For a fixed number N of gridpoints, we set $\Delta s = \Delta x = \frac{1}{N-1}$, as this is the largest value of Δs that satisfies eq. (5.24). Moreover, this guarantees that no actual interpolation is necessary in eq. (5.21), as W_j is only evaluated at gridpoints. We note that solving these discretized equations is equivalent to finding the CDF of a discrete PDMP such as the one pictured in fig. 5.1, except with a larger number N of nodes. We solve this problem for $s \in [0, 1]$, but also precompute $s^0(x)$ and $w_i^0(x)$ (see fig. 5.2(A-B)) to reduce the computational domain for w_i ’s.

The key advantage of our approach is that it approximates the distribution \mathcal{J} for all starting configurations simultaneously. Once w_i ’s are computed, we can freeze (x, i) and

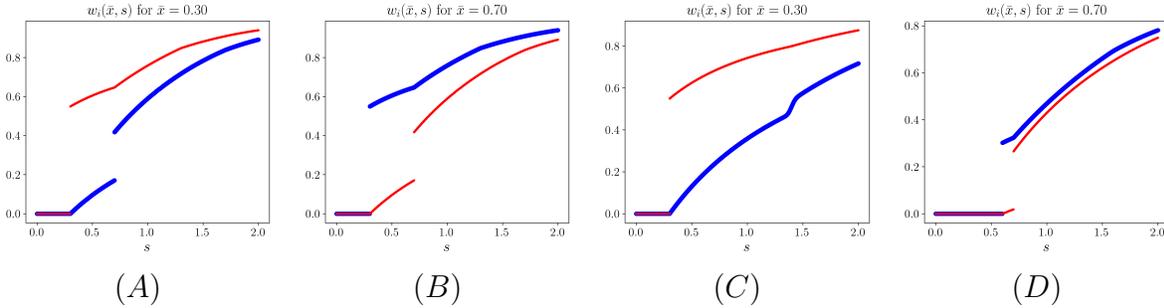


Figure 5.3: CDF for a particle starting at initial position x , in Mode 1 (blue) and Mode 2 (red). Subfigures (A) and (B) are for Example 1 and (C) and (D) are for Example 2.

vary s to study the CDF. In fig. 5.3(A-B) this is illustrated for two starting locations $x = 0.3$ and $x = 0.7$. But it might be even more revealing to fix a particular deadline s and consider the probability of meeting it from all possible initial configurations. In fig. 5.4 we show such graphs of $w_i(x, s)$ for four different s values. Geometric properties of these functions have a natural interpretation, which we highlight focusing on mode 1 and $s = 0.25$ (the blue plot in the first subfigure). First, regardless of mode switches, $s = 0.25$ is not enough time to exit if we start too far from Q ; so, $w_1 = w_2 = 0$ for all $x \in (0.25, 0.75)$. Second, starting from $x = 0.75$ and moving right with speed one we will have just enough time to reach Q provided we experience no mode switches, and if any switches occur the resulting time to target will be higher. So, the jump discontinuity at $x = 0.75$ is precisely the probability of zero mode switches occurring in $s = 0.25$ time units. (We note that this discontinuity disappears in the last subfigure since $s = 1.00$ is enough time to reach Q with no mode switches starting from any $(x, i) \in \Omega \times \mathcal{M}$.) Finally, a similar argument explains the behavior for starting positions on $x \in (0, 0.25)$. Since we start in mode 1, the only hope of meeting the $s = 0.25$ deadline is a quick switch to mode 2. Starting from $x = 0.25$, a timely arrival would require an immediate mode switch, and since this happens with probability zero, w_1 is continuous at this point.

Of course, the probability of meeting a deadline is also significantly influenced by the switching rates. While we do not illustrate this here, the same example is repeated with a

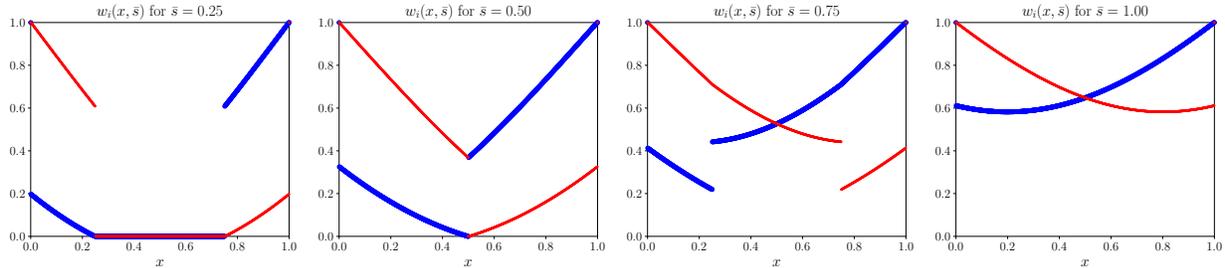


Figure 5.4: Example 1: Equal speeds, with a symmetric transition rate $\lambda = 2$. Each subplot is a snapshot of $w_i(x, s) = \mathbb{P}(\mathcal{J}_i(\mathbf{x}) \leq s)$ for a specific value of s . In Mode 1 (blue), the particle moves to the right with speed 1. In Mode 2 (red), the particle moves to the left with speed 1. Computed on $\Omega \times \mathcal{S} = [0, 1]^2$ with $\Delta x = \Delta s = 0.001$.

range of symmetric and asymmetric rates in Figure 5.7 of section 5.3.

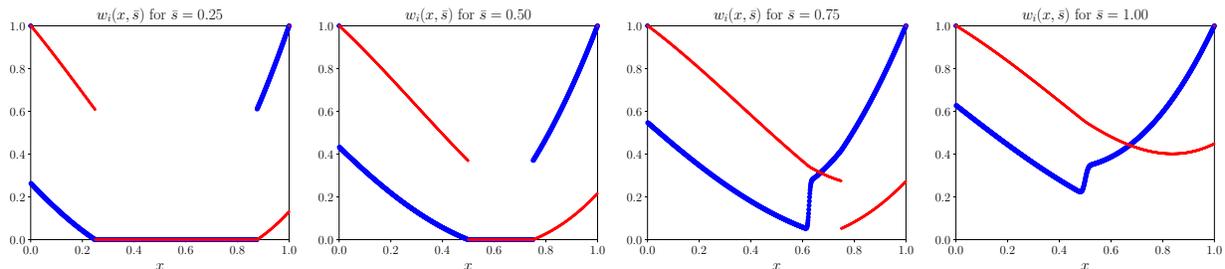


Figure 5.5: Example 2: Unequal speeds, with a transition rate $\lambda = 2$. Each subplot is a snapshot of $w_i(x, s) = \mathbb{P}(\mathcal{J}_i(\mathbf{x}) \leq s)$ for a specific value of s . In Mode 1 (blue), the particle moves to the right with speed $1/2$. In Mode 2 (red), the particle moves to the left with speed 1. Computed on $\Omega \times \mathcal{S} = [0, 1]^2$ with $\Delta x = \Delta s = 0.001$.

Example 2: We modify the previous example by considering unequal speeds of motion in different modes: $f_1 = 0.5$ and $f_2 = -1$. The CDFs for two starting locations $x = 0.3$ and $x = 0.7$ are shown in fig. 5.3(C-D) while the plots of $w_i(x, s)$ for four different values of s can be found in fig. 5.5. We note that in Mode 1, interpolation is now necessary in eq. (5.21), which results in numerical diffusion smoothing out discontinuities, as can be seen in the right two subfigures. The absence of such artifacts in the first two subfigures is an additional benefit of pre-computing $s^0(x)$ and $w_i^0(x)$, shown in fig. 5.2(C-D), to reduce the computational domain for w_i 's. E.g., for the first subfigure, all $x \in [0.25, 0.875)$ have $s^0(x) > s = 0.25$ and so are assigned an exit probability of 0, removing the need of interpolating

across discontinuities. In contrast, at $s = 0.75$ all x have a nonzero probability of exiting, so interpolation across the discontinuity at $x = 0.625$ is unavoidable.

Example 3: We now consider a 2D version of Example 1, with $\Omega = [0, 1] \times [0, 1]$, $Q = \partial\Omega$, $M = 4$, and $\lambda = 1$. In all modes, the motion is with speed $|\mathbf{f}| = 1$, but the directions of motion differ: $\leftarrow, \uparrow, \rightarrow,$ and \downarrow in modes $1, \dots, 4$ respectively. Numerical approximations of w_i 's for different values of s are shown in fig. 5.6. The distinct delineations between darker and lighter regions are analogous to the discontinuities in the earlier one-dimensional cases. For example, given $s < 0.5$ and starting positions along the line $y = 0.5$, a timely exit is only possible to the left (via Mode 1) or to the right (via Mode 3). Therefore, cross sections of w_1 and w_3 along $y = 0.5$ at $s = 0.25$ in fig. 5.6 coincide with the one-dimensional graphs for $s = 0.25$ in fig. 5.4. However, as we move closer to the corners of the domain, all four modes have an effect on the probability of exit. For example, the region along the diagonal near the top right corner of the w_1 graph has higher exit probabilities than surrounding regions because there are multiple possible timely-exit strategies. These 2D phenomena become prevalent for higher s values.

5.3 Bounds on CDF

We now turn to PDMPs with parameter uncertainty – in addition to the inherent aleatoric uncertainty due to mode switches. In section 5.2, the uncertainty of the outcome could be fully characterized by its CDF computed based on the known transition rates between modes, λ_{ij} 's. Here, however, we consider the case where we only know a *range* of potential λ_{ij} values. There are two natural models of epistemic uncertainty in this situation, and it is meaningful to consider the upper and lower bounds on the CDF with each of them. We focus on a case where the true transition rates are free to fluctuate within the given range and may take on

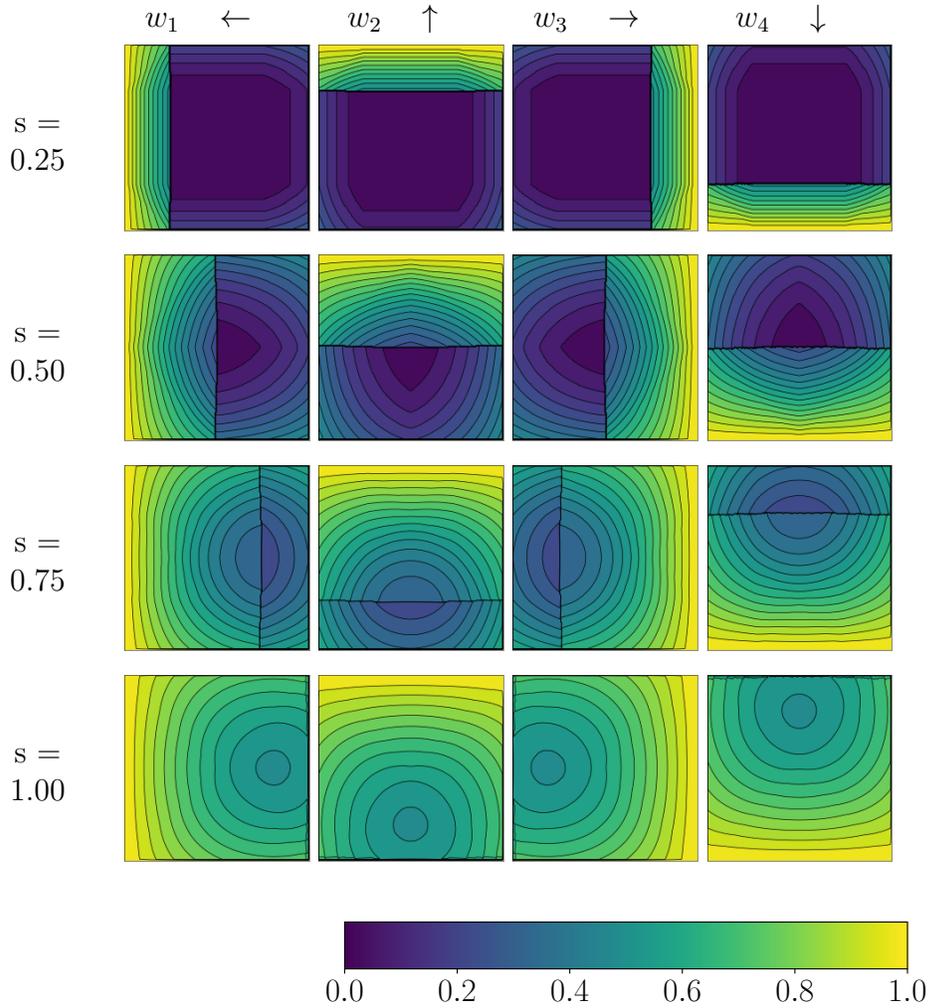


Figure 5.6: Example 3: Mode switching in 2D with transition rates $\lambda = 1$. The particle moves \leftarrow in Mode 1, \uparrow in Mode 2, \rightarrow in Mode 3, and \downarrow in Mode 4. Computed on $\Omega \times \mathcal{S} = [0, 1]^3$ with $\Delta x = \Delta y = \Delta s = 0.01$.

different values at different times. The upper and lower bounds on the CDF can be then found by considering a nonlinear version of the coupled PDEs seen in section 5.2. This can also be viewed as a game against nature, where nature is a player maximizing (or minimizing) over transition rates at any given time and position. The alternative model is to assume that all transition rates remain fixed (though unknown) throughout the process. We provide some experimental results for this case as well, though do not propose any computationally efficient methods for finding sharp CDF bounds.

5.3.1 Deriving PDEs

We now extend the results of section 5.2.2 by considering the case in which the transition rate matrix $\Lambda = (\lambda_{ij})$ is not necessarily constant. Suppose there are known a_{ij}, b_{ij} , for each $i \neq j$, such that each λ_{ij} may vary in the interval $0 \leq a_{ij} \leq \lambda_{ij} \leq b_{ij}$ throughout the process. If L is the set of possible transition matrices satisfying these constants, we will assume that Λ might be changing but remains in L throughout the process.

We compute an upper bound for w_i , denoted w_i^+ , by taking its initial and boundary conditions to be the same as w_i , and for each (\mathbf{x}, s) taking the $\Lambda \in L$ which maximizes $w_i^+(\mathbf{x}, s)$. Similarly, for the lower bound w_i^- we take the $\Lambda \in L$ which minimizes $w_i^-(\mathbf{x}, s)$. Hence these satisfy the PDEs:

$$\nabla w_i^+(\mathbf{x}, s) \cdot \mathbf{f}_i(\mathbf{x}) - C_i(\mathbf{x}) \frac{\partial w_i^+}{\partial s}(\mathbf{x}, s) + \sum_{j \neq i} \max_{\Lambda \in L} \{ \lambda_{ij} [w_j^+(\mathbf{x}, s) - w_i^+(\mathbf{x}, s)] \} = 0; \quad (5.28)$$

$$\nabla w_i^-(\mathbf{x}, s) \cdot \mathbf{f}_i(\mathbf{x}) - C_i(\mathbf{x}) \frac{\partial w_i^-}{\partial s}(\mathbf{x}, s) + \sum_{j \neq i} \min_{\Lambda \in L} \{ \lambda_{ij} [w_j^-(\mathbf{x}, s) - w_i^-(\mathbf{x}, s)] \} = 0 \quad (5.29)$$

with initial and boundary conditions (5.17) and (5.18).

As with the computation of w_i 's through (5.16), it can be helpful to precompute the minimal attainable cost $s^0(\mathbf{x})$ and the probability $w_i^0(\mathbf{x})$ of achieving such a cost. Since Λ is not constant, instead of having a probability of attaining the cost $s^0(\mathbf{x})$ we have a lower bound $w_i^{0,-}(\mathbf{x})$ and an upper bound $w_i^{0,+}(\mathbf{x})$ for that probability. We may compute $s^0(\mathbf{x})$ in precisely the same way as in (5.19) since that formula does not depend on Λ at all. On the other hand, to compute $w_i^{0,+}(\mathbf{x})$ we must modify (5.20) to account for the changing Λ to be

instead:

$$\begin{aligned}
0 &= \nabla w_i^{0,+}(\mathbf{x}) \cdot \mathbf{f}_i(\mathbf{x}) + \sum_{j \neq i} \max_{\Lambda \in L} \{ \lambda_{ij} [w_j^{0,+}(\mathbf{x}) - w_i^{0,+}(\mathbf{x})] \}, & \mathbf{x} \in \Omega \setminus Q, i \in \mathcal{I}(\mathbf{x}); \\
w_i^{0,+}(\mathbf{x}) &= 1, & \mathbf{x} \in Q, i \in \mathcal{I}(\mathbf{x}); \\
w_i^{0,+}(\mathbf{x}) &= 0, & \mathbf{x} \in \Omega, i \notin \mathcal{I}(\mathbf{x}). \quad (5.30)
\end{aligned}$$

Similarly, to compute $w_i^{0,-}(\mathbf{x})$ we have:

$$\begin{aligned}
0 &= \nabla w_i^{0,-}(\mathbf{x}) \cdot \mathbf{f}_i(\mathbf{x}) + \sum_{j \neq i} \min_{\Lambda \in L} \{ \lambda_{ij} [w_j^{0,-}(\mathbf{x}) - w_i^{0,-}(\mathbf{x})] \}, & \mathbf{x} \in \Omega \setminus Q, i \in \mathcal{I}(\mathbf{x}); \\
w_i^{0,-}(\mathbf{x}) &= 1, & \mathbf{x} \in Q, i \in \mathcal{I}(\mathbf{x}); \\
w_i^{0,-}(\mathbf{x}) &= 0, & \mathbf{x} \in \Omega, i \notin \mathcal{I}(\mathbf{x}). \quad (5.31)
\end{aligned}$$

In both cases, $\mathcal{I}(\mathbf{x})$ is the argmin set of eq. (5.19) as in eq. (5.20).

5.3.2 Calculating Bounds

For numerical computations of the bounds described in section 5.3.1, we rely on a discretization similar to that presented in section 5.2. When τ is small enough that the approximations in (5.13) can be made, this optimization can be written using recursively. For the CDF lower bound $w_i^-(\mathbf{x}, s)$, the semi-Lagrangian scheme is

$$w_i^-(\mathbf{x}, s) = w_i^-(\tilde{\mathbf{x}}, \tilde{s}) + \min_{\Lambda \in L} \left\{ \tau \sum_{j \neq i} \lambda_{ij} [w_j^-(\tilde{\mathbf{x}}, \tilde{s}) - w_i^-(\tilde{\mathbf{x}}, \tilde{s})] \right\} \quad (5.32)$$

where $\tilde{\mathbf{x}} = F_i(\mathbf{x})$ and $\tilde{s} = s - \tau C_i(\mathbf{x})$.

Recall from eq. (5.23) that $C_i(\mathbf{x}) > 0$ so it is always the case that $\tilde{s} < s$. Therefore, $\mathbf{w}_i^\pm(\tilde{\mathbf{x}}, \tilde{s})$ has already been calculated and so can be used in the computation of $\mathbf{w}_i^\pm(\mathbf{x}, s)$.

For an efficient implementation of eq. (5.32), the optimal $\Lambda^* \in L$ can be found explicitly. When minimizing $w_i^-(\mathbf{x}, s)$, we would naturally like to subtract as much as possible and add as little as possible to $w_i^-(F_i(\mathbf{x}), s - \tau C_i(\mathbf{x})) = w_i^-(\tilde{\mathbf{x}}, \tilde{s})$. Therefore,

$$\begin{aligned} [w_j^-(\tilde{\mathbf{x}}, \tilde{s}) - w_i^-(\tilde{\mathbf{x}}, \tilde{s})] \leq 0 &\implies \lambda_{ij}^* = b_{ij}; \\ [w_j^-(\tilde{\mathbf{x}}, \tilde{s}) - w_i^-(\tilde{\mathbf{x}}, \tilde{s})] > 0 &\implies \lambda_{ij}^* = a_{ij}. \end{aligned} \tag{5.33}$$

For the the CDF upper bound w_i^+ , the scheme is similar modulo replacing min with max in eq. (5.32) and flipping the signs of inequalities in eq. (5.33).

5.3.3 Experimental Results

Example 4: CDF bounds and comparison to fixed- Λ CDFs

We now generalize Example 1 from section 5.2.4 to consider epistemic uncertainty. Recall that $\Omega = [0, 1]$, $Q = \partial\Omega$, $C \equiv 1$, and $q \equiv 0$ so that the cumulative cost \mathcal{J} corresponds to the exit time, with $M = 2$ and $f_i = (-1)^{i+1}$. We will also assume that $\lambda_{ij} \in [1, 4]$ for all $i \neq j$. In fig. 5.7 we display our results for a particle that starts moving rightward (in Mode 1). The graphs shown in blue are the upper and lower bounds on the probability of a timely exit (i.e., before a specific deadline \bar{s}) for each initial positions x . The bounds on CDF for two starting positions \bar{x} are shown in fig. 5.8. All of these bounds are computed from eq. (5.28) and eq. (5.29) for the model of epistemic uncertainty where $\Lambda = (\lambda_{ij})$ is allowed to fluctuate within L . Under this model, these bounds are sharp since they are computed by finding CDF-maximizing (and minimizing) sequences of Λ 's.

We can also compare the blue bounds to the corresponding timely-exit probabilities for a process containing epistemic uncertainty via fixed and unknown (possibly asymmetric) transition matrix Λ . The green curves shown in Figures 5.7 and 5.8 are computed by repeatedly

solving eq. (5.16) for a coarse grid of specific Λ 's in L . It should be noted that processes with this type of epistemic uncertainty are a subset of those previously discussed, and so the blue bounds will definitely hold but will no longer be sharp. This lack of sharpness is not surprising since changing the transition rate can often result in a “better” (higher or lower – depending on the bound) probability of timely exit. However, calculating tighter bounds for a “fixed-unknown- Λ ” case is computationally expensive. By inspection of the experimental data, it is clear that such sharp bounds would have to be composed of many individual fixed- Λ CDFs.

5.4 Optimizing the CDF

The PDMPs considered in previous sections were not controllable in any way. Since the dynamics are deterministic in every mode, each random trajectory was fully described by the initial (state, mode) pair and the discrete time sequence of mode switches. The goal was to develop efficient methods for approximating the CDF of the cost accumulated up till termination. We now turn to *controlled* PDMPs [45] – a modeling framework useful in a wide range of applications, including production/maintenance planning [20], control of manufacturing processes [2, 18, 96, 107], multi-generational games [65], economic growth & climate change modeling[64], trajectory optimization for emergency vehicles [6], preventing the extraction of protected natural resources [30], and robotic navigation [113].

We start with expectation-optimal controls considered in the above references, but then switch to selecting controls to manage the uncertainty in \mathcal{J} and provide some notion of *robustness*. Robust controls help practitioners to guard against both modeling errors and prohibitively bad rare outcomes, which may result from random switches. It might seem natural to mirror the robust approaches popular in traditional stochastic control, but we find

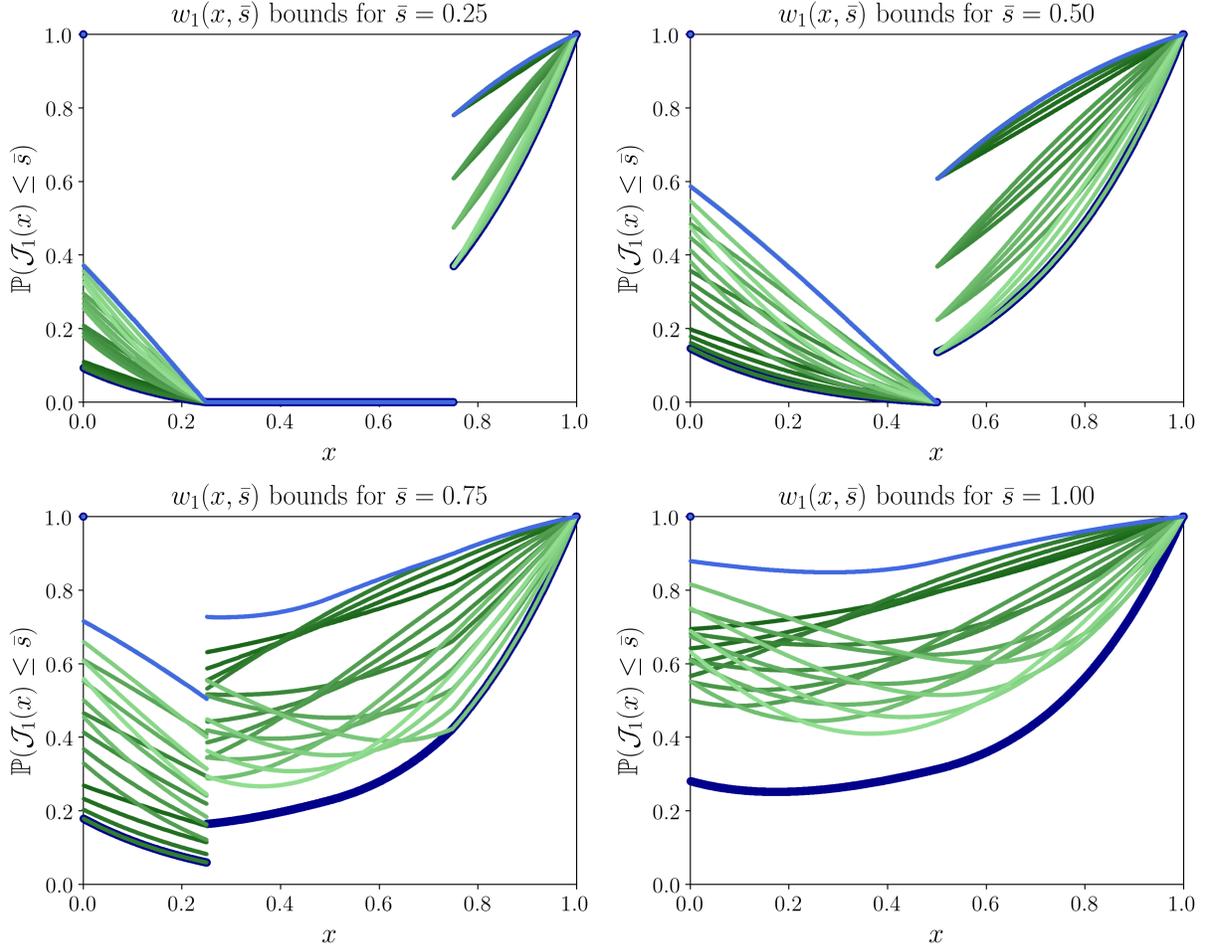


Figure 5.7: Example 4: bounds on the probability of timely exit (starting in Mode 1) for four different deadline values \bar{s} . (Probabilities and bounds for starting in Mode 2 can be obtained by a mirror symmetry relative to the line $x = 0.5$.) Blue bounds are produced under the varying rates assumption by solving eq. (5.28)-eq. (5.29) for $\lambda_{12}, \lambda_{21} \in [1, 4]$. Green curves are produced under the fixed rates assumption by solving eq. (5.16), each corresponding to a specific $(\lambda_{12}, \lambda_{21}) \in \{1, 2, 3, 4\} \times \{1, 2, 3, 4\}$. The darkest four curves are those associated with $\lambda_{12} = 1$, the next four are those associated with $\lambda_{12} = 2$, and so on. Computed on $\Omega \times \mathcal{S} = [0, 1]^2$ with $\Delta x = \Delta s = 0.001$.

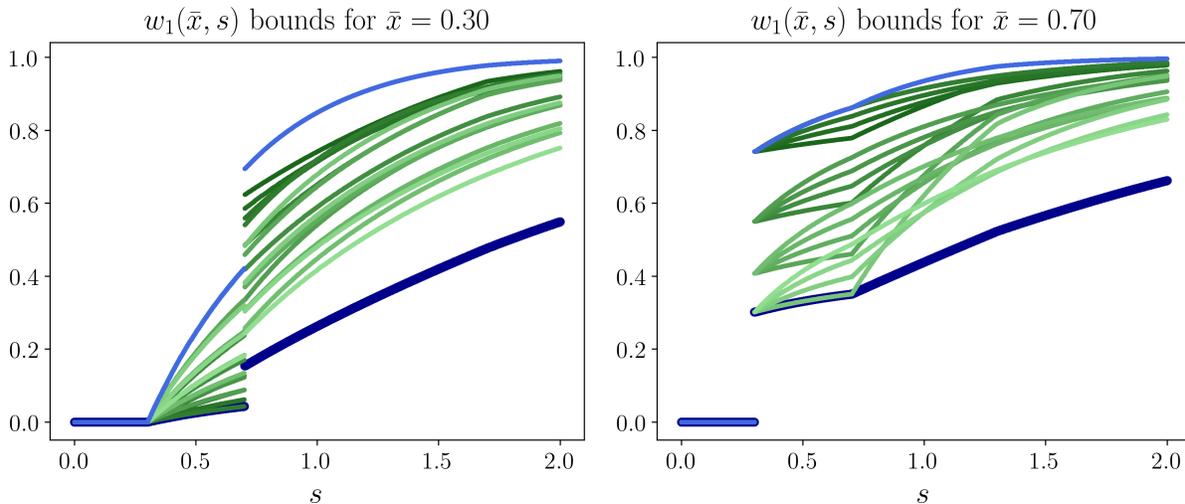


Figure 5.8: Example 4: bounds on CDF (starting in Mode 1) for two different initial positions $\bar{x} = 0.3$ and $\bar{x} = 0.7$. See a detailed legend in caption of fig. 5.7.

them lacking in the PDMP context. H_∞ controls are the mainstay of robustness for many processes with continuous perturbations [13], but they are not easily adaptable for discrete mode-switches. Another popular idea is to minimize $\mathbb{E}[\exp(\beta\mathcal{J})]$, with the risk-sensitivity coefficient $\beta > 0$ reflecting our desire to avoid bad outcomes [54]. For small β values, this is roughly equivalent to minimizing a convex combination of $\mathbb{E}[J]$ and $Var[J]$. While implementable with PDMPs, this method does not provide any guarantees on the likelihood of bad scenarios. We thus develop a different approach to maximize the probability of not exceeding a specific cumulative cost threshold \bar{s} . In subsection 5.4.2 we develop PDEs to find such optimal policies for all initial configurations and all threshold values \bar{s} *simultaneously*. The numerical methods and computational examples (for $d = 1$ and $d = 2$) are covered in subsections 5.4.3 and 5.4.4 respectively.

5.4.1 Controlled PDMPs and expectation-optimal policies

To obtain a controlled PDMP, we will assume that both the running cost C and velocity \mathbf{f} also depend on additional control parameters, which can be changed dynamically while the system travels through $\Omega \times \mathcal{M}$. Assuming that the set of available control values A is a compact subset of \mathbb{R}^n , we will consider a set of measurable *feedback-control* policies $\mathcal{A} = \{\mathbf{a} : (\Omega \times \mathcal{M}) \rightarrow A\}$. Once $\mathbf{a}(\cdot) \in \mathcal{A}$ is chosen, we can define

$$\mathbf{f}(\mathbf{x}, i) = \mathbf{f}(\mathbf{x}, i, \mathbf{a}(\mathbf{x}, i)) \quad \text{and} \quad C(\mathbf{x}, i) = C(\mathbf{x}, i, \mathbf{a}(\mathbf{x}, i)), \quad (5.34)$$

with equations (5.1)-(5.3) describing the resulting trajectory and cumulative cost. The latter will be denoted $\mathcal{J}^{\mathbf{a}(\cdot)}(\mathbf{x}, i) = \mathcal{J}_i^{\mathbf{a}(\cdot)}(\mathbf{x})$ to highlight the dependence on the chosen control policy. The corresponding expected cost $u_i^{\mathbf{a}(\cdot)} = \mathbb{E}[\mathcal{J}_i^{\mathbf{a}(\cdot)}(\mathbf{x})]$ and the CDF $w_i^{\mathbf{a}(\cdot)}(\mathbf{x}, s) = \mathbb{P}[\mathcal{J}_i^{\mathbf{a}(\cdot)}(\mathbf{x}) \leq s]$ can then be found from equations (5.12) and (5.16) respectively.

The usual goal in controlled PDMPs literature is to minimize the expected total cost up to the termination time. It is easy to show that the value function defined as

$$\widehat{u}(\mathbf{x}, i) = \widehat{u}_i(\mathbf{x}) = \inf_{\mathbf{a}(\cdot) \in \mathcal{A}} \mathbb{E}[\mathcal{J}_i^{\mathbf{a}(\cdot)}(\mathbf{x})]$$

must satisfy the following *tail-optimality* property:

$$\widehat{u}(\mathbf{x}, i) = \inf_{\mathbf{a}(\cdot) \in \mathcal{A}} \mathbb{E} \left[\int_0^\tau C_{m(t)}(\mathbf{y}(t), \mathbf{a}(\mathbf{y}(t), m(t))) dt + \widehat{u}(\mathbf{y}(\tau), m(\tau)) \right], \quad (5.35)$$

for all $\tau > 0$ sufficiently small to guarantee that $\mathbf{y}(t) \in \Omega \setminus Q$ for all $t \in [0, \tau]$, $\mathbf{a}(\cdot) \in \mathcal{A}$. Recall from section 5.1 that $\mathbf{y}(0) = \mathbf{x}$, $m(0) = i$, and the changes in mode $m(t)$ are governed by the matrix of switching rates Λ . The evolution of the continuous portion of the state is now control-dependent since $\mathbf{y}'(t) = \mathbf{f}_{m(t)}(\mathbf{y}(t), \mathbf{a}(\mathbf{y}(t), m(t)))$. In this section, we will further assume that all \mathbf{f}_i 's and C_i 's are Lipschitz-continuous in both arguments. A standard argument [45, 121] based on Taylor-expanding (5.35) shows that, if \widehat{u}_i 's are

sufficiently smooth, they must satisfy a weakly-coupled system of Hamilton-Jacobi-Bellman PDEs:

$$\min_{\mathbf{a} \in \mathcal{A}} \{ \nabla \widehat{u}_i(\mathbf{x}) \cdot \mathbf{f}_i(\mathbf{x}, \mathbf{a}) + C_i(\mathbf{x}, \mathbf{a}) \} + \sum_{j \neq i} \lambda_{ij} (\widehat{u}_j(\mathbf{x}) - \widehat{u}_i(\mathbf{x})) = 0, \quad \mathbf{x} \in \Omega \setminus Q, i \in \mathcal{M} \quad (5.36)$$

with boundary conditions $\widehat{u}_i(\mathbf{x}) = q_i(\mathbf{x})$ for all $\mathbf{x} \in Q$. In a non-smooth case, these value functions can be still interpreted as the unique viscosity solution [44]. The system (5.36) is a natural non-linear generalization of (5.12) and can be similarly discretized by semi-Lagrangian techniques. However, the coupling between different modes makes it difficult to solve the discretized system efficiently even in the case of simple/isotropic cost and dynamics. A variety of Dijkstra-like non-iterative methods developed for deterministic problems (e.g., [109, 111, 31, 4, 92]) will not be applicable for $\Lambda \neq 0$ and one has to resort to slower iterative algorithms instead [113].

We will further refer to a policy $\mathbf{a}_*(\cdot) \in \mathcal{A}$ as *expectation-optimal* if $\widehat{u}_i(\mathbf{x}) = u_i^{\mathbf{a}_*(\cdot)}(\mathbf{x})$ for all $(\mathbf{x}, i) \in \Omega \times \mathcal{M}$. Once \widehat{u}_i 's are computed, such an optimal policy can be defined pointwise by utilizing arg min values² from (5.36). We note that the $\mathbf{a}_*(\cdot)$ -determined running cost and dynamics defined by (5.34) will be only piecewise Lipschitz in \mathbf{x} , which is precisely the setting considered in section 5.2.

5.4.2 PDEs for threshold-specific optimization

In contrast to the above expectation-centric approach, our goal is to generalize the CDF-computation methods of section 5.2 by choosing control policies that maximize the probability of desirable outcomes. Two subtleties associated with this approach are worth pointing

² Additional assumptions on \mathbf{f}_i 's and C_i 's can be imposed to ensure that this arg min is a singleton as long as \widehat{u}_i is differentiable [9]. But the expectation-optimal policy is still not uniquely defined at the points where $\nabla \widehat{u}_i$ does not exist, and a tie-breaking procedure (e.g., based on a lexicographic ordering) can be employed to avoid the ambiguity.

out before we start deriving the optimality equations. First, the idea of “generating the optimal CDF” is misleading unless we state the goal more carefully. Given any fixed initial configuration (\mathbf{x}, i) and two control policies $\mathbf{a}_1(\cdot)$ and $\mathbf{a}_2(\cdot)$, it is entirely possible (and actually quite common!) that $w_i^{\mathbf{a}_1(\cdot)}(\mathbf{x}, s_1) > w_i^{\mathbf{a}_2(\cdot)}(\mathbf{x}, s_1)$ while $w_i^{\mathbf{a}_1(\cdot)}(\mathbf{x}, s_2) < w_i^{\mathbf{a}_2(\cdot)}(\mathbf{x}, s_2)$. So, which of the resulting CDFs is preferable depends on which threshold is more important: is our priority to minimize the chances of the cumulative cost exceeding s_1 or s_2 ? In this *threshold-specific optimization* setting, we will say that a policy $\mathbf{a}(\cdot)$ is *s-optimal* if $w_i^{\mathbf{a}(\cdot)}(\mathbf{x}, s) \geq w_i^{\mathbf{b}(\cdot)}(\mathbf{x}, s)$ for all allowable control policies $\mathbf{b}(\cdot)$ and all initial configurations (\mathbf{x}, i) .

The second subtlety is in choosing the set of inputs used to define feedback control policies. In threshold-specific optimization, the optimal actions are no longer fully defined by the current state (\mathbf{x}, i) . In addition, they also depend on the cost incurred so far and the desired threshold for the cumulative cost up to the termination. To handle this complication, we add an extra dimension to our state space, defining a new expanded set of control policies $\mathcal{A}_e = \{\mathbf{a} : (\Omega \times \mathcal{M} \times \mathbb{R}) \rightarrow A\}$ and the expanded PDMP dynamics:

$$\begin{aligned} \mathbf{y}'(t) &= \mathbf{f}_{m(t)}(\mathbf{y}(t), \mathbf{a}(\mathbf{y}(t), m(t), c(t))), & (5.37) \\ \mathbf{y}(0) &= \mathbf{x} \in \Omega, \\ c'(t) &= C_{m(t)}(\mathbf{y}(t), \mathbf{a}(\mathbf{y}(t), m(t), c(t))), \\ c(0) &= 0, \\ m(0) &= i \in \mathcal{M}. \end{aligned}$$

Here $c(t)$ represents the total cost incurred so far and $m(t)$ is the current mode, evolving through a continuous-time Markov process on \mathcal{M} . We can now define our new threshold-aware value function:

$$\widehat{w}_i(\mathbf{x}, s) = \sup_{\mathbf{a}(\cdot) \in \mathcal{A}_e} \mathbb{P} \left[\mathcal{J}_i^{\mathbf{a}(\cdot)}(\mathbf{x}) \leq s \right].$$

We note that a similar expansion of state space and policy class could also be used when

defining \widehat{u} , but it would not make any difference due to the linear properties of expectations. In contrast, the c -dependence of policies is essential for writing down the tail-optimality property of \widehat{w}_i 's:

$$\widehat{w}(\mathbf{x}, i, s) = \sup_{\mathbf{a}(\cdot) \in \mathcal{A}_e} \mathbb{E} \left[\widehat{w}(\mathbf{y}(\tau), m(\tau), s - c(\tau)) \right], \quad (5.38)$$

for all $\tau > 0$ sufficiently small to guarantee that $\mathbf{y}(t) \in \Omega \setminus Q$ for all $t \in [0, \tau]$, $\mathbf{a}(\cdot) \in \mathcal{A}_e$. Similarly to the derivation of (5.16), a Taylor expansion of (5.38) yields a weakly-coupled system of nonlinear PDEs satisfied by \widehat{w}_i 's:

$$\max_{\mathbf{a} \in A} \left\{ \nabla \widehat{w}_i(\mathbf{x}, s) \cdot \mathbf{f}_i(\mathbf{x}, \mathbf{a}) - C_i(\mathbf{x}, \mathbf{a}) \frac{\partial \widehat{w}_i}{\partial s}(\mathbf{x}, s) \right\} + \sum_{j \neq i} \lambda_{ij} (\widehat{w}_j(\mathbf{x}, s) - \widehat{w}_i(\mathbf{x}, s)) = 0, \quad (5.39)$$

$$\forall \mathbf{x} \in \Omega \setminus Q, i \in \mathcal{M}, s > 0;$$

with the same initial and boundary conditions previously specified for w_i 's in (5.17) and (5.18). We can also restrict the computational domain for \widehat{w}_i 's (and decrease the numerical diffusion in the discretization) by generalizing equations (5.19)-(5.20) and defining \widehat{s}^0 and \widehat{w}_i^0 's.

5.4.3 Discretization of PDEs and control synthesis

A semi-Lagrangian discretization of (5.39) can be obtained on a grid similarly to the treatment of an uncontrolled case in section 5.2.3:

$$\widehat{W}_{i,\mathbf{k}}^n = \max_{\mathbf{a} \in A} \left\{ \sum_{j=1}^M p_{ij}(\tau) \widehat{W}_j(\mathbf{x}_{\mathbf{k}} + \tau \mathbf{f}_i(\mathbf{x}_{\mathbf{k}}, \mathbf{a}), s_n - \tau C_i(\mathbf{x}_{\mathbf{k}}, \mathbf{a})) \right\}. \quad (5.40)$$

Here $\widehat{W}_{i,\mathbf{k}}^n \approx \widehat{w}_i(\mathbf{x}_{\mathbf{k}}, s_n)$ is a grid function and \widehat{W}_i is its interpolated version defined on $\Omega \times \mathbb{R}$.

Once all \widehat{W}_i 's are computed, they can be used to approximate the optimal control not just on the grid but for all (\mathbf{x}, i, s) by choosing control values from the set

$$\widehat{A}(\mathbf{x}, i, s) = \arg \max_{\mathbf{a} \in A} \left\{ \sum_{j=1}^M p_{ij}(\tau) \widehat{W}_j(\mathbf{x} + \tau \mathbf{f}_i(\mathbf{x}, \mathbf{a}), s - \tau C_i(\mathbf{x}, \mathbf{a})) \right\}. \quad (5.41)$$

Wherever $\nabla\hat{w}_i$ is well-defined, we can also use $A(\mathbf{x}, i, s)$ to denote the arg max set in equation (5.39), with $\hat{A}(\mathbf{x}, i, s)$ interpreted as its grid approximation.

We note that this allows synthesizing a policy (approximately) optimal with respect to *any* desired threshold value. To obtain an \bar{s} -optimal policy $\mathbf{a}(\cdot) \in \mathcal{A}_e$, we would simply need to select $\mathbf{a}(\mathbf{x}, i, c) \in \hat{A}(\mathbf{x}, i, \bar{s} - c)$. However, such policies will be generically non-unique since $\nabla\hat{w}_i(\mathbf{x}, s) = 0$ might hold on a large part of $\Omega \times (0, +\infty)$. For example, there is always a “hopeless region” $H = \{(\mathbf{x}, s) \mid \hat{w}_i(\mathbf{x}, s) = 0, \forall i \in \mathcal{M}\}$ since this equality holds by definition whenever $s < \hat{s}^0(\mathbf{x})$. If C_i ’s do not depend on \mathbf{a} , then $\nabla\hat{w}_i(\mathbf{x}, s) = 0$ implies $A(\mathbf{x}, i, s) = A$. If we start from (\mathbf{x}_0, i_0) such that $\hat{w}_{i_0}(\mathbf{x}_0, \bar{s}) < 1$, then every policy will have a non-zero probability of exceeding the threshold \bar{s} . Which control values are used on H does not change the probability of “success” $(\mathcal{J}_{i_0}^{\mathbf{a}(\cdot)}(\mathbf{x}_0) \leq \bar{s})$, but it can significantly impact the overall CDF of that policy $\mathbf{a}(\cdot) \in \mathcal{A}_e$. In many problems there is also an “unconditionally successful” region $U = \{(\mathbf{x}, s) \mid \hat{w}_i(\mathbf{x}, s) = 1, \forall i \in \mathcal{M}\}$. If $\hat{w}_{i_0}(\mathbf{x}_0, \bar{s}) = 1$ then an optimal policy will never exceed the threshold \bar{s} regardless of the timing of mode switches. If (\mathbf{x}_0, \bar{s}) is in the interior of U , then the success is guaranteed regardless of control values chosen until we reached ∂U , but these choices will generally affect the CDF. To resolve these ambiguities, we use a tie-breaking procedure in defining optimal policies: whenever $\hat{A}(\mathbf{x}, i, s)$ is not a singleton, we select its element that minimizes the expectation. (On H this will coincide with an expectation-optimal policy $\mathbf{a}_*(\mathbf{x}, i)$. But on U this need not be the case since our optimal policy is c -dependent and we need to account for expected values on ∂U .)

Assuming that $\hat{V}_{i,k}^n$ is a grid function approximating the expected outcome and \hat{V}_i is its

interpolated version, we can summarize the computational process as follows:

$$\widehat{W}_{i,\mathbf{k}}^n = \sum_{j=1}^M p_{ij}(\tau) \widehat{W}_j \left(\mathbf{x}_{\mathbf{k}} + \tau \mathbf{f}_i(\mathbf{x}_{\mathbf{k}}, \hat{\mathbf{a}}_{i,\mathbf{k}}^n), s_n - \tau C_i(\mathbf{x}_{\mathbf{k}}, \hat{\mathbf{a}}_{i,\mathbf{k}}^n) \right); \quad (5.42)$$

$$\widehat{V}_{i,\mathbf{k}}^n = \tau C_i(\mathbf{x}_{\mathbf{k}}, \hat{\mathbf{a}}_{i,\mathbf{k}}^n) + \sum_{j=1}^M p_{ij}(\tau) \widehat{V}_j \left(\mathbf{x}_{\mathbf{k}} + \tau \mathbf{f}_i(\mathbf{x}_{\mathbf{k}}, \hat{\mathbf{a}}_{i,\mathbf{k}}^n), s_n - \tau C_i(\mathbf{x}_{\mathbf{k}}, \hat{\mathbf{a}}_{i,\mathbf{k}}^n) \right); \quad (5.43)$$

$$\hat{\mathbf{a}}_{i,\mathbf{k}}^n \in \arg \min_{\mathbf{a} \in \hat{A}(\mathbf{x}_{\mathbf{k}}, i, s_n)} \left\{ \tau C_i(\mathbf{x}_{\mathbf{k}}, \mathbf{a}) + \sum_{j=1}^M p_{ij}(\tau) \widehat{V}_j \left(\mathbf{x}_{\mathbf{k}} + \tau \mathbf{f}_i(\mathbf{x}_{\mathbf{k}}, \mathbf{a}), s_n - \tau C_i(\mathbf{x}_{\mathbf{k}}, \mathbf{a}) \right) \right\}. \quad (5.44)$$

The above description removes almost all ambiguity from the synthesis of threshold-optimal policies, but the arg min in (5.44) might still have multiple elements on a set of measure zero in $\Omega \times \mathcal{S}$. In such rare cases, additional tie-breaking can be used based on another criterion (e.g., a lexicographic ordering).

5.4.4 Numerical experiments

We illustrate these subtleties of policy synthesis with a simple example on a one-dimensional state space $\Omega = [0, 1]$ and two modes, each with its own preferred (faster) direction of motion.

Example 5: More precisely, the control value $\mathbf{a} \in A = \{-1, 1\}$ specifies the chosen direction of motion, and the dynamics are $\mathbf{f}_i(\mathbf{x}, \mathbf{a}) = \mathbf{a} + (-1)^{i-1} \frac{1}{2}$ with $i = 1, 2$. In other words, in mode 1 we can move right with speed $3/2$ and left with speed $1/2$, while in mode 2 it is the opposite. We use $q \equiv 0$ on $Q = \partial\Omega$ and $C_1 \equiv C_2 \equiv 1$, ensuring that the cumulative cost \mathcal{J} is just the time to target. For simplicity, we also use symmetric switching rates $\lambda_{12} = \lambda_{21} = 2$. The resulting optimal policies and the contour plots of $\widehat{w}_i(\mathbf{x}, s)$ are shown in Figures 5.9 and 5.10 respectively. In Figure 5.11 we fix a starting configuration and compare the CDFs of two different policies. The expectation-optimal policy $\mathbf{a}_*(\cdot) \in \mathcal{A}$ is obtained by solving (5.36) and its CDF is then found by solving (5.16). Unfortunately, the

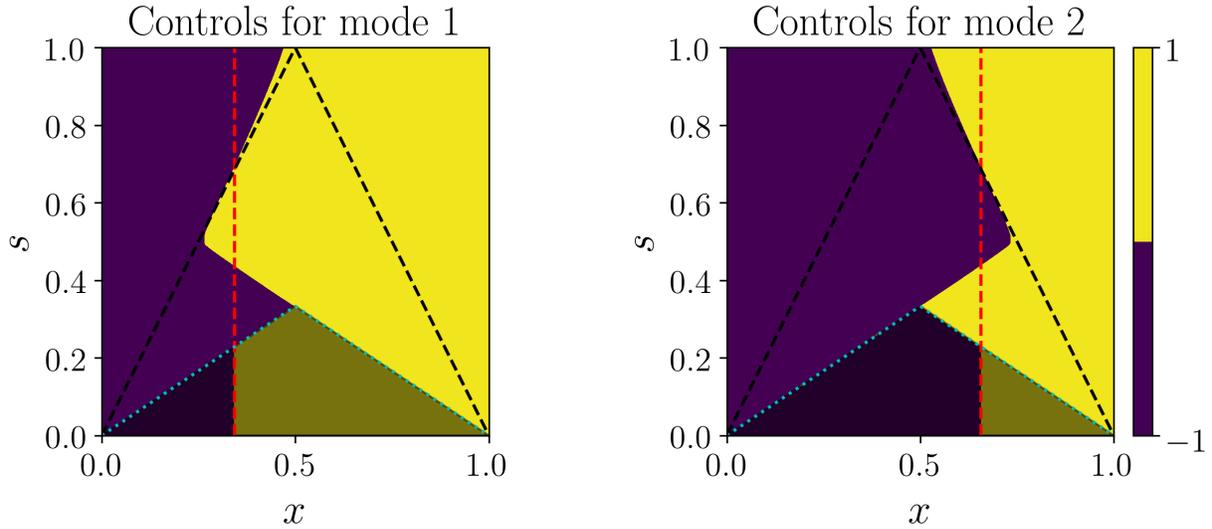


Figure 5.9: Example 5: a map of threshold-optimal control values with position on the horizontal axis and time remaining until the deadline on the vertical axis. The purple color represents the optimal choice of moving to the left, and the yellow color represents the optimal choice of moving to the right. The shaded area with the cyan border represents the “hopeless region” H , where w_i ’s are uniformly zero and the threshold-specific optimal policies coincide with the expectation-optimal policy. The “unconditionally successful” region U is shown above the black dashed line. Under grid refinement, everything in the left part of U becomes purple and everything in the right part of U becomes yellow in both modes. The red dashed vertical lines show the point of direction-switching for the expectation-optimal policy. Computed with $\Delta x = 1.25 \cdot 10^{-4}$, $\Delta s = 0.625 \cdot 10^{-4}$.

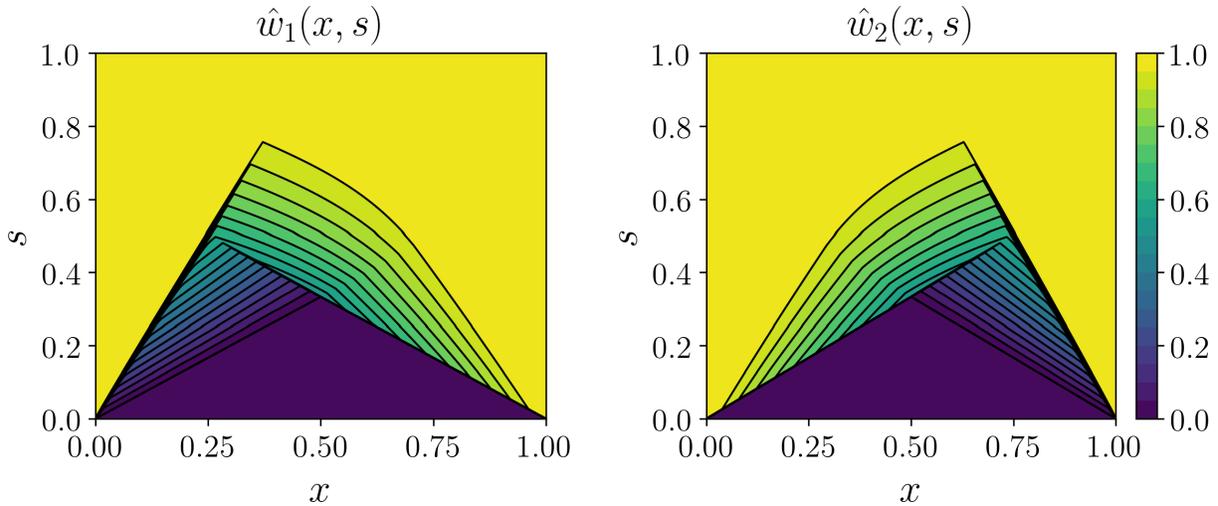


Figure 5.10: Contour plot of $\hat{w}_i(\mathbf{x}, s)$ for Example 5. Computed with $\Delta x = 1.25 \cdot 10^{-4}$, $\Delta s = 0.625 \cdot 10^{-4}$.

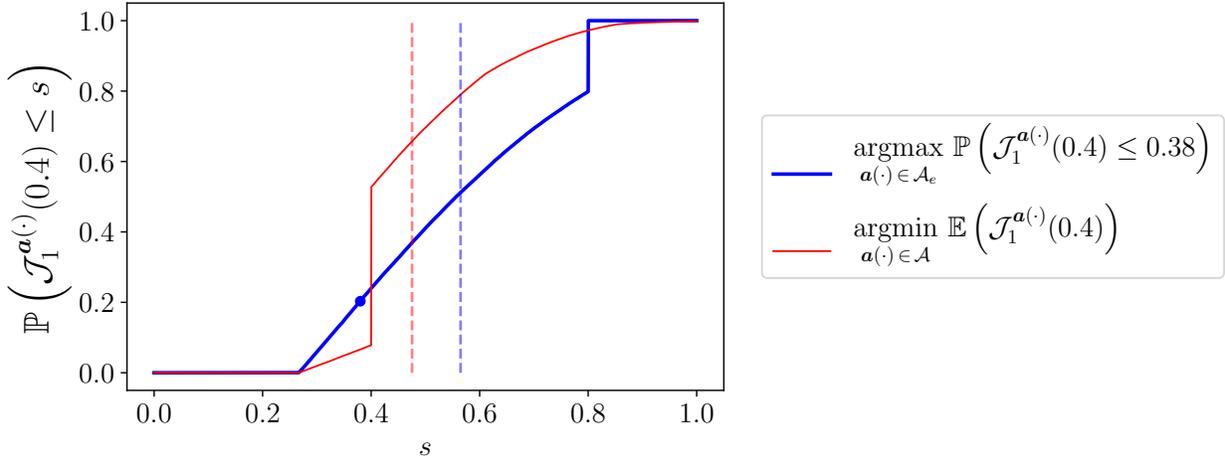


Figure 5.11: Example 5: CDF of an expectation-optimal policy (in red) and CDF of a threshold-specific optimal policy computed for $\bar{s} = 0.38$ (in blue). In both cases, the starting configuration is $(\mathbf{x}_0, i_0) = (0.4, 1)$. The value of the CDF at the threshold $\bar{s} = 0.38$ is marked by a blue dot. The vertical dashed lines indicate the expected value of each policy.

same approach is not available for threshold-specific optimal policies: for an \bar{s} -optimal policy $\mathbf{a}(\cdot) \in \mathcal{A}_e$, there is no reason to expect $\hat{w}_i(\mathbf{x}, s) = \mathbb{P} \left[\mathcal{J}_i^{\mathbf{a}(\cdot)}(\mathbf{x}) \leq s \right]$ unless $s = \bar{s}$. Instead, we approximate their CDF using 100,000 Monte-Carlo simulations³. Not surprisingly, the threshold-specific policy reduces the probability of missing the deadline \bar{s} but at the expense of increasing the expected time to target.

Moreover, threshold-specific optimal policies (and their respective CDFs) may also vary significantly depending on the chosen threshold \bar{s} . To illustrate this, we now consider an example on a two-dimensional state space $\Omega = [0, 1] \times [0, 1]$ with four modes, each with its own faster direction of motion.

Example 6: The control values \mathbf{a} now reside in $A = \{\mathbf{a} \in \mathbb{R}^2 \mid |\mathbf{a}| = 1\}$, and the

³While we do not pursue this alternative here, one could also approximate this CDF by solving the Kolmogorov Forward Equation with initial conditions chosen based on this specific starting configuration (\mathbf{x}, i) .

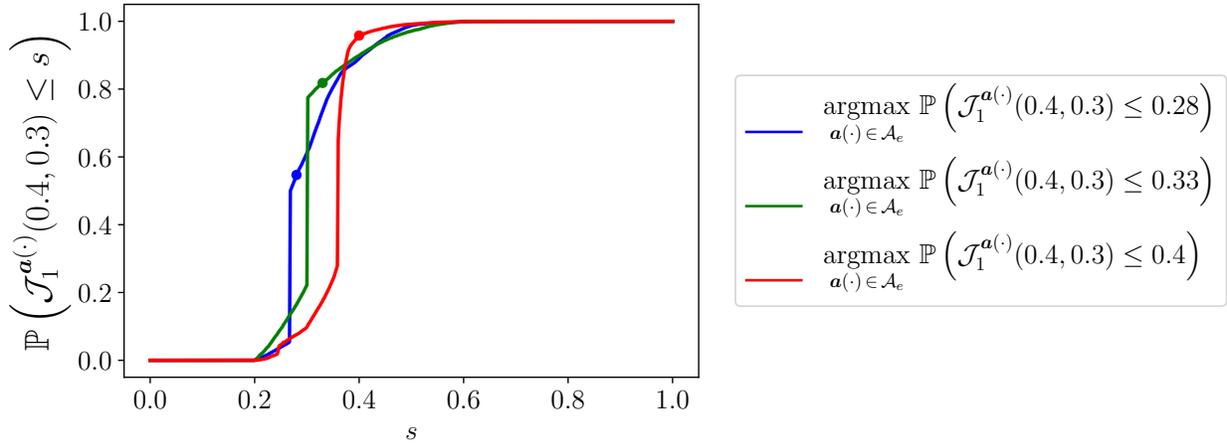


Figure 5.12: Example 6: CDFs of threshold-specific optimal policies computed for $\bar{s} = 0.28$ (in blue), $\bar{s} = 0.33$ (in green), and $\bar{s} = 0.40$ (in red). The value of the CDFs at each threshold are denoted by dots of the corresponding color. In all cases, the starting configuration is $(x_0, y_0, i_0) = (0.4, 0.3, 1)$.

dynamics are given by

$$\mathbf{f}_1(\mathbf{x}, \mathbf{a}) = \mathbf{a} + \begin{bmatrix} -0.5 \\ 0 \end{bmatrix}, \quad \mathbf{f}_2(\mathbf{x}, \mathbf{a}) = \mathbf{a} + \begin{bmatrix} 0 \\ 0.5 \end{bmatrix}, \quad (5.45)$$

$$\mathbf{f}_3(\mathbf{x}, \mathbf{a}) = \mathbf{a} + \begin{bmatrix} 0.5 \\ 0 \end{bmatrix}, \quad \mathbf{f}_4(\mathbf{x}, \mathbf{a}) = \mathbf{a} + \begin{bmatrix} 0 \\ -0.5 \end{bmatrix}. \quad (5.46)$$

Again, we use $q \equiv 0$ on $Q = \partial\Omega$ and $C_i \equiv 1$ for all i , ensuring that the cumulative cost \mathcal{J} is just the time to $\partial\Omega$. The switching rates are $\lambda_{ij} = 1$ for all $i \neq j$. In Figure 5.12, we show the CDFs (each approximated using 10,000 Monte-Carlo simulations) for three different threshold-specific optimal policies with the same starting location. Not surprisingly, each of these policies is strictly better than others with respect to its particular threshold value. The contour plots of $\hat{w}_i(\mathbf{x}, s)$ at various s -slices are also shown in Figure 5.13.

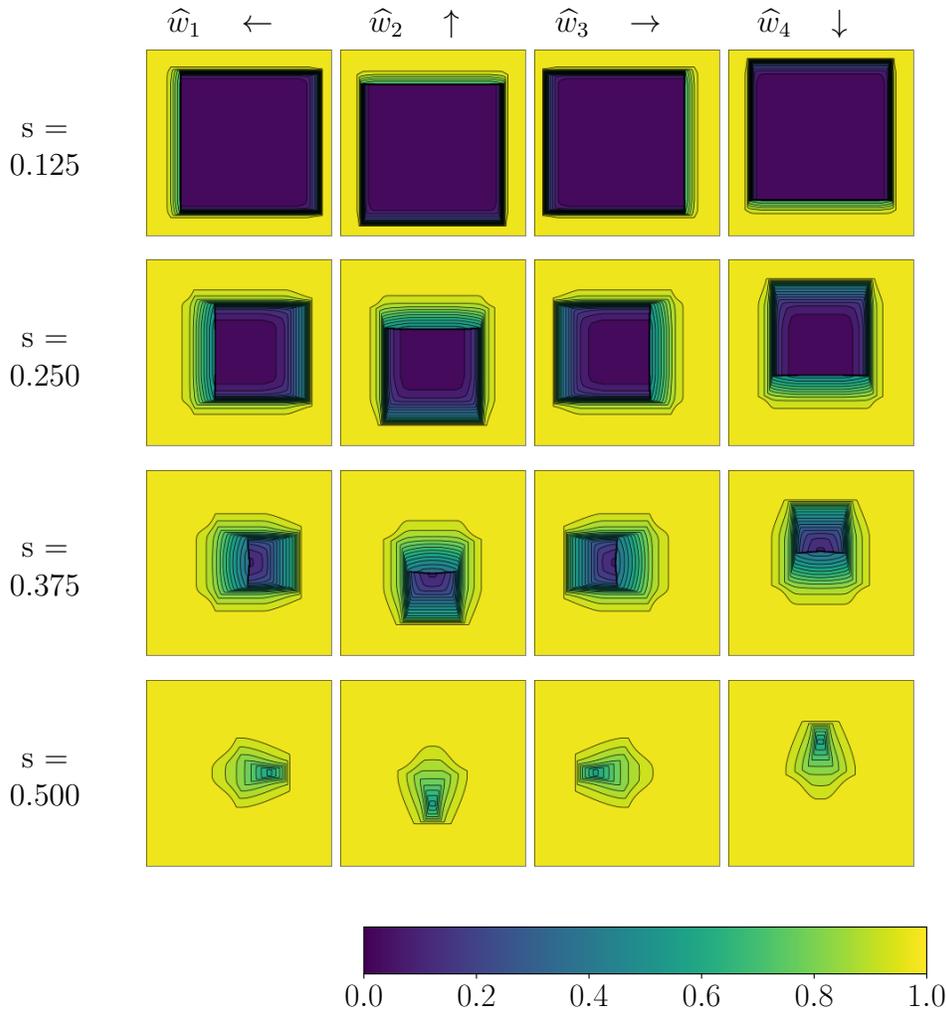


Figure 5.13: Contour plots of $\hat{w}_i(\mathbf{x}, s)$ for Example 6. Transition rate between all modes is $\lambda = 1$. Each subplot contains a snapshot of $\hat{w}_i(\mathbf{x}, s)$. Each row has a fixed s value, and each column has a fixed mode. Dynamics are given by eq. (5.45). Computed on $\Omega \times \mathcal{S} = [0, 1]^3$ with $\Delta x = \Delta y = \Delta s = 0.0025$.

5.5 Conclusion

The versatility of Piecewise-Deterministic Markov Processes (PDMPs) makes them a useful modeling framework for applications with non-diffusive random perturbations. In prior literature on PDMPs, the focus has been mostly on the average/expected performance. Unfortunately, this ignores the practical importance of relatively rare yet truly bad outcomes.

The primary goal of our paper is to address this shortcoming and fully characterize the aleatoric uncertainty in a broad class of discrete and continuous PDMPs. We have accomplished this in section 5.2, approximating the Cumulative Distribution Function (CDF) for their outcomes by solving a system of linear hyperbolic PDEs. Although we did not pursue this here, it would be easy to adapt our approach to compute the CDF of *hitting times* in discrete time Markov chains. In continuous setting, similar ideas could be also extended to stochastic switching in diffusive systems.

For simplicity of exposition, we have assumed the mode-switching rates λ_{ij} to be constant, but it would be trivial to extend our framework to state-dependent switching rates $\lambda_{ij}(\mathbf{x})$. The case of λ_{ij} 's deterministically evolving in time can be treated similarly by increasing the dimension of our state space. But random changes in rates present a more serious challenge, which is also related to handling model uncertainties. The latter is particularly important in PDMPs since in many practical applications these rates are not known precisely and are instead estimated based on historical data. It is thus useful to characterize the *range* of possible CDFs – a task accomplished in section 5.3, where tight CDF bounds are developed under the assumption that each (state-independent) transition rate λ_{ij} has known bounds but does not necessarily remain constant throughout the process.

Finally, in section 5.4 we have extended our methods to *controlled* PDMPs, showing how to maximize the probability of not exceeding a specific cumulative cost threshold \bar{s} . Our approach is also related to the Stochastic On-Time Arrival (SOTA) formulation, developed in discrete setting by transportation engineers to optimize the routing on stochastic networks [52, 106, 49]. In the context of SOTA, there is only one “mode,” but the running cost is random. While we do not pursue it here, our method can be similarly adapted to optimize the CDF of general Markov Decision Processes, where the running cost might be deterministic, but the successor node is random.

Several generalizations of the described methods will broaden their appeal to practitioners. First, all PDMPs considered here were *exit-time* problems, with the process terminating as soon as the system enters a specific subset Q of the state space Ω . It will be easy to extend our approach to finite horizon problems, but the extensions to infinite horizon (with time discounting of running cost) or ergodic (time-averaged cumulative cost) problems will be more challenging.

Second, there are many potential ways to improve the accuracy and computational efficiency. Our approach relies on solving weakly coupled systems of hyperbolic PDEs, whose solutions are typically piecewise continuous. While the described implementation is based on a first-order accurate semi-Lagrangian discretization, it would be useful to replace these with higher-order accurate schemes [51]. Our preliminary experimental results based on ENO/WENO [114] spatial discretization in one dimension seem promising, but we have decided to omit them here due to length constraints. We have also developed a technique restricting the computational domain by pre-computing the minimal attainable cumulative cost. In controlled PDMPs with an “unconditionally successful” region, further domain restriction techniques might be used to maximize the probability of desirable outcomes while also imposing a hard constraint on the worst-case performance. This would mirror the approach previously developed for routing on stochastic networks [49].

For controlled processes, another interesting challenge is to carefully evaluate all trade-offs between conflicting objectives. This is usually done by computing non-dominated (or *Pareto optimal* controls), for which any improvement in one of the objectives must come at the cost of decreased performance based on some other objective(s). With PDMPs, the natural objectives would include traditional minimization of the expected cumulative cost and maximizing the probability of not exceeding a threshold (possibly for several different threshold values). In the fully deterministic case, several methods for multiobjective dynamic

programming have been developed in the last ten years [81, 60, 46]. It would be quite useful but challenging to extend these to the piecewise-deterministic setting.

It would be also very interesting to explore additional notions of robustness for PDMPs. Our approach can be viewed as a dual of optimizing the Value-at-Risk (VaR), in which the goal is to minimize a specific percentile of the random outcome. We minimize the probability of exceeding a specific threshold, but similarly to VaR, we provide no guarantees on how bad the outcomes can be once that threshold is exceeded. The Conditional Value-at-Risk (CVaR) is an extended risk measure which addresses this limitation. A method based on CVaR optimization has been developed for Markov Decision Processes in [34]. It would be useful to extend it to PDMPs and compare with the threshold-optimal policies described here.

CHAPTER 6

CONCLUSIONS

This thesis is motivated by a number of problems at the intersection of optimal control theory and game theory. In Chapter 2, we proved sufficient conditions for guaranteeing that pedestrian flow models are “internally consistent” for both inter-crowd and intra-crowd models. For the inter-crowd models we were even able to prove sufficient conditions for the uniqueness of the Nash Equilibrium. In Chapter 3, we presented efficient numerical methods for finding the Nash Equilibrium of a class of time-dependent Surveillance-Evasion games. Chapter 4 discussed control-theoretic models of environmental crime, presenting two different models that very carefully consider the optimal control problem faced by the extractor. Finally, Chapter 5 focused on uncertainty quantification and uncertainty management for piecewise-deterministic Markov processes, showing how to efficiently compute and even optimize the cumulative distribution function of possible outcomes.

However, all of these problems can be extended in any number of interesting and useful ways, leaving open many avenues for further research. One common theme is that the type and quantity of information available to the agents drastically changes the optimal control problem they face. In fact, one of the most significant distinctions between classical surveillance-evasion games [87, 88] and the games in [56] and Chapter 3 is that the players are now unaware of the exact position of the other players, and only know their set of possible strategies. Moreover, in Chapter 2, we assumed that individual pedestrians used “stroboscopic” planning, where they could not anticipate future movements of other pedestrians, meaning that their path-planning assumed that the current crowd density would remain the same for all time, an assumption that clearly would not hold in practice. On the other hand, allowing pedestrians to anticipate the future actions of others would introduce a number of new analytical and computational challenges. In the case of many interacting agents each

selfishly pursuing their own goals, this could be treated in the framework of Mean Field Games, initially introduced in [85, 69]. However, the well-posedness of solutions to the resulting MFG system is an open problem, as well as the development of efficient and stable numerical methods for computing the solution. Similar MFG models can also be derived by extending the results of Chapter 3 to the case of a large crowd of selfish Evaders, where each Evader’s speed of motion f depends on the density of the crowd ρ at their location \mathbf{x} , as in the pedestrian models of Chapter 2.

However, even in the case of finitely many agents, there still remain open analytical questions. For two-player zero-sum games, it can be shown that any Nash Equilibria have the same payoff, and moreover that any mix of Nash Equilibrium strategies still results in a Nash Equilibrium. Unfortunately, this property no longer holds in general for non-zero-sum games, where there can be multiple Nash Equilibria with different payoffs. This poses a serious problem, as the game no longer has value, and even reaching a Nash Equilibrium solution may require cooperation between the two players [21]. In Chapter 2, we were able to prove sufficient conditions for guaranteeing uniqueness of Nash Equilibria for a specific class of non-zero-sum differential games used in pedestrian flow modeling. A natural question is then whether it is possible to prove general criteria for non-zero-sum differential games, under which the Nash Equilibrium for is guaranteed to be unique.

On a more practical note, the environmental crime models proposed in Chapter 4 rely on a number of unrealistic assumptions. In particular, the assumption of a “single period” Stackelberg game does not hold in reality, where both the authorities and extractors change their strategies as the protected area degrades. Our models also assumed that the detection rate ψ is specified in advance, and focused only on the control problems faced by the extractors, but it would be significantly more useful and challenging to optimize these surveillance strategies directly.

The results of Chapter 5 could also be expanded in a number of ways. Our approach considered only finite horizon optimal control problems, but it would be useful to extend these results to infinite horizon or ergodic problems. Another interesting challenge would be to carefully consider trade-offs between conflicting objectives, extending existing methods for multiobjective dynamic programming.

APPENDIX A

**VELOCITY PROFILE FOR LINEAR DENSITY INTRA-CROWD
ANISOTROPY**

In this section, we derive a formula for the velocity profile of the model used in [41], under the assumption of a linear density distribution within the sensing region. We recall that for this model of intra-crowd anisotropy, the interactional velocity is given by:

$$\mathbf{w}_i(\mathbf{x}, \mathbf{u}) = \int_{S(\mathbf{x}, \mathbf{u}) \cap \Omega} \mathcal{F}(\mathbf{y} - \mathbf{x}) \rho(t, \mathbf{y}) d\mathbf{y}$$

The original formulation, which can be found in Equation (2.12) in Section 2.3.3, involved a cutoff value C in the definition of $\mathcal{F}(\mathbf{y})$. To simplify the analysis we shall take $C \rightarrow \infty$ and use $\mathcal{F}(\mathbf{y}) = \frac{-F\mathbf{y}}{|\mathbf{y}|^2}$. Suppose that $\mathbf{u} = (\cos(\theta), \sin(\theta))$, $S(\mathbf{x}, \mathbf{u})$ is a sector of radius R and angle α centered in the direction \mathbf{u} . Furthermore, suppose \mathbf{x} is far enough away from $\partial\Omega$ so that the sensing region $S(\mathbf{x}, \mathbf{u}) \subset \Omega$ and $\rho(\mathbf{x}) > 0$ on $S(\mathbf{x}, \mathbf{u})$ for all choices of \mathbf{u} . Then fixing the time $t > 0$, we may assume WLOG that our density ρ is of the form $\rho(x, y) = \rho_0 + \rho_x x$, with $\rho_x > 0$. Our formula for \mathbf{w}_i then becomes:

$$\mathbf{w}_i = \int_{S(\theta)} \frac{-F\mathbf{y}}{|\mathbf{y}|^2} (\rho_0 + \rho_x x) d\mathbf{y}$$

which we can write in polar coordinates as:

$$\mathbf{w}_i = \int_{\theta-\alpha/2}^{\theta+\alpha/2} \int_0^R \frac{-F}{r^2} \begin{pmatrix} r \cos(\gamma) \\ r \sin(\gamma) \end{pmatrix} (\rho_0 + \rho_x r \cos(\gamma)) r dr d\gamma$$

After computing this integral, we get that the overall velocity $\mathbf{v} = \mathbf{w}_i + \mathbf{u}$ is given as:

$$\mathbf{v}(\theta) = \begin{pmatrix} \frac{-F\rho_x R^2 \alpha}{4} \\ 0 \end{pmatrix} + (1 - 2F\rho_0 R \sin(\alpha/2)) \begin{pmatrix} \cos(\theta) \\ \sin(\theta) \end{pmatrix} - \frac{F\rho_x R^2 \sin(\alpha)}{4} \begin{pmatrix} \cos(2\theta) \\ \sin(2\theta) \end{pmatrix} \quad (\text{A.0.1})$$

As long as the pedestrian density is linear within the sensing region, this gives an explicit form for $\mathbf{v}(\theta)$.

APPENDIX B

CONVEXITY FOR LINEAR DENSITY INTRA-CROWD ANISOTROPY

For convenience, we shall now use $C_1 = (1 - 2F\rho_0 R \sin(\alpha/2))$, $C_2 = F\rho_x R^2 \sin(\alpha)$, and $C_3 = F\rho_x R^2 \alpha$. With this notation, we can write our explicit form for \mathbf{v} from Section A as:

$$\mathbf{v}(\theta) = \begin{pmatrix} v_1(\theta) \\ v_2(\theta) \end{pmatrix} = \begin{pmatrix} -\frac{C_3}{4} \\ 0 \end{pmatrix} + C_1 \begin{pmatrix} \cos(\theta) \\ \sin(\theta) \end{pmatrix} - \frac{C_2}{4} \begin{pmatrix} \cos(2\theta) \\ \sin(2\theta) \end{pmatrix}. \quad (\text{B.0.1})$$

We first note that as long as $0 < \alpha < \pi$, i.e. a pedestrian is only slowed down by other pedestrians in front of them, then we have $C_3 > C_2 > 0$.

We will need to put two different conditions on our parameters so that our solutions can be logically consistent. First, the component of the overall velocity $\mathbf{v}(\theta)$ in the direction of the behavioral velocity $\mathbf{u} = (\cos(\theta), \sin(\theta))$ is positive for all values of θ . We calculate this to happen when:

$$C_1 - \frac{C_2 + C_3}{4} \cos(\theta) > 0, \quad \text{for all } \theta \in [0, 2\pi].$$

Since C_2 and C_3 are always positive, this is equivalent to the condition:

$$C_1 - \frac{C_2 + C_3}{4} > 0. \quad (\text{B.0.2})$$

We also need a sufficient condition for the strict convexity of the velocity profile. As with the inter-crowd case, our velocity profile is given as a parametrized curve, so to check that it is strictly convex, we must make sure the sign of the cross product $\mathbf{T} \times \mathbf{N}$ does not change. This condition becomes:

$$v_1' v_2'' - v_2' v_1'' > 0, \quad \text{for all } \theta \in [0, 2\pi]. \quad (\text{B.0.3})$$

We note that Equation (2.10) is a special case of Equation (B.0.3) when $v_1(\theta) = f(\theta) \cos(\theta)$ and $v_2(\theta) = f(\theta) \sin(\theta)$. Plugging B.0.1 into B.0.3 reduces the latter to

$$C_1^2 + \frac{C_2^2}{2} + \frac{3}{2} C_1 C_2 \cos(\theta) > 0, \quad \text{for all } \theta \in [0, 2\pi]. \quad (\text{B.0.4})$$

We recall that $C_2, C_3 > 0$. Then as long as Equation B.0.2 is satisfied, we have $C_1 > 0$ as well. Looking at Equation B.0.4, we see that the left hand side is minimized when $\theta = \pi$, so the sufficient condition for strict convexity becomes:

$$C_1^2 + \frac{C_2^2}{2} - \frac{3}{2}C_1C_2 = \frac{(C_1 - C_2)(2C_1 - C_2)}{2} > 0. \quad (\text{B.0.5})$$

The case we want to avoid is when we have a non-strictly-convex velocity profile that contains the origin, i.e. when Equation (B.0.2) is satisfied and Equation (B.0.5) is not. This can only happen when:

$$C_2 > C_1 > \frac{C_2 + C_3}{4}.$$

Writing this in terms of the original parameters:

$$F\rho_x R^2 \sin(\alpha) > 1 - 2F\rho_0 R \sin(\alpha/2) > F\rho_x R^2 \frac{(\alpha + \sin(\alpha))}{4}.$$

These inequalities can only be satisfied if: $\sin(\alpha) > \frac{\alpha + \sin(\alpha)}{4}$, i.e.

$$\alpha < 3 \sin(\alpha),$$

which is true for $0 < \alpha < \approx 130.57^\circ$. This means that if $\alpha > \approx 130.57^\circ$ (as in [41]) and the velocity profile contains the origin, then the velocity profile will always be strictly convex.

BIBLIOGRAPHY

- [1] Shahnawaz Ahmed, Stanley Bak, Joyce McLaughlin, and Daniel Renzi. A third order accurate fast marching method for the eikonal equation in two dimensions. *SIAM Journal on Scientific Computing*, 33(5):2402–2420, 2011.
- [2] Ramakrishna Akella and P Roshan Kumar. Optimal control of production rate in a failure prone manufacturing system. *Automatic Control, IEEE Transactions on*, 31(2):116–126, 1986.
- [3] Heidi J Albers. Spatial modeling of extraction and enforcement in developing country protected areas. *Resource and Energy Economics*, 32(2):165–179, 2010.
- [4] Ken Alton and Ian M Mitchell. An ordered upwind method with precomputed stencil and monotone node acceptance for solving static convex Hamilton-Jacobi equations. *Journal of Scientific Computing*, 51(2):313–348, 2012.
- [5] Debora Amadori and M. Di Francesco. The one-dimensional Hughes model for pedestrian flow: Riemann-type solutions. *Acta Mathematica Scientia*, 32(1):259 – 280, 2012.
- [6] June Andrews and Alexander Vladimírsky. Deterministic control of randomly-terminated processes. *Interfaces and Free Boundaries*, 16(1):1–40, 2014.
- [7] M Annunziato. Analysis of upwind method for piecewise deterministic markov processes. *Computational Methods in Applied Mathematics*, 8(1):3–20, 2008.
- [8] David J Arnold, Dayne Fernandez, Ruizhe Jia, Christian Parkinson, Deborah Tonne, Yotam Yaniv, Andrea L Bertozzi, and Stanley J Osher. Modeling environmental crime in protected areas using the level set method. *SIAM Journal on Applied Mathematics*, 79(3):802–821, 2019.
- [9] Martino Bardi and Italo Capuzzo-Dolcetta. *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations*. Springer, 2008.
- [10] Martino Bardi, Maurizio Falcone, and Pierpaolo Soravia. Numerical methods for pursuit-evasion games via viscosity solutions. In Martino Bardi, T. E. S. Raghavan, and T. Parthasarathy, editors, *Stochastic and Differential Games: Theory and Numerical Methods*, pages 105–175, Boston, MA, 1999. Birkhäuser Boston.
- [11] Guy Barles and Panagiotis E Souganidis. Convergence of approximation schemes for fully nonlinear second order equations. *Asymptotic analysis*, 4(3):271–283, 1991.

- [12] Timothy J Barth and James A Sethian. Numerical schemes for the Hamilton–Jacobi and level set equations on triangulated domains. *Journal of Computational Physics*, 145(1):1–40, 1998.
- [13] Tamer Başar and Pierre Bernhard. *H-infinity optimal control and related minimax design problems: a dynamic game approach*. Springer Science & Business Media, 2008.
- [14] Amir Beck. *First-Order Methods in Optimization*, volume 25. SIAM, 2017.
- [15] Michael Beil, Sebastian Lück, Frank Fleischer, Stéphanie Portet, Wolfgang Arendt, and Volker Schmidt. Simulating the formation of keratin filament networks by a piecewise-deterministic Markov process. *Journal of theoretical biology*, 256(4):518–532, 2009.
- [16] Nicola Bellomo and Christian Dogbe. On the modeling of traffic and crowds: A survey of models, speculations, and perspectives. *SIAM Review*, 53(3):409–463, 2011.
- [17] J-D Benamou, Songting Luo, and Hongkai Zhao. A compact upwind second order scheme for the eikonal equation. *Journal of Computational Mathematics*, pages 489–516, 2010.
- [18] T Bielecki and PR Kumar. Optimality of zero-inventory policies for unreliable manufacturing systems. *Operations research*, 36(4):532–541, 1988.
- [19] Michelle Boué and Paul Dupuis. Markov chain approximations for deterministic control problems with affine dynamics and quadratic cost in the control. *SIAM Journal on Numerical Analysis*, 36(3):667–695, 1999.
- [20] El Kébir Boukas, Alain Haurie, and Philippe Michel. An optimal control problem with a random stopping time. *Journal of optimization theory and applications*, 64(3):471–480, 1990.
- [21] Alberto Bressan. Noncooperative differential games. *Milan Journal of Mathematics*, 79(2):357–427, 2011.
- [22] Alberto Bressan and Benedetto Piccoli. *Introduction to the mathematical theory of control*, volume 2. American institute of mathematical sciences, 2007.
- [23] Alberto Bressan and Fabio S. Priuli. Infinite horizon noncooperative differential games. *Journal of Differential Equations*, 227(1):230 – 257, 2006.
- [24] Martin Burger, Marco Di Francesco, Peter Markowich, and Marie-Therese Wolfram.

- Mean field games with nonlinear mobilities in pedestrian dynamics. *Discrete and Dynamical Systems - B*, 19(5):1311–1333, 2014.
- [25] Simone Cacace, Emiliano Cristiani, and Maurizio Falcone. Numerical approximation of Nash equilibria for a class of non-cooperative differential games. In L. Petrosjan and V. Mazalov, editors, *Game Theory and Applications*, volume 16. Nova Publishers, New York, 2013.
- [26] Jose A Carrillo, Stephan Martin, and Marie-Therese Wolfram. An improved version of the Hughes model for pedestrian flow. *Mathematical Models and Methods in Applied Sciences*, 26(04):671–697, 2016.
- [27] E. Cartee, L. Lai, Q. Song, and A. Vladimirsky. Time-dependent surveillance-evasion games. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 7128–7133, 2019.
- [28] Elliot Cartee, Antonio Farah, April Nellis, Jacob van Hook, and Alexander Vladimirsky. Quantifying and managing uncertainty in piecewise-deterministic markov processes. *arXiv preprint arXiv:2008.00555*, 2020.
- [29] Elliot Cartee and Alexander Vladimirsky. Anisotropic challenges in pedestrian flow modeling. *Communications in Mathematical Sciences*, 16(4):1067–1093, 2018.
- [30] Elliot Cartee and Alexander Vladimirsky. Control-theoretic models of environmental crime. *SIAM Journal on Applied Mathematics*, 80(3):1441–1466, 2020.
- [31] A. Chacon and A. Vladimirsky. Fast two-scale methods for eikonal equations. *SIAM Journal on Scientific Computing*, 34(2):A547–A578, 2012.
- [32] A. Chacon and A. Vladimirsky. A parallel two-scale method for eikonal equations. *SIAM Journal on Scientific Computing*, 37(1):A156–A180, 2015.
- [33] Julien Chiquet, Nikolaos Limnios, and Mohamed Eid. Piecewise deterministic Markov processes applied to fatigue crack growth modelling. *Journal of Statistical Planning and Inference*, 139(5):1657 – 1667, 2009. Special Issue on Degradation, Damage, Fatigue and Accelerated Life Models in Reliability Testing.
- [34] Yinlam Chow, Aviv Tamar, Shie Mannor, and Marco Pavone. Risk-sensitive and robust decision-making: a CVaR optimization approach. In *Advances in Neural Information Processing Systems*, pages 1522–1530, 2015.

- [35] Z. Clawson, A. Chacon, and A. Vladimírsky. Causal domain restriction for Eikonal equations. *SIAM Journal on Scientific Computing*, 36(5):A2478–A2505, 2014.
- [36] Zachary Clawson, Xuchu Ding, Brendan Englot, Thomas A Frewen, William M Sisson, and Alexander Vladimírsky. A bi-criteria path planning algorithm for robotics applications. *arXiv preprint*, 2015.
- [37] Rinaldo M Colombo and Magali Lécureux-Mercier. Nonlocal crowd dynamics models for several populations. *Acta Mathematica Scientia*, 32(1):177–196, 2012.
- [38] Michael G Crandall and Pierre-Louis Lions. Viscosity solutions of Hamilton-Jacobi equations. *Transactions of the American Mathematical Society*, 277(1):1–42, 1983.
- [39] Michael G Crandall and Andrew Majda. Monotone difference approximations for scalar conservation laws. *Mathematics of Computation*, 34(149):1–21, 1980.
- [40] Emiliano Cristiani and Maurizio Falcone. Fast Semi-Lagrangian schemes for the eikonal equation and applications. *SIAM Journal on Numerical Analysis*, 45(5):1979–2011, 2007.
- [41] Emiliano Cristiani, Fabio S Priuli, and Andrea Tosin. Modeling rationality to control self-organization of crowds: an environmental approach. *SIAM Journal on Applied Mathematics*, 75(2):605–629, 2015.
- [42] Indraneel Das and John E Dennis. A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems. *Struct. Optim.*, 14(1):63–69, 1997.
- [43] A. Dassios and H. Zhao. A dynamic contagion process. *Advanced in Applied Probability*, 43(3):814–846, 2011.
- [44] Mark H. A. Davis and Mohammad Farid. *Piecewise-Deterministic Processes and Viscosity Solutions*, pages 249–268. Birkhäuser Boston, Boston, MA, 1999.
- [45] MHA Davis. Control of piecewise-deterministic processes via discrete-time dynamic programming. In *Stochastic Differential Systems*, pages 140–150. Springer, 1986.
- [46] Anna Désilles and Hasnaa Zidani. Pareto front characterization for multiobjective optimal control problems using Hamilton–Jacobi approach. *SIAM Journal on Control and Optimization*, 57(6):3884–3910, 2019.

- [47] Edsger W Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959.
- [48] J. S. Drake, J. L. Schofer, and A. D. May. A statistical analysis of speed-density hypotheses. In L. Edie, R. Herman, and R. Rothery, editors, *Proceedings of the Third International Symposium on the Theory of Traffic Flow*, 1967.
- [49] Stefano Ermon, Carla P. Gomes, Bart Selman, and Alexander Vladimirsky. Probabilistic planning with non-linear utility functions and worst-case guarantees. In Wiebe van der Hoek, Lin Padgham, Vincent Conitzer, and Michael Winikoff, editors, *AA-MAS*, pages 965–972. IFAAMAS, 2012.
- [50] Joep H.M. Evers, Razvan C. Fetecau, and Lenya Ryzhik. Anisotropic interactions in a first-order aggregation model. *Nonlinearity*, 28(8):2847, 2015.
- [51] Maurizio Falcone and Roberto Ferretti. *Semi-Lagrangian Approximation Schemes for Linear and Hamilton-Jacobi Equations*. Society for Industrial and Applied Mathematics, Philadelphia, PA, dec 2013.
- [52] Yueyue Fan and Yu Nie. Optimal routing for maximizing the travel time reliability. *Networks and Spatial Economics*, 6(3):333–344, Sep 2006.
- [53] Fei Fang, Thanh Hong Nguyen, Rob Pickles, Wai Y Lam, Gopalasamy R Clements, Bo An, Amandeep Singh, Brian C Schwedock, Milind Tambe, and Andrew Lemieux. PAWS - a deployed game-theoretic application to combat poaching. *AI Magazine*, 38(1):23–36, 2017.
- [54] Wendell H Fleming and Halil Mete Soner. *Controlled Markov processes and viscosity solutions*, volume 25. Springer Science & Business Media, 2006.
- [55] Marco Di Francesco, Peter A. Markowich, Jan-Frederik Pietschmann, and Marie-Therese Wolfram. On the Hughes’ model for pedestrian flow: The one-dimensional case. *Journal of Differential Equations*, 250(3):1334 – 1362, 2011.
- [56] Marc Aurèle Gilles and Alexander Vladimirsky. Evasive path planning under surveillance uncertainty. *Dynamic Games and Applications*, pages 1–26, 2019.
- [57] Irving L Glicksberg. A further generalization of the Kakutani fixed point theorem, with application to Nash equilibrium points. *Proceedings of the American Mathematical Society*, 3(1):170–174, 1952.

- [58] Diogo A. Gomes and João Saúde. Mean field games models—a brief survey. *Dynamic Games and Applications*, 4(2):110–154, 2014.
- [59] Olivier Guéant, Jean-Michel Lasry, and Pierre-Louis Lions. Mean field games and applications. pages 205–266, 2011.
- [60] Alexis Guigue. Approximation of the Pareto optimal set for multiobjective optimal control problems using viability kernels. *ESAIM: Control, Optimisation and Calculus of Variations*, 20(1):95–115, 2014.
- [61] Flurin S. Hänseler, William H. K. Lam, Michel Bierlaire, Gael Lederrey, and Marija Nikolić. A dynamic network loading model for anisotropic and congested pedestrian flows. Technical Report TRANSP-OR 151026, Transport and Mobility Laboratory, Ecole Polytechnique Fédérale de Lausanne, October 2015.
- [62] Peter E Hart, Nils J Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.
- [63] Dirk Hartmann and Peter Hasel. Efficient dynamic floor field methods for microscopic pedestrian crowd simulations. *Commun. Comput. Phys.*, 16(1):264–286, July 2014.
- [64] A. Haurie and F. Moresino. A stochastic control model of economic growth with environmental disaster prevention. *Automatica*, 42(8):1417–1428, 2006.
- [65] Alain Haurie. A multigenerational game model to analyze sustainable development. *Annals of Operations Research*, 137(1):369–386, Jul 2005.
- [66] Serge Hoogendoorn and Piet HL Bovy. Simulation of pedestrian flows by optimal control and differential games. *Optimal Control Applications and Methods*, 24(3):153–172, 2003.
- [67] Serge P Hoogendoorn and Piet HL Bovy. Dynamic user-optimal assignment in continuous time and space. *Transportation Research Part B: Methodological*, 38(7):571–592, 2004.
- [68] Ling Huang, S.C. Wong, Mengping Zhang, Chi-Wang Shu, and William H.K. Lam. Revisiting Hughes’ dynamic continuum model for pedestrian flow and the development of an efficient solution algorithm. *Transportation Research Part B: Methodological*, 43(1):127–141, January 2009.
- [69] Minyi Huang, Roland P Malhamé, Peter E Caines, et al. Large population stochastic

- dynamic games: closed-loop McKean-Vlasov systems and the Nash certainty equivalence principle. *Communications in Information & Systems*, 6(3):221–252, 2006.
- [70] Roger L Hughes. A continuum theory for the flow of pedestrians. *Transportation Research Part B: Methodological*, 36(6):507–535, 2002.
- [71] Yan-Qun Jiang, Bo-Kui Chen, Bing-Hong Wang, Weng-Fai Wong, and Bing-Yang Cao. Extended social force model with a dynamic navigation field for bidirectional pedestrian flow. *Frontiers of Physics*, 12(5):124502, 2017.
- [72] Yan-Qun Jiang, Ren-Yong Guo, Fang-Bao Tian, and Shu-Guang Zhou. Macroscopic modeling of pedestrian flow based on a second-order predictive dynamic model. *Applied Mathematical Modelling*, 40(23):9806–9820, 2016.
- [73] Yan-Qun Jiang, Wei Zhang, and Shu-Guang Zhou. Comparison study of the reactive and predictive dynamic models for pedestrian flow. *Physica A: Statistical Mechanics and its Applications*, 441:51–61, 2016.
- [74] Yanqun Jiang, Sze Chun Wong, Peng Zhang, Ruxun Liu, Yali Duan, and Keechoo Choi. Numerical simulation of a continuum model for bi-directional pedestrian flow. *Applied Mathematics and Computation*, 218(10):6135–6143, 2012.
- [75] Yanqun Jiang, Tao Xiong, SC Wong, Chi-Wang Shu, Mengping Zhang, Peng Zhang, and William HK Lam. A reactive dynamic continuum user equilibrium model for bi-directional pedestrian flows. *Acta Mathematica Scientia*, 29(6):1541–1555, 2009.
- [76] Yanqun Jiang, Shu-Guang Zhou, and Fang-Bao Tian. A higher-order macroscopic model for bi-direction pedestrian flow. *Physica A: Statistical Mechanics and its Applications*, 425:69–78, 2015.
- [77] Matthew Paul Johnson, Fei Fang, and Milind Tambe. Patrol strategies to maximize pristine forest area. In *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.
- [78] C. Kao, S. Osher, and Y. Tsai. Fast sweeping methods for static Hamilton–Jacobi equations. *SIAM Journal on Numerical Analysis*, 42(6):2612–2632, 2005.
- [79] Debarun Kar, Benjamin Ford, Shahrzad Gholami, Fei Fang, Andrew Plumptre, Milind Tambe, Margaret Driciru, Fred Wanyama, Aggrey Rwetsiba, Mustapha Nsubaga, et al. Cloudy with a chance of poaching: Adversary behavior modeling and forecasting with real-world poaching data. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, pages 159–167. International Foundation for Autonomous Agents and Multiagent Systems, 2017.

- [80] Ron Kimmel and James A Sethian. Fast marching methods on triangulated domains. *Proceedings of the National Academy of Science*, 95:8341–8435, 1998.
- [81] Ajeet Kumar and Alexander Vladimirovsky. An efficient method for multiobjective optimal control and optimal control subject to integral constraints. *Journal of Computational Mathematics*, pages 517–551, 2010.
- [82] Aimé Lachapelle and Marie-Therese Wolfram. On a mean field game approach modeling congestion and aversion in pedestrian crowds. *Transportation research part B: methodological*, 45(10):1572–1589, 2011.
- [83] Jean-Michel Lasry and Pierre-Louis Lions. Jeux à champ moyen. I—le cas stationnaire. *Comptes Rendus Mathématique*, 343(9):619–625, 2006.
- [84] Jean-Michel Lasry and Pierre-Louis Lions. Jeux à champ moyen. II—horizon fini et contrôle optimal. *Comptes Rendus Mathématique*, 343(10):679–684, 2006.
- [85] Jean-Michel Lasry and Pierre-Louis Lions. Mean field games. *Japanese journal of mathematics*, 2(1):229–260, 2007.
- [86] Peter D Lax. Weak solutions of nonlinear hyperbolic equations and their numerical computation. *Communications on pure and applied mathematics*, 7(1):159–193, 1954.
- [87] J Lewin and JV Breakwell. The surveillance-evasion game of degree. *JOTA*, 16(3-4):339–353, 1975.
- [88] J Lewin and GJ Olsder. Conic surveillance evasion. *JOTA*, 27(1):107–125, 1979.
- [89] Dan Li, Shengqiang Liu, and Jing’an Cui. Threshold dynamics and ergodicity of an sirs epidemic model with Markovian switching. *Journal of Differential Equations*, 263(12):8873 – 8915, 2017.
- [90] Fengyan Li, Chi-Wang Shu, Yong-Tao Zhang, and Hongkai Zhao. A second order discontinuous Galerkin fast sweeping method for eikonal equations. *Journal of Computational Physics*, 227(17):8191–8208, 2008.
- [91] M. J. Lighthill and G. B. Whitham. On kinematic waves. II. a theory of traffic flow on long crowded roads. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 229(1178):317–345, 1955.
- [92] Jean-Marie Mirebeau. Efficient fast marching with Finsler metrics. *Numerische mathematik*, 126(3):515–557, 2014.

- [93] Ian M Mitchell and Shankar Sastry. Continuous path planning with multiple constraints. In *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, volume 5, pages 5502–5507. IEEE, 2003.
- [94] John Nash. Non-cooperative games. *Annals of mathematics*, pages 286–295, 1951.
- [95] FP Navin and RJ Wheeler. Pedestrian flow characteristics. *Traffic Engineering, Inst Traffic Engr*, 39, June 1969.
- [96] GJ Olsder and R Suri. Time-optimal control of parts-routing in a manufacturing system with failure-prone machines. In *Decision and Control including the Symposium on Adaptive Processes, 1980 19th IEEE Conference on*, pages 722–727. IEEE, 1980.
- [97] Martin J Osborne and Ariel Rubinstein. *A course in game theory*. MIT press, 1994.
- [98] Stanley Osher and James A Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79(1):12 – 49, 1988.
- [99] F. S. Priuli. First order mean field games in crowd dynamics. *ArXiv pre-print*, February 2014.
- [100] Jianliang Qian, Yong-Tao Zhang, and Hong-Kai Zhao. Fast sweeping methods for eikonal equations on triangular meshes. *SIAM Journal on Numerical Analysis*, 45(1):83–107, 2007.
- [101] TES Raghavan. Zero-sum two-person games. *Handbook of game theory with economic applications*, 2:735–768, 1994.
- [102] Elizabeth J. Z. Robinson. India’s disappearing common lands: Fuzzy boundaries, encroachment, and evolving property rights. *Land Economics*, 84(3):409–422, 2008.
- [103] Elizabeth J.Z. Robinson, Heidi J. Albers, and Jeffrey C. Williams. Spatial and temporal modeling of community non-timber forest extraction. *Journal of Environmental Economics and Management*, 56(3):234 – 245, 2008.
- [104] Ryszard Rudnicki and Marta Tyran-Kaminska. Piecewise deterministic Markov processes in biological models. In Jacek Banasiak, Adam Bobrowski, and Mirosław Lachowicz, editors, *Semigroups of Operators -Theory and Applications*, pages 235–255, Cham, 2015. Springer International Publishing.

- [105] Zeiser S., U. Franz, O. Wittich, and V. Liebscher. Simulation of genetic networks modelled by piecewise deterministic Markov processes. *IET Systems Biology*, 2(3):113–135, 2008.
- [106] S. Samaranayake, S. Blandin, and A. Bayen. A tractable class of algorithms for reliable routing in stochastic networks. *Transportation Research Part C*, 20(1):199–217, 2012.
- [107] Suresh P Sethi and Qing Zhang. *Hierarchical decision making in stochastic manufacturing systems*. Springer Science & Business Media, 2012.
- [108] J. A. Sethian and A. Vladimirsky. Fast methods for the Eikonal and related Hamilton–Jacobi equations on unstructured meshes. *Proceedings of the National Academy of Sciences*, 97(11):5699–5703, 2000.
- [109] James A Sethian. A fast marching level set method for monotonically advancing fronts. *Proceedings of the National Academy of Sciences*, 93(4):1591–1595, 1996.
- [110] James A Sethian. Fast marching methods. *SIAM review*, 41(2):199–235, 1999.
- [111] James A Sethian and Alexander Vladimirsky. Ordered upwind methods for static Hamilton–Jacobi equations: Theory and algorithms. *SIAM Journal on Numerical Analysis*, 41(1):325–363, 2003.
- [112] James Albert 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.
- [113] Zhengdi Shen and Alexander Vladimirsky. Piecewise-deterministic optimal path planning. *arXiv preprint arXiv:1512.08734*, 2015.
- [114] Chi-Wang Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In *Advanced numerical approximation of nonlinear hyperbolic equations*, pages 325–432. Springer, 1998.
- [115] Chi-Wang 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.
- [116] Xiong Tao, Zhang Peng, Wong S. C., Shu Chi-Wang, and Zhang Meng-Ping. A macroscopic approach to the lane formation phenomenon in pedestrian counterflow. *Chinese Physics Letters*, 28(10):108901, 2011.

- [117] SH Tijs. *Semi-infinite linear programs and semi-infinite matrix games*. Katholieke Universiteit Nijmegen. Mathematisch Instituut, 1976.
- [118] Y-HR Tsai, L-T Cheng, Stanley Osher, Paul Burchard, and Guillermo Sapiro. Visibility and its dynamics in a PDE based implicit framework. *Journal of Computational Physics*, 199(1):260–290, 2004.
- [119] Yen-Hsi Richard Tsai, Li-Tien Cheng, Stanley Osher, and Hong-Kai Zhao. Fast sweeping algorithms for a class of Hamilton–Jacobi equations. *SIAM journal on numerical analysis*, 41(2):673–694, 2003.
- [120] John N Tsitsiklis. Efficient algorithms for globally optimal trajectories. *IEEE Transactions on Automatic Control*, 40(9):1528–1538, 1995.
- [121] D Verms. Optimal control of piecewise deterministic Markov process. *Stochastics: An International Journal of Probability and Stochastic Processes*, 14(3):165–207, 1985.
- [122] Chang Wang, Fei Qi, Guangming Shi, and Xiaotian Wang. A sparse representation-based deployment method for optimizing the observation quality of camera networks. *Sensors*, 13(9):11453–11475, 2013.
- [123] S. C. Wong, W. L. Leung, S. H. Chan, William H. K. Lam, Nelson H. C. Yung, C. Y. Liu, and Peng Zhang. Bidirectional pedestrian stream model with oblique intersecting angle. *Journal of Transportation Engineering*, 136(3):234–242, 2010.
- [124] Tao Xiong, Mengping Zhang, Chi-Wang Shu, S.C. Wong, and Peng Zhang. High-order computational scheme for a dynamic continuum model for bi-directional pedestrian flows. *Computer-Aided Civil and Infrastructure Engineering*, 26(4):298–310, 2011.
- [125] Yong-Tao Zhang, Shanqin Chen, Fengyan Li, Hongkai Zhao, and Chi-Wang Shu. Uniformly accurate discontinuous Galerkin fast sweeping methods for eikonal equations. *SIAM Journal on Scientific Computing*, 33(4):1873–1896, 2011.
- [126] Yong-Tao Zhang, Hong-Kai Zhao, and Jianliang Qian. High order fast sweeping methods for static Hamilton–Jacobi equations. *Journal of Scientific Computing*, 29(1):25–56, 2006.
- [127] Hongkai Zhao. A fast sweeping method for eikonal equations. *Mathematics of computation*, 74(250):603–627, 2005.