On Maximum-reward Motion in Stochastic Environments

by

Fangchang Ma

Submitted to the Department of Aeronautics and Astronautics in partial fulfillment of the requirements for the degree of

Master of Science in Aeronautics and Astronautics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2015

© Massachusetts Institute of Technology 2015. All rights reserved.

Signature redacted

Author ...........................................................

Department of Aeronautics and Astronautics

May 21, 2015

Signature redacted

Certified by ....................................................

Sertac Karaman

Assistant Professor of Aeronautics and Astronautics

Thesis Supervisor

Signature redacted

Accepted by ....................................................

Paulo C. Lozano

Associate Professor of Aeronautics and Astronautics

Chair, Graduate Program Committee
On Maximum-reward Motion in Stochastic Environments

by

Fangchang Ma

Submitted to the Department of Aeronautics and Astronautics
on May 21, 2015, in partial fulfillment of the
requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

In this thesis, we consider the problem of an autonomous mobile robot operating in a stochastic reward field to maximize total rewards collected in an online setting. This is a generalization of the problem where an unmanned aerial vehicle (UAV) collects data from randomly deployed unattended ground sensors (UGS).

Specifically, the rewards are assumed to be generated by a Poisson point process. The robot has a limited perception range, and thus it discovers the reward field on the fly. The robot is assumed to be a dynamical system with substantial drift in one direction, e.g., a high-speed airplane, so it cannot traverse the entire field. The task of the robot is to maximize the total rewards collected during the course of the mission, given above constraints. Under such assumptions, we analyze the performance of a simple receding-horizon planning algorithm with respect to the perception range, robot agility and computational resources available.

Firstly, we show that, with highly limited perception range, the robot is able to collect as many rewards as if it could see the entire reward field, if and only if the reward distribution is light-tailed. The second result attained shows that the expected rewards collected scale proportionally to the square root of the robot agility. Finally, we are able to prove that the overall computational workload increases linearly with the mission length, i.e., the distance of travel. We verify our results in simulation examples.

At the end, we present one interesting application of our theoretical study to the ground sensor selection problem. For an inference/estimation task, we prove that sensors with randomized quality outperform those with homogeneous precisions, since random sensors yield a higher confidence level of estimation (lower variance), under certain technical assumptions. This finding might have practical implications on the design of UAV-UGS systems.

Thesis Supervisor: Sertac Karaman
Title: Assistant Professor of Aeronautics and Astronautics
Acknowledgments

First and foremost, I would like to express my sincere gratitude to my advisor, Professor Sertac Karaman, for his caring, patience and continuous support throughout my time at MIT. It is virtually impossible, without Prof. Karaman’s guidance and inspiration, to do my thesis work on such a brand new research topic. I am indebted to his encouragement to grow a higher level of mathematical maturity and rigor, which will certainly help my future research down the road. As one of the nicest and most friendly person I’ve ever met, Prof. Karaman is the best supervisor one could expect.

I would also like to acknowledge the friends I met at MIT. David Miculescu and Quanwen Zhu have been my labmates since day one, and we have had uncountably many stimulating and long-lasting discussions. My roommates, Qingkai Liang, Yili Qian, Rujian Chen, have also provided valuable input and suggestions which helped improve this dissertation.

Finally, I would like to express my heart-felt gratitude to my parents and Yumeng. They have been a source of patience and love for the past years. None of this would be possible without my parent’s understanding or Yumeng’s encouragements. Their support helped me overcome setbacks in difficult times.
Contents

1 Introduction .......................................................... 11
   1.1 Motivation .................................................. 11
   1.2 Related Work ............................................... 13
       1.2.1 Traveling Salesman Problem ...................... 13
       1.2.2 Vehicle Routing Problem ......................... 14
       1.2.3 Persistent Monitoring Problem .................. 14
       1.2.4 Navigation Through Random Obstacle Field .... 15
   1.3 Organization ............................................... 15

2 Problem Definition ................................................. 17
   2.1 Maximum-reward Motion in a Stochastic Environment .... 17
   2.2 Lipschitz-continuous Paths in a Poisson Reward Field ... 21

3 Preliminaries: Discrete Problem .................................. 25
   3.1 Lattice-based Discretization ................................ 26
   3.2 Motion Planning on Lattice .................................. 27
   3.3 On Infinite-horizon Mean Reward ......................... 28
   3.4 On Perception Range ....................................... 31
       3.4.1 For Light-tailed Rewards .......................... 31
       3.4.2 For Heavy-tailed Rewards .......................... 34

4 Motions in Continuous Spaces ................................... 37
   4.1 Motion Planning Algorithm .................................. 37
4.2 On Infinite-Horizon Mean Rewards ........................................... 38
4.3 Performance with respect to Perception Range ............................. 40
4.4 Performance with respect to the Robot’s Agility ............................. 41
4.5 Computational Workload .......................................................... 42

5 Computational Experiments ....................................................... 45
5.1 Conjecture 1 ........................................................................... 45
5.2 Mean Rewards with Discrete Lattices .......................................... 45
5.3 Perception Range with Discrete Lattices ....................................... 48
5.4 Mean Rewards in Continuous Spaces .......................................... 48
5.5 Perception Range in Continuous Spaces ....................................... 50
5.6 Robot Agility in Continuous Space .............................................. 51

6 Application: Sensor Selection ....................................................... 53

7 Conclusions and Remarks ............................................................ 57

A Proofs ......................................................................................... 61
A.1 Proof for Theorem 1 ................................................................. 61
A.2 Proof for Theorem 2 ................................................................. 61
A.3 Proof for Theorem 3 ................................................................. 63
A.4 Proof for Theorem 4 ................................................................. 64
A.5 Proof for Corollary 2 ............................................................... 66
A.6 Proof for Theorem 5 ................................................................. 66
A.7 Discussion for Conjecture 2 ...................................................... 69
A.8 Proof for Theorem 8 ................................................................. 72
List of Figures

2-1 An illustration of the vehicle navigating in a stochastic reward field. The blue cylinders represent the target locations. The yellow region represents the target-detection range of the vehicle. The locations of all targets in this range are known to the vehicle. By visiting these target locations, the vehicle can collect the reward assigned to the same targets, as illustrated by the trajectory of the vehicle, which is shown in red in the figure. .................................................. 18

3-1 The two-dimensional directed regular lattice, $\mathbb{N}^2$, is illustrated in Figure (a). An example state-lattice for a curvature-constrained Dubins vehicle, is shown in Figure (b). The latter lattice can be embedded in $\mathbb{N}^2$. ................................................................. 26

5-1 Each point on the surface shows the mean maximal rewards $T((x, y))/(x + y)$ for a end point $(x, y)$, when rewards are i.i.d. Poisson random variables with $p = 0.5$. This normalized shape function converges to above 1.1, which is the lower bound predicted by Theorem 1 contingent on Conjecture 1. .................................................. 46

5-2 Unit-distance mean rewards $R(n)/n$ versus mission length $n$ on a two-dimensional regular lattice. The mean rewards collected by the robot, shown by the blue curve, quickly approaches to a value above the lower bound $\mu + \sigma$ indicated by the red line. .................................................. 47
5-3  Expected reward collected versus perception range with heavy-tailed distributions on a two-dimensional regular lattice. Rewards follow a Pareto distribution with $\alpha = 1.5$. The blue curve shows the mean maximal rewards $R(n)/n$ collected with a mission length $n$, and the red line is proportional to $x^{2/\alpha-1}$. 

5-4  Expected distance-of-travel versus perception range on a two-dimensional regular lattice for geometric-distributed rewards. The blue curve shows the mean distance-of-travel when the tolerance on loss of unit-distance rewards is $\delta = 0.08$. The red line is a linear fit of the blue dots for large perception range $m$. The $y$-axis is on log-scale.

5-5  Expected reward versus mission length in continuous space $\mathbb{R}^2$. The red dots are the unit-distance rewards obtained from the maximum-reward motion versus distance-of-travel, while the blue dots are those from the strategy that maximizes visited targets. The horizontal lines are the theoretical lower bound and the limit, respectively.

5-6  The average distance-of-travel is plotted against perception range $m$ of the robot. The Poisson process is parameterized with $\lambda = 1$ and each rewards $r_i$ follows an exponential distribution with parameter 1. Note that the $y$-axis is on log scale, so the distance of travel increases exponentially fast with perception range $m$, when $m$ is sufficiently large (even for such unbounded light-tailed rewards).

5-7  Expected reward versus robot agility for the simulation where the intensity $\lambda$ of the Poisson point process is 10 and the rewards follow an exponential distribution with mean 1.

5-8  Expected reward versus both the perception range and the agility for the simulation where the intensity $\lambda$ of the Poisson point process is 1 and the rewards follow an exponential distribution with mean 1.

A-1  The reachable set of the robot with drift, whose dynamics is described mathematically by Equations (2.1).
Chapter 1

Introduction

1.1 Motivation

With advancements in sensor, computation, and communication technologies, micro unmanned aerial vehicles (UAV) are gaining increasing popularity in civilian applications. The same technological advancements also bring unattended ground sensors (UGS) to affordable prices. These ground sensors come in great diversity and employ one or more sensing phenomenologies including but not limited to seismic, acoustic, magnetic, imaging infrared. They are disposable, easily deployable, and provide long-term measurements without requiring human interventions. UAVs, together with randomly deployed stationary UGS, can significantly improve the performance of autonomous surveillance tasks. This integrated UAV-UGS system enables new application areas such as in-depth surveillance of volcanic activities, forest fires, oceanic currents, wildlife detection, monitoring of gas pipelines [13], and pollution control. It also provides an efficient and easy access to large-scale sensing over a long period.

More specifically, consider the application of the UAV-UGS system in environmental monitoring. UAVs and UGS (or more generally, mobile robotic vehicles and stationary sensing devices) work together to collect valuable information about the state of the environment. The small sensing devices house primitive sensors, depending on the actual applications. Along with sensing, these devices include communication equipment and (primitive) computational platforms. Suppose these sensing
devices are deployed throughout the environment for persistent monitoring purposes. Rather than attempting to form an ad-hoc network, a mobile data-harvesting vehicle traverses the environment, discovers the sensing devices on the fly, and approaches them to harvest their data only if they have valuable information. The robot may not know the precise positions of the sensing devices a priori. Instead, sensors are discovered on the fly, and the robot makes small corrections in its trajectory to collect as much information as possible. Clearly, the amount of information that can possibly be collected depends on a number of factors, including the density of sensors, the quality of measurements, the duration of mission, the agility of the robot (its actuation capabilities) as well as its perception range (how soon it can discover the sensors).

Let us note that similar problems arise in a large class of applications. Examples include mobile robotic vehicles dropping cargo to target locations or taking pictures of them, with the targets discovered on the fly. To generalize, we consider a robot that is traversing a stochastic “reward field”, where the precise value of the reward is discovered on the fly, with only the statistics of the rewards given. In this settings, we aim to find an optimal control policy such that the total amount of rewards collected is maximized. We will refer the planning and control problem of this vehicle as Maximum-reward Motion.

In this thesis, we consider the following fundamental problems regarding Maximum-reward Motion: How quickly can the mobile robots collect rewards from the field, given their perception, actuation, and computation capabilities? What are the planning algorithms that achieve optimal performance? We would also like to address some of the practical issues in the design of such robot-sensor systems. For example, with a fixed budget, how do we balance between the quantity and quality of the ground sensors deployed? All these questions will be discussed in detail in the following chapters.

\footnote{Ad-hoc sensor networks may also be very valuable for environmental monitoring. In fact, such technologies have been developed over past several years. We note that the presented application is for motivational purposes only. Yet, it may be beneficial over the ad-hoc network approach due to substantial energy savings at the stationary sensors (as communication requirement is much lower); hence, the sensor nodes require less maintenance. The main drawbacks are the following. First, this approach adds the additional complexity of a mobile vehicle. Second, the data is received with some delay due to the mobile vehicles physically carrying the data.}
1.2 Related Work

There are research problems that are naturally related to the maximum-reward motions but differ in some important aspects. These problems include the traveling salesman problem, vehicle routing problem, and persistent monitoring. These problems differ from the maximum-reward motion in that the state space is usually bounded and, in some cases, the robot has to visit every target. In comparison, in the maximum-reward motion, we focus on dynamical systems with drift. In other words, the robot keeps exploring new region and has to skip some rewards.

1.2.1 Traveling Salesman Problem

The traveling salesman problem (TSP), also known as the traveling salesperson problem, is an important problem in theoretical computer science, operational research and mathematics. Given a list of $n$ cities and the distances between every pair of them, it requires to find the shortest tour that visits each of the cities once and returns to the starting city. The problem has been shown to be NP-hard. Heuristics [20] and polynomial-time constant factor approximate algorithms [27] exist, and branch and bound strategy and linear programming solvers have also been introduced to solve large-scale TSP problems [24]. In the early work, Beardwood et al. consider the stochastic TSP problem where the $n$ points are independently and uniformly distributed over a bounded region of area $v$. They proved that the length of the shortest closed path through such $n$ points is "almost always" asymptotically proportional to $\sqrt{n}v$ for large $n$ [1]. In the more recent work, Savla and Frazzoli consider the traveling salesperson problem for the Dubins vehicle (DTSP), i.e., a non-holonomic vehicle that is constrained to move along planar paths of bounded curvature, without reversing direction. They showed that for the Dubins vehicle, the expected length of the shortest tour through $n$ random targets is of order at least $n^{2/3}$ [28].
1.2.2 Vehicle Routing Problem

The vehicle routing problem (VRP), first introduced by Dantzig and Ramser [8] in 1959, is a generalization of the TSP problem. In this problem, a fleet of vehicles, which start from a central depot, seek to service a number of customers. It is NP-complete to find the optimal solution. A number of variants have been studied, where the information available to the planner may change and these information might contain uncertainties [25]. A version of the VRP problem for single vehicle is due to Wilson and Colvin [37], where customer requests appear dynamically. Dynamic and stochastic routing problems are also studied as an extension of their deterministic counterparts. For example, the dynamic traveling repairman problem (DTRP) was proposed by Bertsimas and Ryzin [2]. In this problem, demands for service arrive according to a Poisson process temporally, and are independently and uniformly distributed over a bounded region. The goal is to find a policy for routing the service vehicle such that the average time demands in the system is minimized. Lower bounds on the expected system time is provided.

1.2.3 Persistent Monitoring Problem

The more recent persistent monitoring problem seeks to generate an optimal control strategy for a team of agents to monitor a dynamically changing environment. For example, a persistent monitoring task with 1-D mission space of fixed length $L$ is considered in [4]. The objective is to minimize an uncertainty metric in a given mission space. It is shown that this monitoring problem can be reduced to a parametric optimization problem in order to find a complete optimal solution. The same approach is extended to a two-dimensional mission space in [5]. A different formulation of the persistent monitoring problem is considered in [32, 33]. Smith et al. models the changing environment with an accumulating function that grows in areas outside the range of the robot and decreases otherwise. In a more recent work [38], Yu et al. focus on a stochastic model of occurrence of events, where the precise occurrence time is not known a priori.
1.2.4 Navigation Through Random Obstacle Field

A dual problem of the maximum-reward motion has been studied. In [16] Karaman et al. investigate high-speed navigation through a randomly-generated obstacle field, where only the statistics of the obstacles are given. They show that with a simple dynamical model of the bird, the existence of an infinite collision-free trajectory through the forest exhibits a phase transition, i.e., there is an infinite obstacle-free trajectory almost surely when the speed is below a threshold and it will collide with some tree eventually otherwise. In [17], they further show that a planning algorithm based on state lattices can navigate the robot with limited sensing range.

1.3 Organization

The remainder of this thesis is organized as follows. We begin in Chapter 2 with the problem definitions of the general maximum-reward motion and its special case in two dimensional space. Chapter 3 is devoted to a preliminary analysis of the discretized maximum-reward motion, where we study the impact of perception ranges on robot performance in two different situations, i.e., when the rewards are light-tailed distributed and when they are heavy-tailed distributed. Following this, we show in Chapter 4 that the same results in Chapter 3 the can be extended to the continuous problem by taking the limit of discretization. We examine the impact of robot agility on rewards collected. We support our theoretical results via simulations in Chapter 5. Finally in Chapter 6, we present one application of the maximum-reward motion as sensor selection problem.
Chapter 2

Problem Definition

This chapter is devoted to a formal definition of the problem. First, we define the notion of maximum-reward motion through a stochastic reward field in its most general form. Second, we introduce an important special case, which this thesis focuses on. Finally, we outline an application that relates the maximum-reward motion problem to a certain sensor selection problem for an inference problem involving mobile robotic vehicles tasked with harvesting data from stationary sensors.

2.1 Maximum-reward Motion in a Stochastic Environment

Consider a robotic vehicle navigating in a stochastic environment, where the locations of targets are distributed randomly and each target location is associated with a random reward value. The precise locations of all of the targets are unknown to the robot a priori. Instead, the vehicle discovers the target locations and the rewards associated with the targets on the fly, when the targets become visible to the robot’s target-detection sensor. Once a target is detected, the vehicle can visit the target location and collect the reward associated with that target. When the vehicle is subject to differential constraints involving substantial amount of drift, the vehicle must visit the most valuable targets that are in the direction of drift in order to
maximize the total reward it collects, often at the expense of skipping some of the target locations, for instance, those that are orthogonal to the drift direction. See Figure 2-1.

Figure 2-1: An illustration of the vehicle navigating in a stochastic reward field. The blue cylinders represent the target locations. The yellow region represents the target-detection range of the vehicle. The locations of all targets in this range are known to the vehicle. By visiting these target locations, the vehicle can collect the reward assigned to the same targets, as illustrated by the trajectory of the vehicle, which is shown in red in the figure.

In this scenario, we are interested in understanding the fundamental limits of the performance of the vehicle with respect to its perception abilities (e.g., the range of its target-detection sensor) and its differential constraints (e.g., its agility).

In this section, we present the reward collection problem in a general form. In the next section, we introduce a special case that captures all aspects of the problem. This special case is also analytically tractable. In particular, we can derive the aforementioned fundamental limits for this special case.

The online motion planning problem is formalized as follows in its most general form:

**Dynamics:** Consider a mobile robotic vehicle that is governed by the following
equations:

\[
\dot{x}(t) = f(x(t), u(t)),
\]
\[
y(t) = g(x(t))
\]

where \( x(t) \in X \subseteq \mathbb{R}^n \) represents the state, \( u(t) \in U \subseteq \mathbb{R}^m \) represents the control input, \( y(t) \in \mathbb{R}^2 \) is the position of the robot on the plane where the targets lie, \( X \) is called the state space, and \( U \) is called the control space. A state trajectory \( x : [0, T] \to X \) is said to be a dynamically-feasible state trajectory and \( y : [0, T] \to \mathbb{R}^2 \) is said to be a dynamically-feasible output trajectory, if there exists \( u : [0, T] \to U \) such that \( u, y, \) and \( x \) satisfy Equation (2.1) for all \( t \in [0, T] \).

We are particularly interested in the case when the robot is subject to drift, for instance, when the robot can not come to a full stop immediately or can not even substantially slow down.\(^1\) Examples include the models of fixed-wing airplanes, racing cars, large submarines, and speed boats. In the next section, we present a model, which we believe is the simplest model that captures the drift phenomenon.

**Targets and rewards:** The target locations and the rewards associated with the targets are assumed to be generated by a stochastic marked point process.\(^2\) A marked point process is defined as a random countably-infinite set of pairs \( \{(p_i, m_i) : i \in \mathbb{N}\} \), where \( p_i \in \mathbb{R}^2 \) is the location of point \( i \) in the infinite plane and \( m_i \in M \) is the mark associated with point \( i \). We denote this random set by \( \Psi \). With a slight abuse of notation, we denote the number of points in a subset \( A \subseteq \mathbb{R}^2 \) of the infinite plane by \( \Psi(A) \). Given a point \( p \) of the point process, we denote its mark by \( R(p) \). In our case, the locations of the points will represent the locations of the targets, and the marks will represent the rewards associated with the locations. Hence, the mark set is the set of all non-negative real numbers, \( i.e., M = \mathbb{R}_{\geq 0} \). Following Stoyan et al. [6], we

---

\(^1\)Let us note at this point that “dynamical systems with drift” can be defined precisely, for instance, through differential geometry [14]. However, we will not need such differential-geometric definitions in this thesis, since we focus on a particular system with drift (introduced in the next section) and leave the generations other systems with drift to future work.

\(^2\)Strictly speaking, stochastic point processes are formalized using counting measures [7]. For the sake of the simplicity of the presentation, we will avoid these measure-theoretic constructs, and instead we will use the simpler notation adopted by Stoyan et al. [6].
also make the following technical assumptions: (i) any bounded subset of the infinite plane contains finitely many points, i.e., $|\Psi(A)| < \infty$ for all bounded measurable $A \subset \mathbb{R}^2$; (ii) no two points are located at the same location, i.e., $p_i \neq p_j$ for all $i \neq j$.

**Sensor footprint:** The locations of the targets and the rewards associated with them is not known a priori, but is revealed to the robot in an online manner. This aspect of the problem is formalized as follows. Let $\mathcal{P}_\gamma(\cdot)$ denote the perception footprint of the robot that associates each state $z \in X$ of the robot with a footprint $\mathcal{P}_\gamma(z) \subset \mathbb{R}^2$. When the robot is in state $z \in X$, it is able to observe only those targets that lie in the set $\mathcal{P}_\gamma(z)$. That is, $\{(p_i, m_i) \in \Psi : p_i \in \mathcal{P}_\gamma(z)\}$ is the set that is known to the robot when it is in state $z$.

**Task:** Roughly speaking, the robot is assigned with the task of collecting the maximal reward per unit time, subject to all the above constraints. We formalize this problem as follows. Suppose the stochastic marked point process that represents the targets is defined on the following probability space: $(\Omega, \mathcal{F}, \mathbb{P})$, where $\Omega$ is the sample space, $\mathcal{F}$ is the $\sigma$-algebra, and $\mathbb{P}$ is the probability measure. Define $\mathcal{F}_t$ as the $\sigma$-algebra generated by the random variables $\bigcup_{\tau \in [0,t]} \mathcal{P}_\gamma(x(\tau))$. A feasible control policy is a stochastic process $\mu = \{u(t) : t \in [0,T]\}$, such that $u(t)$ is defined on $(\Omega, \mathcal{F}_t, \mathbb{P})$, for all $t \geq 0$. This definition implies that the control policy depends only on the locations and the rewards of the targets that were seen by the robot. Note that the control policy may depend on the statistics of the stochastic marked point process that represents the targets, if such statistics are known a priori.

Given a control policy $\mu = \{u(t) : t \in [0,T]\}$, let us denote the resulting state trajectories by $\{x_\mu(t) : t \in [0,T]\}$ and the resulting output trajectories by $\{y_\mu(t) : t \in [0,T]\}$, which are stochastic processes on their own right, defined on the same probability space where the control policies were defined.

Let $Y(\mu; \psi)$ denote the set of targets visited by the robot under the influence of the control policy $\mu$ and when the realization of the targets and rewards process $\Psi$ is

---

3 We omit some of the measure-theoretic details when defining the control policy. Our definition matches the definition of an adapted control policy given in the book by Kushner [19], to which the interested reader is referred.
ψ, i.e.,

\[ Y(\mu; \psi) = \{ p \in \psi : y_\mu(t) = p \text{ for some } t \in [0, T] \}. \]

With a slight abuse of notation, let \( \mathcal{R}(\mu; \psi) \) denote the total amount of reward collected by the robot when it visits these targets, i.e.,

\[ \mathcal{R}(\mu; \psi) = \sum_{p \in Y(\mu; \psi)} \mathcal{R}(p). \]

Then, the maximum-reward motion problem is to find a control policy \( \mu \) such that the total reward \( \mathcal{R}(\mu; \psi) \) is maximized for all realizations of \( \psi \) of the point process \( \Psi \). We stress that an algorithmic solution to this problem, i.e., computing such a policy, is often simple, particularly when the point process \( \Psi \) is completely random (e.g., a Poisson process). In this thesis, we are interested in probabilistic analysis of the maximum reward that can be achieved by an optimal algorithm. Such analysis may allow robotics engineers to design robotic systems, e.g., by choosing the perception, actuation, and computation capabilities of the robots, so that the robots best fit the application at hand, rather than designing algorithms.

### 2.2 Lipschitz-continuous Paths in a Poisson Reward Field

In this section, we present a two dimensional special case. We believe this is the simplest case that captures all aspects of the problem we presented above, namely the drift in the dynamics, stochastic nature of the reward, and the online nature of the problem. This particular case is particularly relevant to the motivational example presented in Chapter 1. Furthermore, this case is also analytically tractable. In particular, we can derive the aforementioned fundamental limits for this special case utilizing some well-known results from the literature on non-equilibrium statistical mechanics.
**Dynamics:** Let $X = \mathbb{R}^2$, and define the dynamics governing the robot with the following ordinary differential equation:

\[
\begin{align*}
\dot{x}_1(t) &= v, \\
\dot{x}_2(t) &= u(t),
\end{align*}
\]  

where $[x_1(t) \ x_2(t)] \in \mathbb{R}^2$ denotes the state of the robot, $v$ is a constant, and $|u(t)| \leq w$ is the control input. This robot travels with constant speed along the longitudinal direction ($x$-axis), and it has bounded speed in the lateral direction ($y$-axis). We define the **agility** of the robot is defined as $w/v$. The larger this number, the more maneuverable is the robot.

**Targets and rewards:** The target locations are generated by a two dimensional Poisson point process with intensity $\lambda$ and the rewards are chosen from the same distribution independently. In other words, the number of targets $\Psi(A)$ for any region $A \in \mathbb{R}^2$ follows a Poisson distribution, i.e.,

\[\Psi(A) \sim \text{Poi}(\lambda |A|).\]

Let $r(p)$ denote the reward associated with the target at location $p \in \mathbb{R}^2$. Then, \(\{r(p_i), i \in \mathbb{N}\}\) are independent identically distributed random variables.

**Sensor footprint:** The robot has a fixed perception range $m$. This implies that when the robot is at state $x(t) = [x_1(t) \ x_2(t)]$, it obtains the target information, namely its location and the associated reward, for all targets located in

\[\mathcal{P}_{\Psi}(x(t)) = \{p \in \mathbb{R}^2 : |p - x(t)| \leq m \text{ and } (p,m) \in \Psi\}.\]

In the rest of this thesis, we focus on the special case in a two dimensional space. Notice that this simple system captures all of the following aspects of the problem:
(i) robot dynamics with drift, (ii) stochastic nature of the reward, and (iii) online nature of the problem. In the rest of the thesis, we analyze the performance of the robot with respect to its actuation capabilities (e.g., agility), perception capabilities
(e.g., perception range), and computation capabilities.
Chapter 3

Preliminaries: Discrete Problem

It is hard to approach the continuous problem defined in Section 2.2 directly. In this chapter, we propose a simpler, approximate version of the original problem based on proper discretization of the set of all dynamically-feasible trajectories. With sufficiently high resolutions, the discrete problem produces a control policy that is arbitrary close to the one for the continuous problem, which is simply the limit of the discrete case.

The approximate problem is constructed and solved by using lattice-based discretization and planning. Lattice-based motion planning algorithms have long been widely adopted in robotics applications [10, 18]. These algorithms form a directed lattice in the state space of the robot and select the optimal path through this lattice. This task is often computationally efficient, making it a practical approach even for challenging problem instances.

Moreover, we analyze this discrete problem and the planning algorithm by establishing strong connections between this class of problems and nonequilibrium statistical mechanics. Roughly speaking, we view the robot as a particle traveling in a stochastic field. This perspective allows us to directly apply some of the recent results from the Last-Passage Percolation Problem [26, 39, 11, 22] to characterize various properties of our robotics.

In the rest of this chapter, we formally describe the discretization problem in Section 3.1. A lattice-based planning algorithm is introduced in Section 3.2. In
Section 3.3, we analyze the fundamental limit of the problem and in Section 3.4, the performance of receding-horizon planning algorithms are discussed in detail.

### 3.1 Lattice-based Discretization

In this section, we mathematically formulate the discrete problem and introduce some notations.

**Dynamics:** We form a 2-dimensional directed regular lattice \( L_2 = (V, E) \) in the continuous state space of the robot. \( V = \mathbb{N}^2 \) is a countable set of vertices, where each vertex is a state of the dynamical system described by Equation (2.1) and the distance between neighboring vertices is a constant. \( E \) is the set of dynamically-feasible trajectories between adjacent vertices, and \((v, v') \in E\) if \( v = (x_1, x_2), v' = (x'_1, x'_2) \) and there exists a dynamically-feasible state trajectory \( x_e : [0, T_e] \rightarrow X \) such that \( x(0) = v_1 \) and \( x(T_e) = v_2 \).

This two-dimensional directed lattice is illustrated in Figure 3-1a. A robot starts from the origin and can choose to either go up or go right in Figure 3-1a. An example state-lattice for the non-holonomic vehicle (such as a Dubins vehicle) that we described in Section (2.2) is shown in Figure 3-1b.

![Figure 3-1: The two-dimensional directed regular lattice, \( \mathbb{N}^2 \), is illustrated in Figure (a). An example state-lattice for a curvature-constrained Dubins vehicle, is shown in Figure (b). The latter lattice can be embedded in \( \mathbb{N}^2 \).](image)

More generally, this definition can be extended to higher dimensional space. Let \( L_d = (V, E) \) be a \( d \)-dimensional directed regular lattice. Let us denote the set of all paths in \( L_d = (V, E) \) by \( \text{Paths}(L_d) \). Given a path \( \pi \in \text{Paths}(L_d) \), let \(|\pi|\) denote the length of \( \pi \) measured by the number of vertices that \( \pi \) visits. Furthermore, \( \Pi(v_{\text{init}}, n) \)
denotes the set of all paths that start from the vertex $v_{\text{init}}$ and cross at most $n$ vertices. Similarly, $\Pi(v_{\text{init}}, v_{\text{dest}})$ denotes the set of all paths that start from the vertex $v_{\text{init}}$ and ends at $v_{\text{dest}}$.

**Targets and rewards:** Each vertex $v \in V$ is associated with an independent and identically distributed random reward $\rho(v)$.

**Sensor footprint:** The perception range is a positive integer $m$, such that any vertex reachable with a path of length $m$ is within the perception range. This perception range limitation allows the robot to observe only the rewards associated with a subset of the vertices.

### 3.2 Motion Planning on Lattice

Instead of doing motion planning in the continuous space, in this discrete problem we only consider motions on the discrete lattice, which significantly reduces the number of candidate paths and is computationally tractable.

If the robot’s perception range is limited, then a receding-horizon planning scheme can be applied. Specifically, suppose the robot starts at an initial state $z_{\text{init}} \in V$. In each iteration, the best path $(e_1, e_2, \ldots, e_k)$ within the “visible” region of the lattice is computed, and the robot executes this dynamically-feasible trajectory. Once the robot reaches the last state $v' = x_{e_k}(T_{e_k})$, the same procedure is repeated with an updated vision region. We will call this algorithm the **lattice-based receding-horizon motion planning algorithm**, which is formalized in Algorithm 1. CurrentState() is a procedure that returns the current state of the robot and Execute($x$) denotes the command that makes the robot follow a trajectory $x$. The algorithm first retrieves the robot’s current state (Line 2). Subsequently, it computes the visible region $G_P$ of the lattice (Line 3). It then searches for the optimal path over this region (Line 4) and finally executes this optimal path (Line 5). This procedure continues for $N$ iterations (Lines 1-5).

In Line 4, the algorithm computes the maximum-weight path on a finite weighted graph. Let us note that this problem is NP-hard in general [29]. However, the problem
Algorithm 1 Lattice-based receding-horizon online motion planning

1: for $t = 1, \ldots, N$ do  
2: $z \leftarrow \text{CurrentState}()$  
3: $G_P \leftarrow \text{Perceive}(P(z))$  
4: $\pi \leftarrow \arg \max \{ R(\text{Trajectory}(\pi) : \pi \in \text{Paths}(G_P)) \}$  
5: Execute($\pi$)

3.3 On Infinite-horizon Mean Reward

Before presenting the main theoretical result of this section, let us provide some notation. Let $T(v_{init}, v)$ denote the reward of the maximum reward path that starts from the origin and reaches the vertex $v$, i.e.,

$$T(v_{init}, v) := \max_{\pi \in \Pi(v_{init}, v)} \sum_{v' \in \pi} \rho(v').$$

and $R(v_{init}, n)$ denotes the maximal reward collected by following any path that starts from $v_{init}$ and crosses at most $n$ vertices, i.e.,

$$R(v_{init}, n) := \max_{\pi \in \Pi(v_{init}, n)} \sum_{v \in \pi} \rho(v).$$

Note that both $T(v_{init}, v)$ and $R(v_{init}, n)$ are the maximum rewards that the robot can collect given unlimited perception range. However, $T(v_{init}, v)$ differs from $R(v_{init}, n)$ in the sense that $T(v_{init}, v)$ is the mean maximal reward at a particular node $v$, while

---

1Acyclic graphs arise in lattice-based motion planning, for instance, when the robot does not return to previously visited locations, i.e., when the robot constantly explores new regions in the environment. When lattice-based motion planning algorithms are applied to robots subject to substantial drift, the resulting lattice is also often acyclic.
$R(v_{\text{init}}, n)$ is the mean maximal reward that the robot can collect in $n$ steps regardless the actual end point. In the sequel, when $v_{\text{init}}$ is the origin, we drop $v_{\text{init}}$ from our notation and write $T(v)$ and $R(n)$, respectively.

The asymptotic properties of the of the function $T(v)$ has been extensively analyzed by mathematical physicists in the non-equilibrium statistical mechanics literature. Let us recall some of the main results from this literature and subsequently state and prove our main results for this section.

**Definition 1** (Shape function). The function $g(\cdot)$, defined as

$$g(v) := \sup_{k \in \mathbb{N}} \frac{\mathbb{E}[T(kv)]}{k}$$

is called the shape function.

Firstly, the function $T(v)$ has been shown to be converge to a limit.

**Proposition 1** (See Proposition 2.1 in [22]). Assume the rewards $\rho(v)$ at each vertex $v$ are i.i.d. random variables and $\mathbb{E}[\rho(v)] < \infty$. Then, $\frac{T(nv)}{n}$ converges to the shape function $g(v)$ almost surely as $n$ diverges to infinity, i.e.,

$$\mathbb{P}\left( \lim_{n \to \infty} \frac{T(nv)}{n} = g(v) \right) = 1.$$

In other words, for any vertex $v$, the function $T(v)$ is finite and $T(v)$ converges almost surely to some finite value $g(v)$.

In addition, the results in [22] imply that $T(v)$ can be computed exactly for at least two cases, namely, when $F$ (the distribution of the rewards) follows either an exponential distribution or a geometric distribution. More specifically, if $F$ is an exponential distribution with parameter $\lambda = 1$, then the shape function $g$ defined in Proposition 1 is

$$g((x, y)) = (\sqrt{x} + \sqrt{y})^2, \quad \text{for all } (x, y) \in \mathbb{N}^2. \quad (3.1)$$

If $F$ is a geometric distribution with parameter $p$, i.e., $\mathbb{P}(X = k) = p(1 - p)^{k-1}$ for
\[ g((x, y)) = \frac{x + 2\sqrt{xy(1 - p)}}{p} + y, \quad \text{for all } (x, y) \in \mathbb{N}^2. \]

In [30], it is conjectured that these results might generalize; specifically, the shape function \( g \) for general distributions \( F \) with mean \( \mu \) and variance \( \sigma^2 \) on \( L_2 \) might be computed as follows.

**Conjecture 1 (See [30]).** For any distribution \( F \) with mean \( \mu \) and variance \( \sigma^2 \) on \( L_2 \), the shape function \( g(\cdot) \) is

\[ g((x, y)) = \mu(x + y) + 2\sqrt{\sigma^2 xy}, \quad \text{for all } (x, y) \in \mathbb{N}^2. \tag{3.2} \]

At this stage, a rigorous proof is not known to the mathematical physics researchers. A rigorous proof of this conjecture is also beyond the scope of this thesis. However, if this conjecture holds, it has an important implication for the problem described in this thesis: the expected maximal rewards collected, for arbitrary destination that is sufficiently distant, can be computed analytically. Results from computational simulations that support this conjecture are shown in Chapter 5.

The first main result of this section is that the unit-step mean maximal reward, regardless of the actual destination, is also well defined and converges to some fixed value.

**Proposition 2.** The following holds:

\[ \lim_{n \to \infty} \frac{\mathbb{E}[R(n)]}{n} = \sup_{n \in \mathbb{N}} \frac{\mathbb{E}[R(n)]}{n}. \]

**Proof.** The result follows directly from Fekete’s lemma [34], noting that the sequence \( \mathbb{E}[R(n)] \) is superadditive, and hence \(-\mathbb{E}[R(n)]/n\) is subadditive. \( \square \)

We call this value the *unit-step mean maximal reward on the d-dimensional di-*
rected regular lattice and denote it by \( R^*_d \), i.e.,

\[
R^*_d := \lim_{n \to \infty} \frac{E[R(n)]}{n} = \sup_{n \in \mathbb{N}} \frac{E[R(n)]}{n}.
\]

The second equation follows from Proposition 2. Note that \( R^*_d \) might be infinity, depending on the distribution \( F \) of rewards. Although \( R^*_d \) cannot be computed analytically, the second main result of this section is that, if Conjecture 1 holds, then \( R^*_d \) is bounded below by some constant and the lower bound depends only on the distribution \( F \).

**Theorem 1.** If Conjecture 1 holds, then for any distribution \( F \) with mean \( \mu \) and variance \( \sigma^2 \) on a two-dimensional lattice \( L_2 \), the unit-step mean maximal reward \( R^*_2 \) satisfies

\[
R^*_2 \geq \mu + \sigma.
\]

The proof is given in Appendix A.1. Theorem 1 implies that the mean reward collected by the robot converges to some constant bounded below by \( \mu + \sigma \), as the distance of travel increases to infinity. We show experimentally in Chapter 5 that this limit is actually very close to \( \mu + \sigma \). Note that this conclusion always holds for exponential and geometric-distributed rewards, since Conjecture 1 holds for these two distributions.

### 3.4 On Perception Range

#### 3.4.1 For Light-tailed Rewards

Let \( R_1 \) denote the reward that can be collected by a path that starts from \( v_{\text{init}} \) and has length \( m \), i.e., \( R_1 := R(v_{\text{init}}, m) \). Let \( v_1 \) denote the vertex where the maximum-reward path (achieving reward \( R_1 \)) ends. Similarly, define \( R_k := R(v_{k-1}, m) \), and let \( v_k \) be the vertex where the path achieving reward \( R_k \) ends. Finally, assume \( n \) is a
multiple of $m$ and define

$$Q(n; m) := \sum_{i=1}^{n/m} R_i.$$  

We compare the unit-step rewards $Q(n; m)/n$ and $R^*$. Recall that the former is the unit-distance reward that the robot can collect with limited perception range $m$, and the latter is the reward collected with infinite perception range. We discuss two different situations, i.e., when the rewards follow light-tailed distributions and when they follow heavy-tailed distributions. Roughly speaking, a light-tailed distribution is a distribution whose tail is bounded by an exponentially decreasing function. This definition can be formalized as follows.

**Definition 2.** The distribution of a multivariate random variable $X = [X_1, \ldots, X_d]$ with distribution function $F$ is said to be light tailed if

$$\int_0^\infty (1 - F(x))^{1/d} dx < \infty.$$  

A heavy-tailed distribution is one that is not light-tailed.

Theorem 2 shows that when the rewards are bounded almost surely, receding-horizon algorithms with limited perception range achieve near-optimal performance.

**Theorem 2.** Suppose $R^*_d$ is finite, and the rewards $\rho(v)$ are independent and identically distributed on $L_d$. Suppose that these rewards are uniformly almost-surely bounded random variables, i.e., there exists some $L$ such that $\mathbb{P}(|\rho(v)| \leq L) = 1$ for all $v \in V$. Then, for any $\delta > 0$, there exists a constant $c$ depending on $\delta$ such that

$$\lim_{n \to \infty} \mathbb{P}\left( \left| \frac{Q(n,c \log n)}{n} - R^*_d \right| \geq \delta \right) = 0.$$  

The proof of Theorem 2 is provided in Appendix A.2. Roughly speaking, Theorem 2 implies that the robot can navigate to any vertex that is at most $n$ steps away almost optimally (as if it had infinite perception range), if its perception range is at order $\log n$. In other words, as the perception range increases, the distance that the robot can travel optimally increases exponentially fast, as stated below.
Corollary 1. Suppose the assumptions of Theorem 2 hold. Then, for any $\delta > 0$, there exists some constant $c$ such that

$$\lim_{m \to \infty} \mathbb{P}\left( \left| \frac{Q(L(m),m)}{L(m)} - R_d^* \right| \geq \delta \right) = 0,$$

where $L(m) = e^{cm}$ for some constant $c$ that is independent of $m$ (but depends on $\delta$).

This corollary follows from Theorem 2 with a change of variables.

Note that Theorem 2 and Corollary 1 are statements on regular lattice $L_d$ of any dimension $d$. Furthermore, we conjecture that both statements hold true not only for bounded distributions $F$, but also for any light-tailed distributions. Although we do not have a rigorous proof for this conjecture at the moment, our results from the simulation experiments provided in Chapter 5 support this conjecture.

Notice that the constant $c$ in the statement of Corollary 1 is not specified. We know that $c$ is independent of $n$ but depends on $\delta$, the tolerance on loss of performance. Our next result characterizes how this constant $c$ depends on $\delta$ on a two-dimensional regular lattice, by utilizing relatively recent results from the nonequilibrium statistical mechanics literature. This is achieved by utilizing more accurate characterizations of the function $T(\cdot)$ defined in Section 3.3. It is shown in [15] that, for the aforementioned exponential and geometric distributions, the following holds:

$$T(\left| \{x_n, y_n\} \right| - n g((x,y)) \to F_2,$$

as $n$ goes to infinity, where $F_2$ is the Tracy-Widom distribution [28]. Under the same conditions, we find that $c = \kappa \delta^{3/2}$ for some constant $\kappa$ that is independent of $\delta$ and $n$.

Theorem 3. Suppose the lattice $G = (V, E)$ is embedded in $\mathbb{N}^2$ and $\rho(v)$ are independent and identically distributed random variables. Suppose their common distribution $F$ is either an exponential distribution or a geometric distribution. Then,

$$\lim_{m \to \infty} \mathbb{P}\left( \left| \frac{Q(L(m),m)}{L(m)} - R_2^* \right| \geq \delta \right) = 0,$$
where \( L(m) = \exp(\kappa \delta^{3/2} m) \), for some constant \( \kappa > 0 \) independent of \( m \) and \( \delta \).

The proof for Theorem 3 is in Appendix A.3.

Let us compare it with Corollary 1. While Corollary 1 characterizes the distance of travel \( L(m) \) with respect to perception range \( m \) on any regular lattice \( L_d \), Theorem 3 also identifies its dependence on the tolerance term \( \delta \) for two-dimensional lattice \( L_2 \).

A natural conjecture is that the result of Theorem 3 holds for any distribution \( F \) with finite variance. In Chapter 5, we present simulation results that support this conjecture.

### 3.4.2 For Heavy-tailed Rewards

In this chapter, we focus on the situations where the rewards follow heavy-tailed distributions. We show that, in this case, the receding-horizon planning algorithm requires a substantially larger perception range to achieve performance matching the fundamental limits. We prove this statement for the case when the rewards follow of Pareto distribution, a widely-studied heavy-tailed probability distribution.

Before we present the main results, let us first define regularly varying distributions and Pareto distributions.

**Definition 3** (Regularly Varying Distribution, See [12]). A cumulative distribution function \( F(x) \) is said to be regularly varying with index \( \alpha \in (0, 2) \), if

\[
\lim_{x \to \infty} \frac{1 - F(tx)}{1 - F(x)} = t^{-\alpha}, \quad \text{for all } t > 0.
\]

Note that a regularly varying distribution with index \( \alpha \in (0, 2) \) is also heavy-tailed, which is defined in Definition 2.

**Definition 4** (Pareto Distribution). The Pareto distribution \( F(x) \) with index parameters \( x_m \) and \( \alpha \) is defined as follows:

\[
P(X \leq x) = F(x) = 1 - \left( \frac{x_m}{x} \right)^\alpha, \quad \text{for } x \geq x_m,
\]

where \( x_m > 0 \) and \( \alpha \in (1, 2) \).
All Pareto distributions are heavy-tailed, and they are regularly varying with index $\alpha$.

Now we present the main result of this section. We consider the case when the rewards follow a Pareto distribution on two-dimensional lattice $L_2$, and we show that the receding horizon algorithm does not achieve near-optimal performance with limited perception range $m$.

**Theorem 4.** Suppose the rewards are distributed randomly over $F$ on a two-dimensional lattice $L_2$, where $F$ is a Pareto distribution with parameter $\alpha \in (1, 2)$. Then there exists a probability space $(\Omega, \mathcal{F}, P)$ such that as $m$ goes to infinity,

$$\mathbb{E}\left[\frac{Q(n;m)}{n}\right] = c \cdot m^{(2/\alpha)-1}, \text{ for all } n > m,$$

for some positive constant $c$.

See Appendix A.4 for the proof of Theorem 4. Theorem 4 implies that the unit-distance rewards collected is an increasing function of perception range $m$. To gain a better understanding of Theorem 4, see the next corollary.

**Corollary 2.** Suppose the rewards are distributed randomly over $F$ on a two-dimensional lattice $L_2$, where $F$ is a Pareto distribution with parameter $\alpha \in (1, 2)$. Then there exists a probability space $(\Omega, \mathcal{F}, P)$ such that when $L(m)$ is a super-linear function of $m$,

$$R^*(L(m)) - \mathbb{E}\left[\frac{Q(L(m);m)}{L(m)}\right] \geq c \cdot (L(m))^{(2/\alpha)-1},$$

for some positive constant $c$, as $m$ goes to infinity. $R^*(L(m))$ is the expected rewards collected with infinite horizon, when the mission length is $L(m)$.

See Appendix A.5 for the proof of Corollary 2. Compared with Theorem 3, it can be observed that with Pareto-distributed rewards, the deviation from the optimal rewards is growing with the perception range $m$, whenever $L(s)$ is increasing faster than $m$. In other words, it is impossible to obtain near-optimal performance with a small perception range, as in the case of light-tailed rewards.
Chapter 4

Motions in Continuous Spaces

In this chapter, we return to the original maximum-reward motion problem in continuous space \( \mathbb{R}^2 \) defined in Section 2.2. We will show that maximum-reward motion in continuous space is a natural extension of the discrete problem on regular lattice \( L_d \). Specifically, the continuous problem is the limiting case of the lattice-based motion in Chapter 3 when discretization of space goes to the finest.

4.1 Motion Planning Algorithm

Similar to the planning algorithm on discrete lattices, in the continuous space the planning algorithm proceeds in a receding-horizon manner. Suppose the robot starts at an initial state \( z_{\text{init}} \). The best feasible trajectory \( x_e : [0, T_e] \to X \) within the “visible” region of the lattice is computed, and the robot follows this dynamically-feasible trajectory. After the robot executes this trajectory, the same procedure is repeated. This algorithm is formalized in Algorithm 2.

More specifically, \texttt{PerceiveEnvironment()} (Line 3) is a procedure that returns \( z(t) \), which contains positions of targets and amounts of rewards associated with them within the current perception range of the robot. The robot then computes the optimal path within the set of trajectories \( \texttt{Paths}(z(t)) \) to maximize the rewards collected (Line 4). In this problem, \( \texttt{Paths} = \{ \pi : \dot{x}_1 = v, |\dot{x}_2| \leq w \} \). The procedure \texttt{Execute}(\( \pi \)) (Line 5) commands the robot to move along the planned path \( \pi : [0, m/v] \to X \). After
completion of this command, the entire procedure is repeated until time distance is greater than mission length \( L \) (Lines 2-7).

**Algorithm 2** Receding-horizon online motion planning

1: \( \text{distance}_x \leftarrow 0 \)
2: \( \text{while } \text{distance}_x < L \) do
3: \( z(t) \leftarrow \text{PerceiveEnvironment}() \)
4: \( \pi_N \leftarrow \arg \max \{ R(\text{Trajectory}(\pi) : \pi \in \text{Paths}(z(t))) \} \)
5: \( R_i \leftarrow \text{Execute}(\pi_N) \)
6: \( Q \leftarrow Q + R_i \)
7: \( \text{distance}_x \leftarrow \text{distance}_x + m \)

### 4.2 On Infinite-Horizon Mean Rewards

Recall that \( \text{Paths} = \{ \pi : \dot{x}_1 = v, |\dot{x}_2| \leq w \} \). Define \( \Pi(L) \subset \text{Paths} \) as follows.

\[
\Pi(L) = \{ \pi : [0, L/v] \rightarrow \mathbb{R}^2, \pi \in \text{Paths} \}
\]

It is the set of all paths that start from the origin and travels a distance of at most \( L \) in the \( x_1 \) axis. Let \( T(\pi) \) be the set of reward targets that fall on the trajectory \( \pi \), i.e.,

\[
T(\pi) = \{ p_i : p_i \in \pi \}
\]

Recall the assumption that the reward locations \( \{ p_i \} \) are generated by a Poisson point process with intensity \( \lambda \). The amount of rewards at each target are i.i.d. random variables \( r(p_i) \) that follow a common distribution \( F \). Let \( \mathcal{R}(L) \) denote the maximal total reward collected by following some path in \( \Pi(L) \), i.e.,

\[
\mathcal{R}(L) := \max_{\pi \in \Pi(L)} \sum_{p_i \in T(\pi)} r(p_i).
\]

The first result for the continuous problem is an extension of Theorem 1 for the discrete problem.

**Theorem 5** (Mean Maximal Reward). Suppose the reward locations are generated
by a Poisson point process with intensity \( \lambda \) on \( \mathbb{R}^2 \). The robot dynamics satisfies the following ordinary differential equation:

\[
\dot{x}_1(t) = v, \quad \dot{x}_2(t) = u(t),
\]

where \( |u(t)| \leq w = v \). If we define \( R_2^* = \sup L \frac{E[R(L)]}{L} \), then

\[
\lim_{L \to \infty} \frac{R(L)}{L} = R_2^* \quad \text{almost surely.}
\]

Moreover, if Conjecture 1 holds, then

\[
R_2^* \geq \sqrt{\lambda E[r^2]}. 
\]

The proof for Theorem 5 is given in Appendix A.6. As is in the discrete problem, the mean maximal reward in the continuous space is very close to \( R_2^* \). Note that here the robot agility \( w/v \) is fixed to be 1, and more discussion regarding agility is presented in Section 4.4.

In some applications, it is more valuable to maximize the number of targets visited, regardless of the actual amount of reward at each locations. This problem of maximizing targets visited, which are generated by a Poisson process, can be formulated as Ulam's problem in the applied probability literature [36]. Formally, let \( \mathcal{N}(\pi) \) denote the number of targets that a path \( \pi \) visits, i.e.,

\[
\mathcal{N}(\pi) := \text{card}(\mathcal{T}(\pi))
\]

The path that visits the maximal number of targets within the set \( \Pi(L) \) is

\[
\pi_N = \arg \max_{\pi \in \Pi(L)} \mathcal{N}(\pi).
\]

Contingent upon Conjecture 1, we can show that there exists a lower bound \( \frac{E[\mathcal{N}(\pi_N)]}{L} \geq \)
\[ \sqrt{\lambda}. \] This follows by defining \( r(p_i) = 1 \) for all \( p_i \) and then apply Theorem 5. However, more accurate result has been proved in the applied probability literature and is rephrased as follows with our notation.

**Theorem 6** (See [31]). *Suppose the reward locations are generated by a Poisson point process with intensity \( \lambda \) on \( \mathbb{R}^2 \). The robot dynamics satisfies the following ordinary differential equation:

\[
\dot{x}_1(t) = v, \quad \dot{x}_2(t) = u(t),
\]

where \( |u(t)| \leq w = v \). Then with the control policy \( \pi^*_N = \arg \max_{\pi \in \Pi(L)} N(\pi) \) maximizing number of targets, it follows that

\[
\lim_{L \to \infty} \frac{N(\pi^*_N)}{L} = \sqrt{\lambda} \quad \text{almost surely.}
\]

### 4.3 Performance with respect to Perception Range

Let the perception range \( m \) be a positive number, then any target attainable by a path \( \pi \in \text{Paths} \) of at most distance \( m \) on the \( x_1 \)-axis can be perceived by the robot. \( \mathcal{R}_i \) is the amount of rewards collected during the \( i^{th} \) iteration of Algorithm 2, and \( Q(L; m) \) denotes the total rewards collected with Algorithm 2 throughout the entire mission, i.e.,

\[
Q(L; m) := \sum_{i=1}^{L/m} \mathcal{R}_i.
\]

The following result extends Theorem 2 and shows that the receding horizon algorithm still has near-optimal performance even in the continuous problem, when the perception range \( m \) is at the order of \( \log L \).

**Conjecture 2.** *Suppose the reward locations are generated by a Poisson point process with intensity \( \lambda \) on \( \mathbb{R}^2 \). Suppose that these rewards \( r(p_i) \) are uniformly almost-surely bounded random variables, i.e., there exists some \( b \) such that \( \mathbb{P}(|r(p_i)| \leq b) = 1 \) for*
all $i \in \mathbb{N}$, and that $R^*_2$ is finite. The robot dynamics satisfies the following ordinary differential equation:

$$\dot{x}_1(t) = v, \quad \dot{x}_2(t) = u(t),$$

where $|u(t)| \leq w = v$. Then, for any $\delta > 0$, there exists some constant $c$ such that

$$\lim_{m \to \infty} \mathbb{P}\left( \left| \frac{Q(L(m), m)}{L(m)} - R^*_2 \right| \geq \delta \right) = 0.$$ 

where $L(m) = e^{cm}$ for some constant $c$ that is independent of $m$ (but depends on $\delta$).

The discussion for Conjecture 2 in Appendix A.7 is an attempted proof with some details remained to be worked out. It is again based on discretization of the continuous space and employing Theorem 2, similar to the proof in Appendix A.6. We leave it as a conjecture here, and show simulations results in Chapter 5.

We can further show that Theorem 4 also extends to the continuous space.

**Theorem 7.** Suppose the reward locations are generated by a Poisson point process with intensity $\lambda$ on $\mathbb{R}^d$. Suppose that these rewards $r(p_i)$ follow $F(x)$, which is a Pareto distribution with parameter $\alpha \in (1, 2)$. The robot dynamics satisfies the following ordinary differential equation:

$$\dot{x}_1(t) = v, \quad \dot{x}_2(t) = u(t),$$

where $|u(t)| \leq w = v$. Then there exists a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ such that as $m$ goes to infinity,

$$\mathbb{E}\left[ \frac{Q(L; m)}{L} \right] = c \cdot m^{2/\alpha - 1}, \forall L > m,$$

for some positive constant $c$.

### 4.4 Performance with respect to the Robot’s Agility

In this section, we examine how agility impacts the performance of the robot, measured by the total reward collected. Recall that the agility of the robot is defined in
Section 2.2 as
\[ \alpha = \frac{w}{v}, \]
where \( w \) is the bound on \( \dot{x}_2(t) \), the velocity in the lateral direction (y-axis), and \( \dot{x}_1(t) \) is a constant speed along the longitudinal direction (x-axis).

**Theorem 8.** Suppose the reward locations are generated by a Poisson point process with intensity \( \lambda \) on \( \mathbb{R}^d \). The robot dynamics satisfies the following ordinary differential equation:
\[
\dot{x}_1(t) = v, \quad \dot{x}_2(t) = u(t),
\]
where \( |u(t)| \leq w \). Then for any finite \( L > 0 \), there exists a constant \( c > 0 \) such that
\[
\mathbb{E}[R(L)] = c\sqrt{\alpha} = c\sqrt{w/v}.
\]

The proof can be found in Appendix A.8. With Theorem 8, the maximal reward for a robot with any agility \( \alpha \) other than 1 can also be computed accordingly. Theorem 8 also has a strong implication on the design of the reward-robot system; specifically, how one should balance the tradeoff between rewards and robot agility.

### 4.5 Computational Workload

In this section, we assess the amount of computational operations carried out onboard during the environmental monitoring task discussed in Chapter 1. The computational workload stems from two different tasks, including both the motion planning and the inference for each sensor data collected.

For the motion planning task, dynamic programming is applied to compute the optimal path, so the computational complexity for motion planning in this particular problem is \( O(m^2) \), which depends mainly on the perception range of the robot. The overall workload for planning can be easily handled.
The inference task, however, generally induces significantly heavier workload. For example, assume the UAV-UGS system is designed for wildlife detection and tracking in a forest. The ground sensors are capable of intermittent capture of images, storage of data, and uploading them to the UAV when it is within distance of communication. The UAV needs to process the downloaded images onboard with real-time object detection (whether a target animal is found) and localization (where the animal is in the forest) using state-of-the-art computer vision techniques. These computations usually exhausts all the computational resource of processors carried onboard. Therefore, it is important to evaluate the number of inference tasks executed over the mission, i.e., the expected number of sensors the vehicle visits.

It is easy to observe that \( N \), the number of sensors visited, has to scale at least linearly with the distance of travel \( L \). If the number of sensors visited increases sub-linearly, then as the robot moves the number of sensors it could see declines, which contradicts the assumption that the sensor distribution is ergodic. In other words, \( N = o(L) \). On the other hand, Theorem 6 dictates that the expected number of targets visited is also upper bounded by \( L\sqrt{\lambda} \), where \( L \) is the distance of travel and \( \lambda \) is the density of sensors in the field. That is, \( N = O(L) \). This follows from the fact that, by construction, \( \mathbb{E}[N(\pi_N)] = L\sqrt{\lambda} \) is the expected number of targets a robot can possibly visit, even when \( N(\pi_N) \) is exactly the quantity that the robot wants to maximize. Summarizing these two facts, we conclude that \( N = \Theta(L) \), so the overall computational workload for inference tasks is \( cL \) for some positive constant \( c > 0 \).
Chapter 5

Computational Experiments

In this chapter, we perform simulation experiments to verify the theoretical analysis.

5.1 Conjecture 1

The first set of experiments was run to verify Conjecture 1 on a two-dimensional regular lattice $L_1$. In the experiments, a two-dimensional matrix is created where each element in the matrix is the amount of rewards associated with a vertex on the regular lattice. The path with maximal rewards is computed using dynamic programming. We tested the conjecture with commonly seen distributions, including Bernoulli, exponential, geometric, and Poisson. The resulting "shape functions" from all these distributions support the conjecture.

As an example, Figure 5-1 shows the "shape functions" where rewards at each vertex follow a Poisson distribution with $p = 0.5$. This plot shows the unit-distance mean rewards $T((x, y))/(x + y)$.

5.2 Mean Rewards with Discrete Lattices

This second experiment aims to verify Theorem 1. Figure 5-2 shows the experiment where rewards $\rho$ at each vertex are sum of a random number of exponential random variables. Specifically, $\rho = \sum_{i=1}^{T} Y_i$, where $T$ is a Poisson random variable with
Figure 5-1: Each point on the surface shows the mean maximal rewards $T((x, y))/(x + y)$ for a end point $(x, y)$, when rewards are i.i.d. Poisson random variables with $p = 0.5$. This normalized shape function converges to above 1.1, which is the lower bound predicted by Theorem 1 contingent on Conjecture 1.

parameter 0.5 and $Y_i$ are independent exponential random variables with parameter 1, i.e.,

$$T \sim Poi(0.5), \quad Y_i \sim Exp(1)$$

This distribution of rewards is interesting because it resembles the case we encounter in the continuous problem. The lower bound $\mu + \sigma = 1.5$ for this distribution.

As a result of the simulation, the unit-distance mean rewards $R(n)/n$ collected by the robot, represented by the blue curve, quickly saturates and converges to an optimal value above $\mu + \sigma$, indicated by the red line. This supports Theorem 1.

In comparison, when the rewards follow a Pareto distribution, the unit-distance mean rewards collected by the robot is monotonically increasing with the mission length, as predicted by Theorem 4. See Figure 5-3 for the experiment where rewards follow a Pareto distribution with parameter $\alpha = 1.5$. In this case, the mean rewards collected grows unbounded with mission length $n$. 

46
Figure 5-2: Unit-distance mean rewards $R(n)/n$ versus mission length $n$ on a two-dimensional regular lattice. The mean rewards collected by the robot, shown by the blue curve, quickly approaches to a value above the lower bound $\mu + \sigma$ indicated by the red line.

Figure 5-3: Expected reward collected versus perception range with heavy-tailed distributions on 1 two-dimensional regular lattice. Rewards follow a Pareto distribution with $\alpha = 1.5$. The blue curve shows the mean maximal rewards $R(n)/n$ collected with a mission length $n$, and the red line is proportional to $x^{2/\alpha-1}$.
5.3 Perception Range with Discrete Lattices

This third experiment shows the impact of robot perception range on its performance, measured by the optimal distance of travel.

Recall in Theorem 2 we showed that with \( \log n \) perception range, the robot can collect rewards almost optimally, as if it could see the entire reward field, given the assumption that the reward distribution is bounded. This simulation not only shows that Theorem 2 is true, but also indicates possible existence of even stronger theoretical result. More specifically, Theorem 2 holds not only for bounded distributions as we proved, but also for all light-tailed distributions such as exponential, geometric and Poisson distributions.

We conducted experiments on the optimal distance-of-travel, i.e., the longest distance travelled without losing more than \( \delta \) rewards within any iterations, with respect to the perception range. In this experiment, the robot is instructed to move forward and collect rewards until the mean rewards collected drops below \( \mu + \sigma - \delta \). This distance-of-travel is plotted against the perception range in Figure 5-4 for rewards that follow a geometric distribution with mean 1. Note that the \( y \)-axis is on log-scale, and thus the distance is increasing exponentially fast with perception range, as is indicated by Corollary 1. Moreover, Theorem 3 identifies that the exponent depends on \( \delta^{3/2} \), which can also observed by the red line, which is a linear fit of the blue dots.

5.4 Mean Rewards in Continuous Spaces

In this set of experiments, we verify Theorem 5 and Theorem 6 for the problem in continuous space.

The reachable region of the robot in the continuous problem is an isosceles triangle region with equal sides \( L \) (distance-of-travel) and the angle between them as \( 2 \arctan \alpha \), where \( \alpha \) is the agility of the robot. The targets are generated by a Poisson point process with intensity \( \lambda \). More specifically, the total number of targets \( N \) is
Figure 5-4: Expected distance-of-travel versus perception range on a two-dimensional regular lattice for geometric-distributed rewards. The blue curve shows the mean distance-of-travel when the tolerance on loss of unit-distance rewards is $\delta = 0.08$. The red line is a linear fit of the blue dots for large perception range $m$. The $y$-axis is on log-scale.

first drawn from a Poisson distribution with density $\lambda \cdot S$, where $S$ is the area of the isosceles triangle region. Subsequently, we simulate $N$ targets uniformly in the reachable region. The rewards associated with each target are drawn i.i.d. from some distribution $F$ of our choice. To simplify the computational process, we use a larger square region that contains the isosceles triangle when we create these targets and rewards. This implies that some targets are in fact outside the reachable set and are never visited.

Figure 5-5 shows simulation results where the intensity $\lambda$ of the Poisson point process is 1 and the rewards follow an exponential distribution with mean 1. The red dots are the unit-distance rewards obtained from the maximum-reward motion versus mission length, while the blue dots are those from the strategy that maximizes visited targets with respect to mission length. Both the mean rewards quickly converges with the entire mission length. In addition, the rewards from maximum-reward motion are lower bounded by $\sqrt{\lambda E[r^2]}$ (the horizontal red line) as length goes up to infinity, as predicted by Theorem 5. The rewards from the maximum-target motion, on the other
hand, converges to $\sqrt{\lambda}$ (the horizontal magenta line), as dictated by Theorem 6.

![Mean Rewards vs. Mission Length](image)

Figure 5-5: Expected reward versus mission length in continuous space $\mathbb{R}^2$. The red dots are the unit-distance rewards obtained from the maximum-reward motion versus distance-of-travel, while the blue dots are those from the strategy that maximizes visited targets. The horizontal lines are the theoretical lower bound and the limit, respectively.

5.5 Perception Range in Continuous Spaces

In this experiment, we verify Conjecture 2 and explore the impact of robot perception range on its performance in the continuous problem. A receding-horizon algorithm is used for motion planning. That is, the robot plans an optimal path within its current perception range, executes this path, and collects rewards on the path. This process is repeated until the amount of rewards collected within one iteration falls below $\sqrt{\lambda \mathbb{E}[r^2]} - \delta$, where $\delta$ is a user-specified tolerance on loss of performance.

Figure 5-6 shows the results of this experiment, where the Poisson process is parameterized with $\lambda = 1$ and each rewards $r_i$ follows an exponential distribution with parameter 1. The average distance-of-travel is plotted against perception range $m$ of the robot. Note that the y-axis is on log scale, so the distance of travel increases exponentially fast with perception range $m$, when $m$ is sufficiently large, as dictated
by Conjecture 2. Note that although Conjecture 2 only predicts such exponential
growth for bounded rewards, our simulation shows that the same conclusion holds
true for general light-tailed rewards.

\[ \delta = 0.125 \]

Figure 5-6: The average distance-of-travel is plotted against perception range \( m \) of
the robot. The Poisson process is parameterized with \( \lambda = 1 \) and each rewards \( r_i \)
follows an exponential distribution with parameter 1. Note that the y-axis is on log
scale, so the distance of travel increases exponentially fast with perception range \( m \),
when \( m \) is sufficiently large (even for such unbounded light-tailed rewards).

5.6 Robot Agility in Continuous Space

These last experiments were run to demonstrate how robot agility impacts the per-
formance. See Figure 5-7 for the simulation where the intensity \( \lambda \) of the Poisson point
process is 10 and the rewards follow an exponential distribution with mean 1. The
distance-of-travel is fixed to be \( L = 30 \), and each data point is averaged over 300
iterations. As can be observed, with relatively high level of agility, the mean rewards
collected are proportional to the square root of robot agility. We also show a 3D plot
of mean rewards versus both the perception range and the robot agility in Figure 5-8
for exponentially-distributed rewards.
Figure 5-7: Expected reward versus robot agility for the simulation where the intensity \( \lambda \) of the Poisson point process is 10 and the rewards follow an exponential distribution with mean 1.

Figure 5-8: Expected reward versus both the perception range and the agility for the simulation where the intensity \( \lambda \) of the Poisson point process is 1 and the rewards follow an exponential distribution with mean 1.
Chapter 6

Application: Sensor Selection

In this chapter we introduce the sensor selection problem as a concrete application of the maximum-reward motions. Again, recall the UAV-UGS system in Chapter 1. One major application of such systems lies in estimation of certain unknown (but fixed) values. For example, estimating the density of trees or animals in a forest, level of pollutions in the urban environments, probability of a volcanic eruption and so on. We are interested in the question that, given a fixed UAV and limited budget in deployment of ground sensors, how should one balance between the number and the quality of sensors?

To answer this question, let’s formulate the above mentioned applications as the classic static estimation problem in statistics. Specifically, we want to estimate the value of a fixed scalar $\theta$. Assume our prior belief on $\theta$ is Gaussian with mean $\mu_0$ and variance $\frac{1}{\beta_0}$, i.e.,

$$\theta \sim \mathcal{N}(\mu_0, \frac{1}{\beta_0}).$$

Suppose there are different sensors that provide measurements of $\theta$. The likelihood function of the sensor measurement $y_i$ given $\theta$ is a Gaussian distribution centered at $\theta$ with variance $\frac{1}{\beta_i}$, i.e.,

$$y_i | \theta \sim \mathcal{N}(\theta, \frac{1}{\beta_i}).$$

Given sensor measurements $y = [y_1, y_2, \ldots, y_n]$, we can compute the posterior prob-
ability of \( \theta \) conditioning on \( y \) using Bayes’ rule

\[
\theta | y \sim \mathcal{N}(\mu_n, \frac{1}{\beta_n}),
\]

with the updated mean \( \mu_n \) and variance \( \frac{1}{\beta_n} \) satisfying

\[
\beta_n' = \beta_0 + \beta_1 + \cdots + \beta_n.
\] (6.1)

Suppose the sensors are randomly distributed in the field, and a robot is tasked with estimating \( \theta \) by navigating in the field and collecting sensor measurements. The quality of the planning algorithm is measured by the final posterior precision \( \beta_n' \). This is exactly the maximum-reward motion problem; specifically, in this case the “reward” is the precision gain \( \beta_i \) after visiting a sensor.

Now suppose we have the option to choose the sensors before they are distributed in the field. Due to limited budgets, the average quality of sensors is fixed and the total number of sensors is given, i.e.,

\[
E[\beta_i] = \mu_\beta,
\]

where \( \mu_\beta \) is some positive constant and \( \lambda \) is known. With above constraints, we want to address the following question: Which one of the following two strategies yields a higher level of confidence for the estimation?

1. Assign the same level of precision to all sensors, i.e., \( \beta_i = \mu_\beta \) for all \( i \)

2. Randomize the level of precision \( \beta_i \) over some probability distributions \( F_\beta \) with mean \( \mu_\beta \)

By now we can see that this sensor selection problem is an instance of the reward-collection problem in two dimensional space. In this case, the reward is the precision gain \( \beta_i \) after getting the observation from a sensor.

We compare the two strategies. The first strategy assigns equal precision to all sensor. As a result, the robot should visit as many sensors as possible in order to
maximize the total precision gain. Based on Theorem 6, the average number of sensors visited is $\sqrt{\lambda}$, and hence the precision gain would be $\sqrt{\lambda} \cdot \mathbb{E}[\beta]$. The second strategy, on the other hand, utilizes sensors with random precisions. According to Theorem 5, the average precision gain is

$$\sqrt{\lambda} \cdot R^*_2 \geq \sqrt{\lambda} \cdot \sqrt{\mathbb{E}[\beta^2]} \geq \sqrt{\lambda} \cdot \mathbb{E}[\beta].$$

In other words, a random distribution of sensor precision outperforms homogeneous sensors in terms of the level of confidence for the inference problem. More surprisingly, the more variations in sensor quality $\beta_i$, the better we expect the final inference outcome to be.
Chapter 7

Conclusions and Remarks

This thesis studied the maximum-reward motion problem, where an autonomous mobile robot operates in a stochastic reward field in an online setting, with the aim to maximize total rewards collected. This mathematical model is an abstraction of many robotic applications arising from various areas. Examples include, but are not limited to, unmanned aerial vehicle (UAV) collecting data from randomly deployed unattended ground sensors (UGS), transportation ships unloading cargo to various locations, and surveillance planes taking pictures of targets, where the targets in these applications are all discovered on the fly.

We started out with the most general definition of the maximum-reward motion problem and later boiled down to a concrete model in two dimensional continuous spaces. In this special case, we assumed that the rewards are generated by a Poisson point process. The locations and exact amount of rewards associated with each target are not known a priori to the robot. Instead, only the statistics of the rewards are given. The robot is assumed to have a limited perception range, and thus it discovers the reward field on the fly. It is also assumed to be a dynamical system with substantial drift in one direction, such as a high-speed airplane or ground vehicle. This assumption implies that the robot cannot traverse the entire field, which separates itself from the traditional traveling salesman problem (TSP) and the dynamic vehicle routing problems. The robot is assigned the task of maximizing the total rewards collected during the course of the mission, given above constraints.
The theoretical results were structured in two main parts. In the first part, we proposed an approximate problem to the original continuous problem by using lattice-based discretization. This discretization technique enabled us to design practical planning algorithm, i.e., receding-horizon motion planning on the lattice, with tractable computational time. By building a strong connection with statistical mechanics, we were able to analyze the performance of such algorithms and provide performance guarantees. Specifically, we gave a lower bound on $R^*$, the expected rewards collected with infinite horizon. We then proceeded to show that when rewards follow a light-tailed distribution, the receding-horizon motion planning algorithm provides an average reward that is very closed to $R^*$, even when the perception range $m$ is on the order of $\log n$, with $n$ being the mission length. On the other hand, when rewards are heavy-tailedly distributed, the performance of receding-horizon motion planning is significantly worse than planning with infinite horizon.

The second part of the theoretical study focused on the original continuous problem. We analyzed the performance of the receding-horizon planning algorithm with respect to the perception range, robot agility and computational resources available. Similar results with perception range extends from the discrete case, i.e., near-optimal performance is attained with log-distance perception range if and only if the rewards are light-tailed distributed. We also showed that the performance of the robot increases with the square root of robot agility, which has strong implications on tradeoff in UAV-UGS system design. The computational workload for the UAV was also studied, and we proved that the overall computational workload increases linearly with the mission length, in some inference tasks. All theoretical results are verified using simulation examples.

Finally we presented one interesting application of our theoretical study to the ground sensor selection problem. Under certain Gaussian noise assumptions, we proved that sensors with randomized quality outperform homogeneous sensors for some inference/estimation tasks. Specifically, by formulating this as the maximum-reward motion problem and employing theorems from previous chapters, we proved that random sensors yield a higher confidence level of estimation (lower variance).
For potential future work, we would like to analyze more practical (and better) planning algorithms. In this thesis, we studied a sub-optimal algorithm, where the robot does not update information about the world or recompute its path until the incumbent task is completed. A better alternative algorithm re-plans whenever a new target is discovered, and thus outperforms the current one with respect to average rewards collected. However, analysis of such algorithm requires extra complications in proof techniques. We would also like to study performance of robots with more complicated dynamical models, such as a Dubins vehicle, in maximum-reward motion problems. A much more challenging and interesting work would be to incorporate obstacle avoidance constrains in the problem formulation.
Appendix A

Proofs

A.1 Proof for Theorem 1

Proof. The following holds:

\[ R^* := \sup_{n \in \mathbb{N}} \frac{\mathbb{E}[R(n)]}{n} \quad (A.1) \]
\[ \geq \sup_{n \in \mathbb{N}} \frac{\mathbb{E}[R(2n)]}{2n} \quad (A.2) \]
\[ = \sup_{n \in \mathbb{N}} \frac{\mathbb{E}[\max_{v \in \{w:x+y=2n\}} T(v)]}{2n} \quad (A.3) \]
\[ \geq \sup_{n \in \mathbb{N}} \mathbb{E}\left[ \frac{T((n,n))}{2n} \right] \]
\[ = \mathbb{E}[g((1,1))]/2 \quad (A.4) \]
\[ = \mu + \sigma \]

Line A.1 is the definition of \( R^* \); Line A.3 is the definition of \( R(n) \); Line A.4 uses Proposition 1, and the last line is simply Conjecture 1. \( \square \)

A.2 Proof for Theorem 2

Before proving Theorem 2, we state an intermediate result that enables our proof. This intermediate result is a concentration inequality, which plays a key role in de-
riving many results in nonequilibrium statistical mechanics [21].

**Lemma 1** (See [21]). Let \( \{Y_i, i \in I\} \) be a finite collection of independent random variables that are bounded almost surely, i.e., \( \mathbb{P}(|Y_i| \leq L) = 1 \) for all \( i \in I \). Let \( C \) be a collection of subsets of \( I \) with maximum cardinality \( R \), i.e., \( \max_{C \in C} |C| \leq R \) and let \( Z = \max_{C \in C} \sum_{i \in C} Y_i \). Then for any \( u > 0 \),

\[
\mathbb{P}(|Z - \mathbb{E}Z| \geq u) \leq \exp \left(-\frac{u^2}{64RL^2} + 64\right).
\]

Finally, we present the proof for Theorem 2.

**Theorem 2.** Let \( I \) be the collection of nodes in the lattice. Define \( C = \{N(\pi), \pi \in \Pi\} \), where \( N(\pi) = \{v \in \pi\} \) is the set of nodes in the path \( \pi \). Then, for the maximum-reward path with at most \( n \) steps, the maximum cardinality is \( \max_{C \in C} |C| \leq n \). Then, by substituting \( R(n) \) for \( Z \) in Lemma 1,

\[
P(|R(n) - \mathbb{E}R(n)| \geq u) \leq \exp \left(-\frac{u^2}{64nL^2} + 64\right).
\]

Therefore, for any \( \delta = \frac{u}{n} > 0 \),

\[
P \left( \left| \frac{Q(n, m) - \mathbb{E}R(n)}{n} \right| \geq \delta \right)
= P \left( \left| \sum_{i=1}^{n} \frac{R_i(m)}{n} - \frac{\mathbb{E}R(n)}{n} \right| \geq \delta \right)
= P \left( \left| \sum_{i=1}^{n} \frac{R_i(m)}{n} - \frac{\mathbb{E}R(m)}{m} + \frac{\mathbb{E}R(m)}{m} - \frac{\mathbb{E}R(n)}{n} \right| \geq \delta \right)
\leq P \left( \left\{ \left| \sum_{i=1}^{n} \frac{R_i(m)}{n} - \frac{\mathbb{E}R(m)}{m} \right| \geq \frac{\delta}{2} \right\} \cup \left\{ \left| \frac{\mathbb{E}R(m)}{m} - \frac{\mathbb{E}R(n)}{n} \right| \geq \frac{\delta}{2} \right\} \right),
\]

\[
\leq P \left( \left| \sum_{i=1}^{n} \frac{R_i(m)}{n} - \frac{\mathbb{E}R(m)}{m} \right| \geq \frac{\delta}{2} \right) + P \left( \left| \frac{\mathbb{E}R(m)}{m} - \frac{\mathbb{E}R(n)}{n} \right| \geq \frac{\delta}{2} \right)
\]

\[
= \mathbb{P} \left( \sum_{i=1}^{n} \left| R_i(m) - \mathbb{E}R(m) \right| \geq \frac{n\delta}{2} \right) + \mathbb{P} \left( \left| \frac{\mathbb{E}R(m)}{m} - \frac{\mathbb{E}R(n)}{n} \right| \geq \frac{\delta}{2} \right).
\]

The inequality between line (A.5) and line (A.6) can be seen if we take the com-
plements on both sides, where \( \{ (\sum_{i=1}^{m} R_i(m) - \frac{\mathbb{E}R(m)}{m}) + (\frac{\mathbb{E}R(m)}{m} - \frac{\mathbb{E}R(n)}{n}) \} < \delta \} \supset \{ (\sum_{i=1}^{m} R_i(m) - \frac{\mathbb{E}R(m)}{m}) < \frac{\delta}{2} \} \cap \{ (\frac{\mathbb{E}R(m)}{m} - \frac{\mathbb{E}R(n)}{n}) < \frac{\delta}{2} \} \). Union bound is applied between between line (A.6) and line (A.7). Now we set \( m = c \log n \). Taking limit on both sides, we get

\[
\lim_{n \to \infty} \mathbb{P} \left( \left| \frac{Q(n, m)}{n} - \frac{\mathbb{E}R(n)}{n} \right| \geq \delta \right) \\
\leq \lim_{n \to \infty} \mathbb{P} \left( \sum_{i=1}^{m} |R_i(m) - \mathbb{E}R(m)| \geq \frac{n\delta}{2} \right) + \lim_{n \to \infty} \mathbb{P} \left( \left| \frac{\mathbb{E}R(m)}{m} - \frac{\mathbb{E}R(n)}{n} \right| \geq \frac{\delta}{2} \right) \tag{A.9}
\]

\[
\leq \lim_{n \to \infty} \mathbb{P} \left( \sum_{i=1}^{m} |R_i(m) - \mathbb{E}R(m)| \geq \frac{m\delta}{2} \right) \tag{A.10}
\]

\[
\leq \lim_{n \to \infty} \sum_{i=1}^{n} \mathbb{P} \left( |R_i(m) - \mathbb{E}R(m)| \geq \frac{m\delta}{2} \right) \tag{A.11}
\]

\[
\leq \lim_{n \to \infty} \frac{n}{m} \cdot \exp \left( -\frac{(m\delta)^2}{64mL^2} + 64 \right) \tag{A.12}
\]

\[
= \lim_{n \to \infty} \frac{1}{c \log n} \cdot \exp \left( \left(1 - \frac{\delta^2}{256L^2} \cdot c \right) \log n + 64 \right) \tag{A.13}
\]

The first inequality comes from line (A.8). The inequality between line (A.9) and line (A.10) is due to Proposition 2. As \( n \) increases, \( m \to \infty \), and thus both \( \frac{\mathbb{E}R(m)}{m} \) and \( \frac{\mathbb{E}R(n)}{n} \) converge to the same constant \( R^* \). Union bound is again applied between between line (A.10) and line (A.11). Lemma 1 is applied in line (A.12). Line(A.13) converges to 0 when the constant \( c \) is sufficiently large, i.e., for any constant \( c \geq \frac{256L^2}{\delta^2} \). \( \Box \)

### A.3 Proof for Theorem 3

The proof of Theorem 3 is similar to that of Theorem 2 in Appendix A.2. We omit the full proof; but we outline the main differences.

**Proof.** Let \( TW \) be a random variable with the Tracy-Widom distribution. Then, the
results in [15] imply the following: For all \( u \geq 0 \),

\[
\mathbb{P}(TW \geq u) = \lim_{n \to \infty} \mathbb{P}
\left( \frac{R(n) - n^2}{n^{1/3}} \geq u \right)
\]

\[
= \lim_{n \to \infty} \mathbb{P}
\left( \frac{n^{2/3} (R(n)/n - R^*) \geq u} \right)
\]

\[
= \lim_{n \to \infty} \mathbb{P}
\left( \frac{R(n)}{n} - R^* \geq u n^{-2/3} \right)
\]

Define \( \delta := u n^{-2/3} \). Hence, \( u = \delta n^{2/3} \). It was showed very recently [9] that the right tail of the Tracy-Widom distribution \( F_2 \) can be characterized as follows:

\[
\lim_{u \to \infty} \mathbb{P}(TW \geq u) = \alpha \exp \left( -\frac{4}{3} u^{3/2} \right).
\]

Combining this with the previous equality, we obtain:

\[
\lim_{n \to \infty} \mathbb{P}
\left( \frac{R(n)}{n} - R^* \geq \delta \right)
\]

\[
= \alpha \exp \left( -\frac{4}{3} (\delta n^{2/3})^{3/2} \right)
\]

\[
= \alpha \exp \left( -\frac{4}{3} \delta^{3/2} n \right).
\]

The rest of the proof follows the proof of Theorem 2. \( \square \)

### A.4 Proof for Theorem 4

Let’s first introduce a lemma that is useful for our proof.

**Lemma 2** (See [12]). Suppose the CDF \( F(x) \) is regularly varying with index \( \alpha \in (0, 2) \). Let \( a_N = F^{-1}(1 - 1/N) \), for all \( N \in \mathbb{N} \). Then, \( a_n^{-1} R(n) \) converges in distribution to a random variable \( T \) that is almost surely finite, and moreover \( E[T^3] \) is finite for all \( 0 < \beta < \alpha \).

**Proof for Theorem 4.** Recall that \( a_N = F^{-1}(1 - 1/N) \). Then, for the Pareto distribution with parameters \( x_m > 0 \) and \( \alpha \in (0, 1) \), we have that \( a_n^{-2} R_t(m) \) converges to a random variable \( T \) in distribution, where
By Skorokhod’s Theorem [3], we can construct another probability space \( (\Omega, \mathcal{F}, P) \) where

\[
m^{-2/\alpha} R_i(m) \to T \text{ a.s.}
\]

This implies that

\[
\lim_{m \to \infty} \mathbb{E}[m^{-2/\alpha} R_i(m)] = \mathbb{E}[T]. \tag{A.14}
\]

Now we can compute the following

\[
\lim_{m \to \infty} \mathbb{E} \left[ \frac{Q(n; m)/n}{m^{2/\alpha-1}} \right] = \mathbb{E}[T]. \tag{A.15}
\]

\[
= \lim_{m \to \infty} \mathbb{E} \left[ \frac{1}{m^{2/\alpha-1}} \cdot \frac{\sum_{i=1}^{n/m} R_i(m)}{n/m \cdot m} \right] \tag{A.16}
\]

\[
= \lim_{m \to \infty} \mathbb{E} \left[ \frac{1}{m^{2/\alpha-1}} \cdot \frac{R_i(m)}{m} \right] \tag{A.17}
\]

\[
= \lim_{m \to \infty} \mathbb{E} \left[ m^{-2/\alpha} R_i(m) \right] \tag{A.18}
\]

\[
= \mathbb{E}[T]. \tag{A.19}
\]

Line A.16 is simply the definition of \( Q(n; m) \). The equation between Line A.16 and Line A.17 is due to linearity of expectation. Equation (A.14) is applied between Line A.18 and Line A.19. This implies that

\[
\mathbb{E}[Q(n; m)/n] = \mathbb{E}[T] \cdot m^{2/\alpha-1},
\]

asymptotically for large \( m \).
A.5 Proof for Corollary 2

Proof. The first term comes from the fact that $R^* = \lim_{n \to \infty} E \left[ \frac{Q(n;m)}{n} \right]$ and the second one results from plugging in $n = L(m)$. Subtracting these two terms and using Theorem 4, it then follows that for some $c_1 > 0$,

$$R^* - E \left[ \frac{Q(L(m); m)}{L(m)} \right] = \lim_{m \to \infty} E \left[ \frac{Q(L(m); L(m))}{L(m)} \right] - E \left[ \frac{Q(L(m); m)}{L(m)} \right]$$

$$\geq c_1 \left((L(m))^{(2/\alpha)-1} - m^{(2/\alpha)-1}\right)$$

(A.20)

(A.21)

Let $K > 0$ be some sufficiently large real number, then there must exists constant $c_2 \in (0, c_1)$ such that $L(m) \geq c_2 m$ for all $m > K$, since $L(m)$ is a super-linear function of $m$. Therefore, for $m > K$, we have

$$c_1 \left((L(m))^{(2/\alpha)-1} - m^{(2/\alpha)-1}\right) \geq (c_1 - c_2)(L(m))^{(2/\alpha)-1}$$

Inserting back to Equation (A.21) and let $c = c_1 - c_2$, it follows that

$$R^* - E \left[ \frac{Q(L(m); m)}{L(m)} \right] \geq c \cdot (L(m))^{(2/\alpha)-1},$$

which completes the proof. \qed

A.6 Proof for Theorem 5

Proof. To analyze the optimal performance that any robotic vehicle can possibly achieve, we approximate the continuous reward field with a two-dimensional $N \times N$ regular lattice. This approximation turns the continuous problem into a discrete problem that we are already familiar with from previous discussions.

From the assumptions, the reward field is a square of size $L \times L$, and the reward points are distributed randomly in the field according to a Poisson process with
parameter λ. Therefore, each grid on the discretized lattice has an area of

\[ s = \frac{L^2}{N^2}. \]

Therefore, the number of targets within any grid follows a Poisson distribution with intensity

\[ p = \lambda \cdot s = \lambda \frac{L^2}{N^2}. \]

Let’s denote the number of targets in any grid \((i, j)\) as \(T\). Therefore, \(\mathbb{E}[T] = p\), \(\text{var}(T) = p\), and the second moment is

\[ \mathbb{E}[T^2] = \mathbb{E}[T]^2 + \text{var}(T) = p^2 + p. \]

The amount of rewards associated with any target is a random variable \(r(p_i)\), which follows some light-tailed distribution \(F\). Let’s denote the total amount of reward in a grid to be

\[ R = \sum_{i=1}^{T} r(p_i). \]

This quantity is the sum of a random number of random variables. Note that \(r(p_i)\) is independent of \(T\), and by linearity of expectations we have

\[ \mathbb{E}[R] = \mathbb{E} \left[ \sum_{i=1}^{T} r(p_i) \right] = \mathbb{E}[T] \cdot \mathbb{E}[r(p_i)] = p \mathbb{E}[r(p_i)] \]
The variance of $\mathcal{R}$ can also be computed as

\[
\text{var}[\mathcal{R}] = \mathbb{E}[\mathcal{R}^2] - \mathbb{E}[\mathcal{R}]^2
\]

Apply Theorem 1 to derive the lower bound on the reward per step on the discrete lattice, i.e.,

\[
R_2^* \geq \mathbb{E}[\mathcal{R}] + \sqrt{\text{var}(\mathcal{R})} = p\mathbb{E}[r] + \sqrt{p\mathbb{E}[r^2]}.
\]

Here we fix the size $s$ of each grid, i.e., $L/N$ is a constant. Therefore, when the length of the reward field $L$ goes to infinity, the lattice size $N$ also diverges. The optimal reward normalized by the length of the region $L$ can be computed as

\[
\lim_{L \to \infty} \frac{\mathbb{E}[\mathcal{R}(L)]}{L} = \lim_{N \to \infty} \frac{\mathbb{E}[\mathcal{R}(N)]}{N} \cdot \frac{N}{L}, \text{ chain rule}
\]

As the discretization gets finer, $L/N$ goes to zero and the approximation becomes exact. Therefore, in the continuous case where $N/L \to \infty$, the first term vanishes and we get

\[
\lim_{L \to \infty} \frac{\mathbb{E}[\mathcal{R}(L)]}{L} \geq \sqrt{\lambda \mathbb{E}[r^2]}
\]
A.7 Discussion for Conjecture 2

We would like to show Conjecture 2 by leveraging the same discretization technique applied in Appendix A.6. We approximate the continuous reward field with a 2-dimensional regular lattice $L_2$. This approximation turns the continuous problem into a discrete problem that we are already familiar with.

Let’s first define $R(N)$ as the total rewards collected on the lattice with infinite horizon, when we discretize the continuous space with a $N \times N$ lattice. Similarly, let $R^*$ denote the total rewards collected in the continuous space with infinite horizon. Then with adaptation of notations, we have a lemma from [26] which is useful in our proof.

**Lemma 3** (See Theorem 1 in [26]). For a fixed, finite mission length $L$,

$$\lim_{N \to \infty} R(N) = R^* \quad \text{almost surely}$$

Moreover, for any $\delta > 0$, there are $c, C > 0$ such that

$$P(|R(N) - R^*| > \delta) < Ce^{-cN}$$

holds for all $N \in \mathbb{N}$.

Lemma 3 implies that with sufficiently fine discretization ($N \to \infty$), the rewards collected from the lattice approach the ones from the continuum fast. Now we proceed to the attempted proof for Conjecture 2.

**Proof.** Recall that the rewards collected using a receding-horizon algorithm with a perception range $m$ is

$$Q(L; m) = \sum_{i=1}^{L/m} R_i$$
Let’s denote the rewards collected from the $N \times N$ lattice with the same perception range as $Q_N(N; m\frac{N}{L})$, where

$$Q_N(N; m\frac{N}{L}) = \sum_{i=1}^{L/m} R_i(N)$$

Then with Lemma 3 we have

$$\lim_{N \to \infty} Q_N(N; m\frac{N}{L}) = Q(L; m) \quad \text{almost surely} \quad (A.22)$$

because each $R_i(N)$ converges to $R_i$.

By assumption, the number of targets within a finite region is bounded and the rewards $r(p_i)$ are also uniformly bounded. Consequently, the amount of rewards on each site on the lattice must be bounded. Now we can apply Theorem 2 directly. For any $\delta > 0$, there exists a constant $c_1 > 0$ such that

$$\lim_{N \to \infty} \mathbb{P}\left( \left| \frac{Q_N(N, c_1 \log N)}{N} - \frac{R(N)}{N} \right| \geq \delta \right) = 0.$$ 

where $\frac{R(N)}{N}$ is the optimal rewards per step on the lattice with infinite horizon. Assuming $L/N$ being constant (fixed resolution) and multiplying both sides of the inequality with $N/L$, it follows that

$$\lim_{N \to \infty} \mathbb{P}\left( \left| \frac{Q_N(N, c_1 \log N)}{L} - \frac{R(N)}{L} \right| \geq \frac{\delta N}{L} \right) = 0. \quad (A.23)$$

Therefore,

$$\mathbb{P}\left( \left| \frac{Q(L; m)}{L} - R^*_2 \right| \geq \delta \right) \quad (A.24)$$

$$\leq \mathbb{P}\left( \left\{ \left| \frac{Q(L; m)}{L} - R^*_2 \right| \geq \frac{\delta}{2} \right\} \cup \left\{ \left| \frac{R^*}{L} - R^*_2 \right| \geq \frac{\delta}{2} \right\} \right)$$

$$\leq \mathbb{P}\left( \left| \frac{Q(L; m)}{L} - R^*_2 \right| \geq \frac{\delta}{2} \right) + \mathbb{P}\left( \left| \frac{R^*}{L} - R^*_2 \right| \geq \frac{\delta}{2} \right) \quad (A.25)$$

The second term in Line A.25 goes to 0 as $L \to \infty$ by Theorem 5, so we can focus on
the first term.

\[
P \left( \left| \frac{Q(L; m)}{L} - \frac{R^*}{L} \right| \geq \delta/2 \right)
\]

\[
= P \left( \left| \frac{Q(L; m)}{L} - \frac{Q_N(N; m L)}{L} \right| \geq \delta/6 \right) \cup \left\{ \left| \frac{Q_N(N; m L)}{L} - \frac{R(N)}{L} \right| \geq \delta/6 \right\}
\]

\[
\cup \left\{ \left| \frac{R(N)}{L} - \frac{R^*}{L} \right| \geq \delta/6 \right\}
\]

\[
\leq P \left( \left| \frac{Q(L; m)}{L} - \frac{Q_N(N; m L)}{L} \right| \geq \delta/6 \right) + P \left( \left| \frac{Q_N(N; m L)}{L} - \frac{R(N)}{L} \right| \geq \delta/6 \right) + P \left( \left| \frac{R(N)}{L} - \frac{R^*}{L} \right| \geq \delta/6 \right)
\]

(A.27)

Note that the inequality between Line A.26 and Line A.27 holds for arbitrary \( N \).

For a fixed \( L > 0 \), the first term in Line A.27 goes to zero as \( N \to \infty \) because of Equation (A.22). On the other hand, the third term approaches 0 following Lemma 3 when \( N \to \infty \) and \( m = O(\log L) \). As a result of Equation (A.23), the second term vanishes when \( L/N \) is a constant, \( N \) goes to infinity, and \( m = c_2 \log N = c_2 \log L^N c_2 \log L + c_3 \). It is tricky here because the second term requires a constant \( L/N \), i.e., a fixed resolution, and \( L \to \infty \). In contrast, the the other two terms vanish when \( L \) is fixed and \( N \) diverges, i.e., when the resolution increases. Therefore, with some extra work, we might have

\[
P \left( \left| \frac{Q(L; m)}{L} - \frac{R^*}{L} \right| \geq \delta/2 \right) \to 0,
\]

when \( m = O(c_2 \log L) \), \( L/N \) is a constant and \( N \to \infty \). Plugging it back into Line A.25, it follows that

\[
\lim_{L \to \infty} P \left( \left| \frac{Q(L; c \log L)}{L} - R^* \right| \geq \delta \right) = 0,
\]

which could complete the proof with a change of variables. \quad \square
A.8 Proof for Theorem 8

*Proof.* When $\alpha = 1$, the robot is operating on an isosceles right triangle region, as shown in Figure A-1a. However, if $\alpha \neq 1$, the robot is operating on some isosceles triangle with the vertex angle being $2\arctan \alpha$, as shown in Figure A-1b.

![Figure A-1: The reachable set of the robot with drift, whose dynamics is described mathematically by Equations (2.1).](image)

We derive all previous results on this isosceles triangle region by using invariance of Poisson point processes. More specifically, we convert this problem with non-unit $\alpha$ back to the one in a right triangle by undergoing a deterministic transformation of the Poisson point process.

This transformation process is characterized by the mapping theorem of Poisson point process in [35]. Formally, let $f : \mathbb{R}^2 \to \mathbb{R}^2$ be the transformation between two Poisson point processes. If $f$ is an affine transformation

$$f(x) = Ax + b,$$

with $A \in \mathbb{R}^{2 \times 2}$ being invertible, then the transformed Poisson has the new intensity

$$\nu(y) = \frac{1}{|A|} \lambda \left( f^{-1}(y) \right) \left( A^{-1}(y - b) \right),$$

where $\lambda$ is the intensity of the original Poisson point process.

In this case, the transformation from an isosceles triangle to an isosceles right
triangle is linear. Therefore, \( f = Ax, \) where

\[
A = \begin{bmatrix}
1 & 0 \\
0 & 1/\alpha
\end{bmatrix}.
\]

Geometrically, this transformation stretches (or compresses) the isosceles triangle vertically until the vertex angle is 90 degrees, as shown in Figure A-1b. Note that we assume \( \lambda \) is constant in this problem, and therefore the new intensity is \( \nu = \alpha \cdot \lambda \). Therefore, the maximum-reward motion problem in \( \mathbb{R}^2 \), where a robot has agility \( \alpha \) and the reward is generated by a Poisson point process with intensity \( \lambda \), is equivalent to the maximum-reward motion problem with agility 1 and intensity \( \alpha \lambda \), with respect to the mean rewards collected.

\[\square\]
Bibliography


