**Title**
Path Planning and Communication Strategies to Enable Connectivity in Robotic Systems



**Author**
Muralidharan, Arjun

**Publication Date**
2018

Peer reviewed|Thesis/dissertation

University of California
Santa Barbara

# Path Planning and Communication Strategies to Enable Connectivity in Robotic Systems

A dissertation submitted in partial satisfaction
of the requirements for the degree

Doctor of Philosophy
in
Electrical and Computer Engineering

by

Arjun Muralidharan

Committee in charge:

Professor Yasamin Mostofi, Chair
Professor Francesco Bullo
Professor João Hespanha
Professor Jason Marden

December 2018

The Dissertation of Arjun Muralidharan is approved.

Professor Francesco Bullo

Professor João Hespanha

Professor Jason Marden

Professor Yasamin Mostofi, Committee Chair

September 2018

Path Planning and Communication Strategies to Enable Connectivity
in Robotic Systems

To Amma, Acha, and Aishoo

## Acknowledgements

First and foremost, I would like to thank my advisor Professor Yasamin Mostofi for all her support and guidance through the years. I have learnt a lot from her; from figuring out what are the questions worth asking to how one clearly organizes and presents ideas. She has significantly influenced how I think about and do research. Her enthusiasm for research, and uncanny eye for detail are qualities I hope to emulate going forward.

I would also like to express my gratitude to Professors Francesco Bullo, João Hespanha, and Jason Marden for serving on my committee, as well as for all the interesting discussions we have had over the years.

A significant portion of my time here at UCSB has been spent with lab members: Chitra, Belal, Saandeep, Herbert, and Yuan. I've truly enjoyed every moment of working (and not working) alongside them. They have been steadfast friends and have patiently endured my bouncing ideas off them. Special thanks to Chitra for carefully providing feedback on many of my drafts.

I thank Tamheed, Gokul, and Manish, for taking me in when I first arrived at Santa Barbara; my roommates past and present Gaurav, Srikanth, Gosavi, Badki and Talele, for putting up with me; and all the other friends I've made over the years, who have made my time here at beautiful Santa Barbara a true delight.

Finally, I thank my parents, my sister, and my grandparents, for all their love.

## Curriculum Vitæ
Arjun Muralidharan

### Education

| | |
|---|---|
| 2018 | Ph.D. in Electrical and Computer Engineering (Expected), University of California, Santa Barbara. |
| 2014 | M.S. in Electrical and Computer Engineering, University of California, Santa Barbara. |
| 2012 | B.Tech in Electronics and Communication Engineering, Indian Institute of Technology Guwahati. |

### Publications

Journal:

- A. Muralidharan and Y. Mostofi, "Path Planning for Minimizing the Expected Cost until Success," *IEEE Transactions on Robotics* (under review).
- A. Muralidharan and Y. Mostofi, "Statistics of the Distance Traveled until Connectivity for Unmanned Vehicles," *Autonomous Robots, special issue on Robot Communication Challenges* (submitted).
- A. Muralidharan and Y. Mostofi, "Energy Optimal Distributed Beamforming using Unmanned Vehicles," *IEEE Transactions on Control of Network Systems*, 2017.
- S. Depatla, A. Muralidharan and Y. Mostofi, "Occupancy Estimation using only WiFi Power Measurements," *IEEE Journal on Selected Areas in Communications, special issue on Location-Awareness for Radios and Networks*, volume 33, issue 7, 2015.

Conference:

- A. Muralidharan and Y. Mostofi, "Path Planning for a Connectivity Seeking Robot," *IEEE Globecom, Workshop on Wireless Networking for Unmanned Autonomous Vehicles*, 2017.
- A. Muralidharan and Y. Mostofi, "First Passage Distance to Connectivity for Mobile Robots," *American Control Conference*, 2017.
- A. Muralidharan and Y. Mostofi, "Distributed Beamforming using Mobile Robots," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2016.
- A. Muralidharan, Y. Yan and Y. Mostofi, "Binary Log-Linear Learning with Stochastic Communication Links," *IEEE Military Communications Conference*, 2015.

## Abstract

Path Planning and Communication Strategies to Enable Connectivity in Robotic Systems

by

Arjun Muralidharan

There has been considerable interest in the area of communication-aware robotics in recent years, where the sensing, communication and motion objectives of robotic systems are jointly optimized. One particular open problem in this area is that of exploiting the mobility of unmanned vehicles in order to improve or satisfy communication objectives in realistic communication environments. Progress in this field could not only affect robust networked operation of unmanned vehicles but also would improve communication systems of the future (e.g. 5G), thus contributing to both areas of robotics and communications. This mobility-enabled connectivity and communication is the main area of interest in this dissertation.

This dissertation is focused on path planning and communication strategies for robotic systems seeking to satisfy certain communication objectives in realistic communication environments experiencing path loss, shadowing and multipath fading. We consider realistic communication environments by leveraging and incorporating a probabilistic channel prediction framework that allows the robots to predict the channel quality at unvisited locations. This thesis then contributes to the area of mobility and connectivity through three main topics 1) *energy-optimal distributed beamforming*, 2) *finding the statistics of the distance traveled until connectivity*, and 3) *path planning for connectivity.* First, in *energy-optimal distributed beamforming*, we utilize the motion of a group of initially unconnected mobile robots to enable new forms of connectivity. More specifically,

we co-optimize their locations and transmission powers to cooperatively enable connectivity through distributed beamforming. We further bring a foundational theoretical understanding to robotic distributed beamforming. Next, in *finding the statistics of the distance traveled until connectivity*, we analytically characterize the probability density function of the distance traveled by an initially unconnected robot until it gets connected to a remote node as it moves along a given path. We utilize tools from the stochastic differential equation literature to develop this characterization. Finally, in *path planning for connectivity*, we actively plan the path of a mobile robot such that it finds a connected spot with a minimum expected traveled distance (i.e., energy). The scenario considered in this part is in fact a more general one, and tackles the problem of path planning on a graph to minimize the expected cost incurred until the successful completion of a task. This framework has applications beyond path planning for connectivity, in areas such as celestial body imaging, human-robot collaboration, and search scenarios. We bring a foundational understanding to this problem. We show how this problem is inherently hard to solve (NP-complete) and also propose a path planner, based on a game-theoretic framework, that provides an asymptotic optimality guarantee.

Overall, this thesis proposes novel strategies for utilizing the mobility of unmanned vehicles and enabling connectivity while considering the underlying energy constraints. We also provide a rigorous theoretical analysis of the aforementioned problems using a wide range of tools from communications theory, game theory, optimal control and time series literature. Moreover, through extensive realistic numerical studies using real channel parameters/data, we show the efficiency and performance of our proposed approaches.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

Networked robotic systems have been envisioned to carry out a wide range of tasks such as search and rescue, surveillance, exploration and sensing of the environment, agriculture, and emergency response [5–10]. Maintaining proper connectivity either within the network or between the network and a remote station is a key factor in successfully carrying out many of these tasks. Moreover, for robotic systems engaged in gathering data, establishing connectivity with a remote station is essential for the transfer of the collected information. The robots may not be able to establish connectivity to the remote station at their current location, and may need to move to establish a connection, exploiting the spatial variations of the channel quality. Thus, the mobility of unmanned vehicles can be exploited to establish the needed connectivity among themselves or to outside nodes.

Utilizing mobility for communication also has potential applications in the communication systems of the future, e.g., 5G. For instance, unmanned vehicles could be utilized to provide connectivity to areas with dynamic communication loads. As a motivating example consider the scenario of a large warehouse or farm that is sparsely populated, i.e., occupied by a few humans with communication needs. In this setting it would be more feasible to have a few unmanned vehicles, serving as mobile access points and satisfying the communication requirements of the area, rather than having a dense deployment of static access points. The unmanned vehicles can use their mobility to dynamically position themselves such that they provide the desired connectivity to the end users. Other examples of such demanding spatially-varying and time-varying communication loads are during events such as parades and sports games. Another possible application of such mobility-enabled communication in communication systems is in the use of unmanned aerial vehicles as aerial base stations [11–14]. In all these scenarios, the mobility of unmanned vehicles can be used to provide communication services. This mobility-enabled connectivity and communication is the main focus of this dissertation.

An important consideration when planning the control and communication strategies for the robots is accounting for realistic communication channels. Much of the existing literature on networked robotics assumes over-simplified models of connectivity such as disk models (where robots are connected if the distance between them is smaller than a given threshold) or path-loss only models. Such over-simplified models do not accurately capture the true nature of realistic communication channels, and the control strategies developed using them experience significant performance degradation when implemented in practice. It is a goal of this thesis to plan for the operations of the robot in *realistic communication environments* experiencing path loss, shadowing and multipath fading. Moreover, in order for an unmanned vehicle to effectively exploit it's mobility to establish connectivity, it must be able to predict the channel quality reliably at unvisited locations. We use the realistic channel prediction framework, proposed in [2, 3], to predict the channel quality at unvisited locations based on a few prior measurements collected in the environment. In this dissertation, we will utilize this prediction framework to design communication and motion strategies for robots operating in such realistic, harsh and complex communication environments.

Another key consideration when dealing with unmanned vehicles is their energy consumption. These vehicles typically have a limited energy budget and thus energy efficiency can be critical. *Energy efficiency* is thus another important theme that runs through this dissertation.

Overall, the focus of this dissertation is on the following question: *how best to utilize the mobility of unmanned vehicles to enable and optimize communication*? We are interested in path planning and communication strategies for robots seeking to satisfy certain communication objectives, while considering 1) realistic fading communication environments, and 2) energy efficiency. We will explore this area in this dissertation through three main topics:

- *Energy optimal beamforming using unmanned vehicles*: A group of unmanned vehicles use their mobility to enable new forms of connectivity. More specifically, the robots cooperatively enable connectivity to a remote station through distributed beamforming, while being energy efficient.
- *Statistics of the distance traveled until connectivity*: We characterize the probability density function of the distance that will be traveled by an initially unconnected robot until it gets connected, as it moves along a path. This characterization allows the robot to assess its distance to connectivity and plan accordingly.
- *Path planning for minimizing the expected cost until success/connectivity*: We tackle the general problem of path planning on a graph where each edge has a cost associated with it, and each node has a probability of success (with respect to some task) associated with it. We plan a path on the graph that minimizes the expected cost until success. An immediate application of this, relevant to the main focus of this thesis, is for planning the path of an unmanned vehicle to minimize the expected traveled distance until it gets connected to the remote station.

Our work not only provides an analytical characterization of the impact of mobility on connectivity, but also describes how to utilize this mobility to enable connectivity as well as new communication paradigms. Each topic will be discussed in detail in a separate chapter.

We first review the current state-of-the-art in the field of communication-aware robotics and mobility-enabled communication. We will discuss the literature relevant to each of our main topics in more detail later.

**Literature survey:**

Communication-aware robotics, where a group of robots carry out their task while being aware of the communication requirements, has been the focus of considerable research interest in recent years. There has been work on communication-aware robotics in diverse application areas such as coverage [15–17], surveillance [18–20], flocking and rendezvous [21, 22], sensing [23, 24], target tracking [25], and task allocation [26, 27]. For instance, [15] proposes a distributed control scheme to maximize the coverage area of a robotic network while ensuring reliable communication between the robots. In [18], a group of mobile robots search an area while periodically guaranteeing connectivity of the network within a fixed interval, and in [21], traditional control problems of rendezvous, flocking and formation are tackled while being aware of the connectivity of the network. In [27], a distributed task allocation framework is proposed that can also handle dynamic communication network topologies. However, the work in [15, 18, 21, 27] considers simplistic models of connectivity and communication. In contrast, in [16, 19, 20, 25, 28, 29], a realistic communication environment is considered. In [16], a number of mobile robots periodically cover a set of points of interests, and in [19], a team of mobile robots search an area to detect an unknown number of targets. In both scenarios, the networked operation of the robots are carried out in realistic communication environments. In [28, 29], a robot co-optimizes its motion speed and communication transmission rate to a remote station as it moves along a trajectory.

On the mobility-enabled communication side, there has been work on using the mobility of unmanned vehicles for point-to-point communication [30–32], relaying and routing [33–37], beamforming [38–40], data gathering and muling [41–45], and UAV-aided communication coverage [11–14]. For instance, in [30], robots optimize their motion to spend more time at locations where the channel is good for improved communication performance, and in [31], an energy-efficient trajectory is designed for the reliable transfer of information to a remote station. In [33], robots utilize mobility to maintain an optimal communication chain between a source and a destination node assuming a simplified path-loss only model, while in [34], a number of nodes utilize their mobility to form a communication relay network using a more realistic communication model. In [42–44] mobile robots act as data mules collecting data from sources and transmitting them to a remote station. The robots utilize their mobility to enable data gathering and the corresponding optimized trajectories are generated. In [13], unmanned aerial vehicles (UAVS) are utilized as aerial base stations, and the problem of their optimal placement is considered.

We next discuss each of the three aforementioned topics of this dissertation, where we place our work in context by reviewing the relevant literature, and summarize our contributions.

## 1.1 Energy Optimal Distributed Beamforming using Unmanned Vehicles

In Chapter 3, we consider a scenario where a team of unmanned vehicles utilize their mobility to cooperatively enable connectivity to a remote station. We are interested in an *energy-aware distributed transmit beamforming using unmanned vehicles*. Consider the case where a number of unmanned vehicles need to establish connectivity to a remote node but they are in a poorly connected area. Can they cooperatively generate a strong link? More specifically, the unmanned vehicles are tasked with distributed transmit beamforming (virtual antenna array placement and design), in order to cooperatively transmit information to a remote station in realistic communication environments while minimizing the total energy consumption including both motion and communication energy costs. In this manner, the unmanned vehicles can achieve a strongly-connected link with a minimum energy consumption. Fig. 1.1 shows an illustration of distributed robotic transmit beamforming. We are then interested in characterizing the optimal motion and communication strategies of the robots, including the optimization of the transmit power and robot paths. Fig. 1.2 shows an example of such a scenario.

Traditional distributed transmit beamforming is a cooperative communication strategy where a number of fixed transmitters cooperate to emulate a virtual centralized antenna array. For instance, consider the case where a node needs to transmit information to a remote station. If the corresponding link quality is not good, successful communication may not be possible. Instead, a number of transmitters can perform transmit beamforming, which means co-phasing and properly weighing their transmitted signals to communicate the same message while maintaining the same total communication power. In this manner, transmit beamforming creates an equivalent strong link to the receiving node. Transmit beamforming was originally proposed in the context of multiple co-located antennas for improving transmission quality of communication systems. More recently, it has been extensively studied in the context of fixed nodes that are spatially distributed over a given area [46, 47]. Then, the nodes align their transmission phases such that the wireless signals merge constructively at the remote station, thus providing dramatic gains in the signal to noise ratio (SNR). Using unmanned vehicles creates new possibilities for distributed transmit beamforming by enabling the transmitters to position themselves in better locations for beamforming, thus improving the

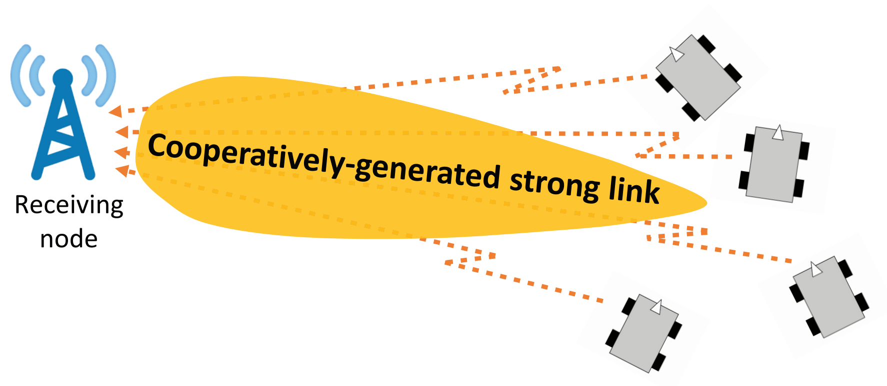


Figure 1.1: Distributed robotic transmit beamforming. The robots can cooperatively generate a strong communication link by optimizing their locations.

overall performance significantly. However, several challenges for motion and communication co-planning need to be addressed before realizing this vision, which is the main motivation for Chapter 3.

As compared to the existing literature on distributed beamforming, most work are not concerned with unmanned vehicles and the resulting challenges in terms of path planning and motion energy. In [38–40], where robots act as collaborative relay beamformers, motion energy-related issues are not considered, resulting in a different problem formulation. Moreover, there is no channel learning and prediction. Finally, the motion of the relays is myopic, and they can get stuck in a local minimum. In [48], the robots self organize to form a distributed pattern for beamforming. However, an unrealistic path loss-only model is considered and the operation is not energy-aware.

**Contributions**: In Chapter 3, we use the mobility of unmanned vehicles to create new possibilities for traditional distributed transmit beamforming by enabling the transmitters to position themselves in better locations for beamforming, thus improving the overall performance significantly. We are interested in the energy-aware (both motion and communication energy) co-optimization of robotic paths and transmission powers for cooperative transmit beamforming under a reception quality requirement. Chapter 3

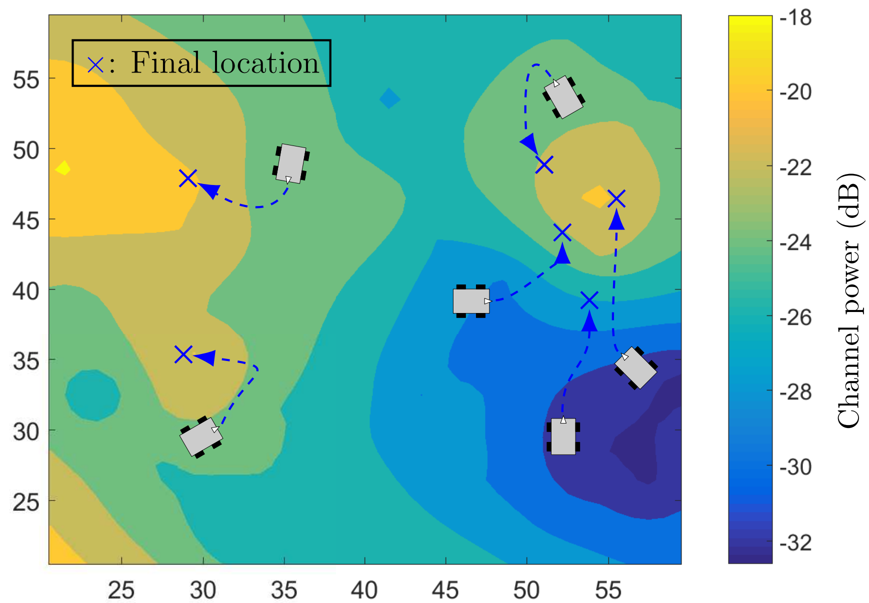


Figure 1.2: Distributed robotic beamforming – The robots move to locations (marked by 'x') better for satisfying the cooperative connectivity requirement, while minimizing the total energy consumption (both motion and communication).

is different from the existing work on cooperative beamforming in that it deals with the *co-optimization of motion and communication* strategies, while considering 1) the *total energy consumption*, 2) *channel learning and prediction* in realistic communication environments, and 3) the *coupled decision making* that arises when dealing with multi-agent systems.

We first consider the case where the channel is known. For this case, we propose an efficient approach for getting arbitrarily close to the optimum solution, which involves solving a series of multiple-choice knapsack problems. We then extend our analysis and methodology to the case where the channel is not known. The robots then probabilistically predict the channel at unvisited locations and integrate it with path planning and decision making for energy-aware distributed transmit beamforming. Finally, we extensively confirm our proposed approach with several simulation results with real channel parameters. Our results highlight the underlying trends of the optimum strategy and indicate a considerable energy saving.

## 1.2 Statistics of the Distance Traveled until Connectivity for Unmanned Vehicles

In Chapter 4, we consider a scenario where a robot needs to establish connectivity with a remote operator or another robot, as it moves along a given path. We are interested in answering the following question: what is the distance traveled by the robot along the path before it finds a connected spot? More specifically, we are interested in characterizing the statistics of the distance traveled along the path before it gets connected, in realistic channel environments experiencing path loss, shadowing and multipath effects. We are in particular interested in a predictive characterization that the robot can implement on the field. Figure 1.3 shows an example of such a scenario.

Such a characterization would not only bring a foundational analytical understanding to the distance traveled until connectivity but can also significantly affect networked robotic operations. For instance, the analysis can help with the operation of the robotic system in the field, as well as in the design of robotic paths. The five underlying parameters of the channel can typically be measured based on very few channel samples in the environment, either online or from prior operations. Thus, the robot can assess the statistics of its distance to connectivity, using our derivations, when on a field mission. Furthermore, the derivations can be used to explicitly co-optimize and design robotic sensing and path planning. We finally emphasize that the derivations of the chapter are applicable to both cases of trying to establish communication with remote operators as well as trying to establish communication with another robot.

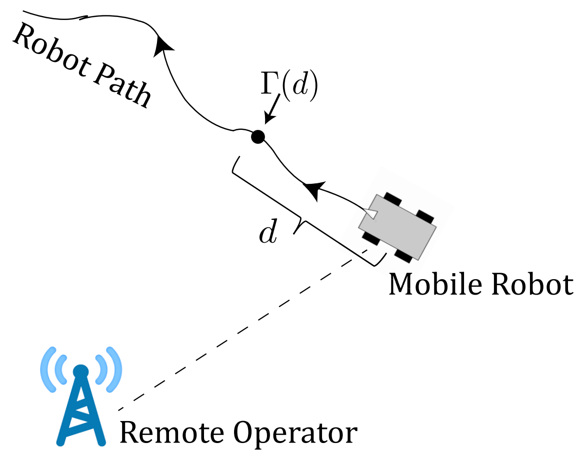


Figure 1.3: An example of the considered scenario for a general path. In Chapter 4, we are interested in characterizing the statistics of the distance traveled until connectivity.

There has been some recent research interest in the area of connectivity in robotic systems. For instance, in [21], the connectivity of a network is maximized using a graph-theoretic analysis while in [34], connectivity is optimized using a more realistic channel model. There has also been work on path planning to enable connectivity [31, 34, 35, 49, 50] as well as on communication-aware sensing [44]. In [50], comm-aware trajectory planning with connectivity repair is carried out, and in [31], an energy efficient trajectory is designed for a UAV communicating with a remote station.

**Contributions:** We mathematically characterize the statistics of the distance traveled until connectivity, for a robot traveling on a given path, as a function of the underlying channel parameters of the environment, such as shadowing, path loss, and multipath fading parameters. To the best of the our knowledge, such an analysis of the distance traveled until connectivity, is lacking in the literature. We first develop an exact characterization of the statistics of the distance traveled until connectivity for straight-line paths. We utilize tools from the stochastic differential equation literature to characterize the statistics of the distance traveled until connectivity while ignoring the multipath component (which could be of interest when the robot looks for an area of good connectivity as opposed to a single spot, or when multipath is negligible). For the case when we include multipath, we develop a recursive characterization of the PDF of the distance traveled. We then mathematically characterize a more general space of non-straight paths for which the analysis holds, based on properties of the path such as its curvature. Our characterizations significantly reduce the computational complexity of computing this PDF when compared to a direct computation via a high dimensional integration. Finally, we confirm our theoretical analysis using extensive numerical results with real channel parameters from downtown San Francisco.

## 1.3 Path Planning for Minimizing the Expected Cost until Success/Connectivity for Unmanned Vehicles

In Chapter 5, we consider the problem of planning the path of an unmanned vehicle seeking to get connected to a remote station. The unmanned vehicle is initially unconnected and must incur motion energy to find a connected spot. We are then interested in *energy-aware path planning* of a robot, to ensure reaching a *guaranteed connected spot* in a realistic channel environment that experiences path loss, shadowing and multipath fading. More precisely, we consider the problem of planning the path of a robot in order to find a connected spot while *minimizing the expected traveled distance*. We note that in a realistic channel environment, the robot's knowledge of the connectivity at any location is stochastic. Hence, the traveled distance until connectivity is not known a priori, and is a random variable. Our objective is then to find a path that minimizes the expected traveled distance until connectivity.

We pose this problem as a graph-theoretic problem where each edge has a cost associated with it and each node has a Boolean value of success or failure with a given probability. Success in this setting refers to the state of being connected to a remote station. The goal is then to *plan a path* through the graph that would *minimize the expected cost* until success.

Although our main motivation for posing this graph-theoretic problem is to plan the path for a connectivity seeking robot, the underlying general path planning problem on a graph has several other applications. For instance, consider the scenario of a rover on mars looking for an object of interest, for instance a sample of water, for scientific studies. Based on prior information, it has an estimate of the likelihood of finding such an object at any particular location. The goal in such a scenario would be to locate one such object with a minimum expected cost. Note that there may be multiple such objects in the environment, and that we only care about the expected cost until the first such object is found. We can then pose this in our graph-theoretic framework where there is a probability of success in finding an object associated with each node. The goal is then to plan a path through the graph that would minimize the expected cost until an object of interest is successfully found. Several other problems of interest also fall into this formulation. For instance, consider the scenario of astronomers searching for a habitable exoplanet. Researchers have characterized the probability of finding exoplanets in different parts of space [51]. However, repositioning satellites to target and image different celestial objects is costly and consumes fuel. Thus, a problem of interest in this context, is to find an exoplanet while minimizing the expected fuel consumption, based on the prior probabilities. Finally, consider a human-robot collaboration scenario, where an office robot needs help from a human, for instance in operating an elevator [52]. If the robot has an estimate of different people's willingness to help, perhaps from past observations, it can then plan its trajectory to minimize its energy consumption until it

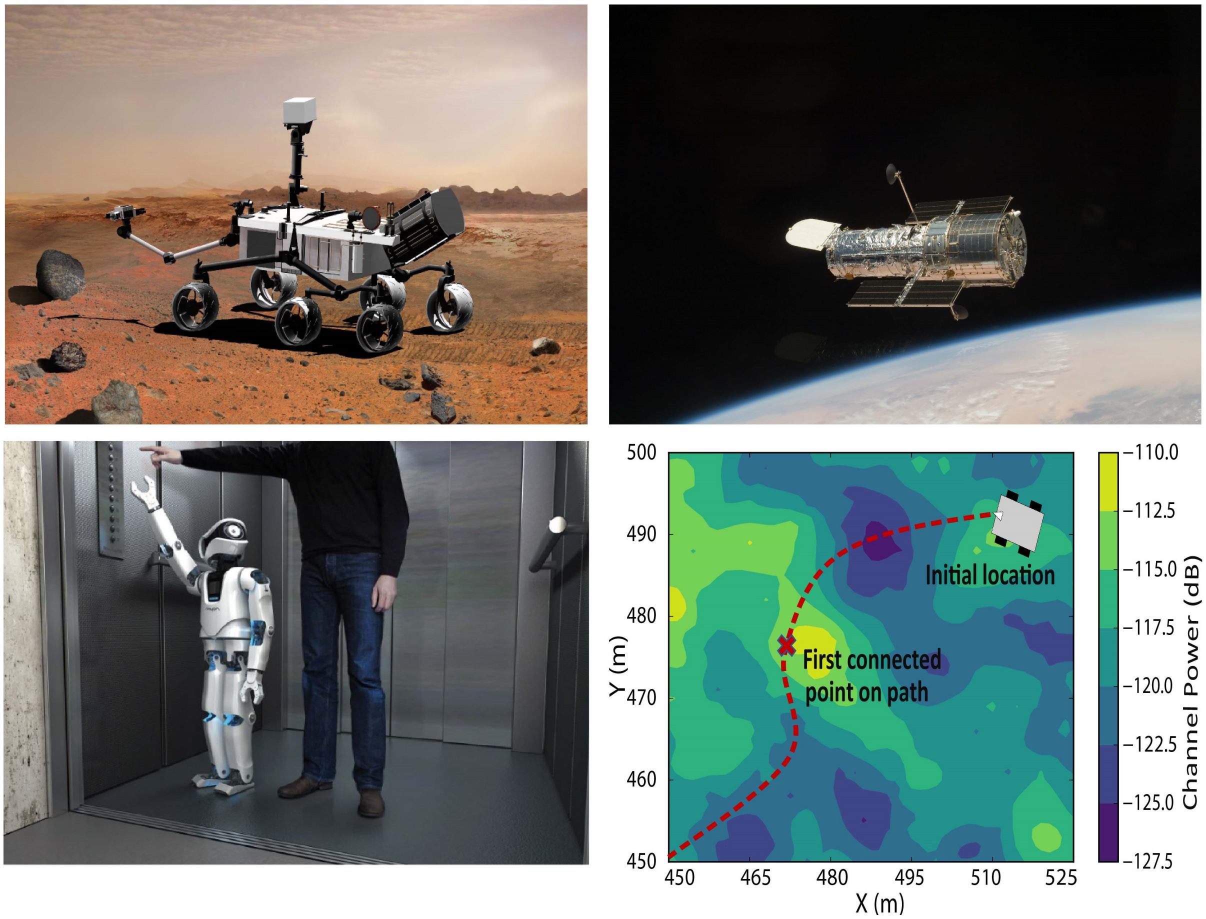


Figure 1.4: Possible applications of the problem of interest: (top left) path planning for a rover, (top right) imaging of celestial objects, (bottom left) human-robot collaboration and (bottom right) path planning to find a connected spot. Image credit:(top left) and (top right) NASA, (bottom left) Noto: `http://www.noto.design/`.

finds help. Fig. 1.4 showcases a sample of these possible applications.

With regard to planning for connectivity in robotic networks, there has been some work in recent years. In [53], an energy-aware trajectory is designed for a robotic network in order to enable within-network connectivity. In [54], a controller that ensures persistent intermittent connectivity is designed for a robotic network. However, oversimplified and unrealistic channel models are considered in all the above references [33, 53–55]. In [28, 34, 56, 57], a realistic channel model and probabilistic prediction framework based on [3] is utilized. In [34], locations of mobile robotic routers are optimized for enabling end-to-end connectivity. In [56], an optimal control-based framework is proposed to co-optimize motion and communication.

With regard to the general graph-theoretic problem, optimal path planning for a robot has received considerable interest in the research community, and several algorithms have been proposed in the literature to tackle such problems, e.g., A*, RRT* [58, 59]. These works are concerned with planning a path for a robot, with a minimum cost, from an initial state to a predefined goal state. However, this is different from our problem of interest in several aspects. For instance, the cost metric is additive in these works, which does not apply to our setting due to its stochastic nature. In the probabilistic traveling salesman problem [60] and the probabilistic vehicle routing problem [61], each node is associated with a prior probability of having a demand to be serviced, and the objective is to plan an a priori ordering of the nodes which minimizes the expected length of the tour. A node is visited in a particular realization only if there is a demand to be serviced. Thus, each realization has a different tour associated with it, and the expectation is computed over these tours, which is a fundamentally different problem than ours. Another area of active research is in path planning strategies for a robot searching for a target [62–64]. For instance, in [62], a mobile robot is tasked with locating a stationary target in minimum expected time. In [63], there are multiple mobile robots and the objective is to find a moving target efficiently. In general, these papers belong to a body of work known as optimal search theory where the objective is to find a *single* hidden target based on an initial probability estimate, where the probabilities over the graph sum up to one [64, 65]. The minimum latency problem [66] is another problem related to search where the objective is to design a tour that minimizes the average wait time until a node is visited. In contrast, our setting is fundamentally different, and involves an *unknown* number of targets where each node has a probability of containing a target ranging from 0 to 1. Moreover, the objective is to plan a path that minimizes the expected cost to the *first target* found. This results in a different analysis and we utilize a different set of tools to tackle this problem. Another related problem is that of satisficing search in the artificial intelligence literature which deals with planning a sequence of nodes to be searched until the first satisfactory solution is found, which could be the proof of a theorem or a task to be solved [67]. The objective in this setting is to minimize the expected cost until the first instance of success. However, in this setting there is no cost associated with switching the search from one node to another. Our problem also has interesting analogies (while still different) to the discounted-reward traveling salesman problem [68] and the stochastic shortest path with recourse problem [69] which are discussed in Appendix C.4 and C.5 respectively.

**Contributions:** In Chapter 5, we consider a general path planning problem of a robot on a graph with edge costs, and where each node has a Boolean value of success or failure (with respect to some task) with a given probability. The objective is to plan a path for the robot on the graph that minimizes the expected cost until success. To the best of our knowledge, the problem considered in Chapter 5 has not been explored before. We bring a foundational understanding to this problem. We start by showing how this problem can be posed as an infinite horizon Markov Decision Process (MDP) and solved optimally, but with an exponential space complexity. We then formally prove its NP-hardness. To address the space complexity, we then propose an asymptotically $\epsilon$-suboptimal (i.e., within $\epsilon$ of the optimal solution value) path planner for this problem, using a game-theoretic framework. We further show how it is possible to solve this problem very quickly by proposing two sub-optimal but non-myopic approaches. Our proposed approaches provide a variety of tools that can be suitable for applications with different needs. To show the performance of our framework, we do extensive simulations for the scenario of a rover on Mars searching for an object for scientific studies. Our numerical results show a considerable performance improvement over existing state-of-the-art approaches.

We also implement our proposed approaches for the scenario of an unmanned vehicle looking for a connected spot to a remote station. Our extensive simulations, with real channel parameters from downtown San Francisco, confirm that our approaches can significantly reduce the expected traveled distance until connectivity, thus minimizing the total energy consumption.

# Chapter 2

# Preliminaries

In this chapter we will review the wireless channel model, as well as the realistic channel prediction framework proposed in [2,3]. The contents of this chapter are used throughout the rest of this dissertation.

## 2.1 Probabilistic Channel Modeling [1]

A communication channel is well modeled as a multi-scale random process with three major dynamics: path loss, shadowing and multipath fading [1]. Path loss is the slowest dynamic and is the result of a distant dependent signal attenuation. The shadowing component or large scale fading is a faster dynamic, and represents fluctuations of the channel power due to the blocking impact of large objects such as buildings and trees. The multipath component or small-scale fading represents fluctuations in the channel power in the order of a wavelength due to multiple reflecting objects in the environment. Fig. 2.1 shows the received signal power collected across an indoor route, and marked are the three underlying dynamics.

Let $\Gamma_{\text{lin}}(q_1)$ represent the received channel power from a transmitter at location $q_1 \in \mathcal{W}$ ($\mathcal{W} \subseteq \mathbb{R}^2$ is the workspace) to the remote station located at $q_b$. The received channel

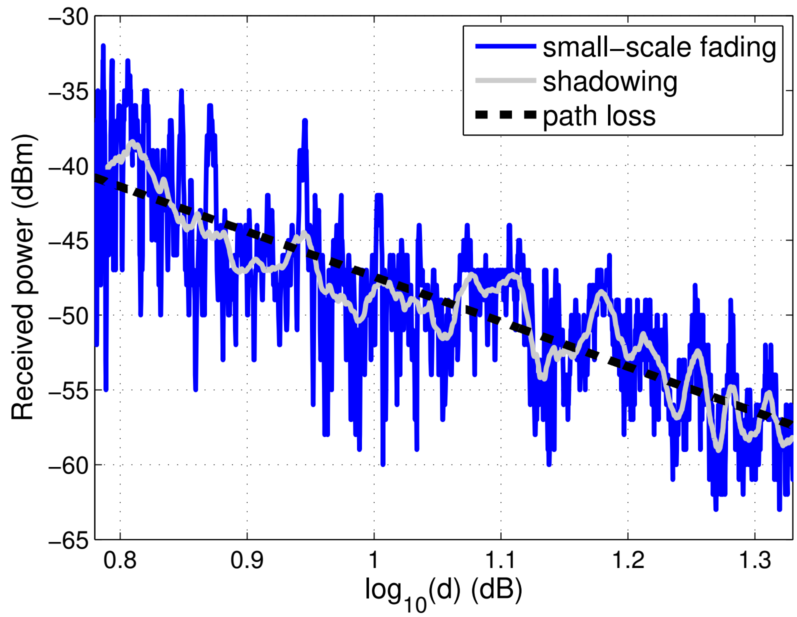


Figure 2.1: Underlying dynamics of the received power across an indoor route [4].

power in dB, $\Gamma(q_1) = 10\log_{10}\Gamma_{\text{lin}}(q_1)$, can be expressed as

$$\Gamma(q_1) = \gamma_{\text{PL}}(q_1) + \Gamma_{\text{SH}}(q_1) + \Gamma_{\text{MP}}(q_1)$$

where $\gamma_{\text{PL}}(q_1) = K_{\text{dB}} - 10n_{\text{PL}}\log_{10}\|q_1 - q_b\|$ is the distance-dependent path loss with $n_{\text{PL}}$ representing the path loss exponent, and $\Gamma_{\text{SH}}(q_1)$ and $\Gamma_{\text{MP}}(q_1)$ are random variables denoting the impact of shadowing and multipath respectively. $\Gamma_{\text{SH}}(q_1)$ is best modeled as a Gaussian random variable with an exponential spatial correlation, i.e., $\mathbb{E}\{\Gamma_{\text{SH}}(q_1)\Gamma_{\text{SH}}(q_2)\} = \sigma^2_{\text{SH}}e^{-\|q_1-q_2\|/\beta_{\text{SH}}}$ where $\sigma^2_{\text{SH}}$ is the shadowing power and $\beta_{\text{SH}}$ is the decorrelation distance. As for multipath, a number of distributions such as Nakagami, Rician and lognormal have been found to be a good fit (in the linear domain) [70, 71].

## 2.2 Realistic Channel Prediction [2, 3]

We next show how a robot can predict the channel quality at unvisited locations based on a small number of a priori-gathered measurements. Let $\vartheta = [K_{\text{dB}}\ n_{\text{PL}}]^{\text{T}}$ denote the vector of path loss parameters. Let $\Gamma_q$ represent the vector of $m$ a priori-gathered received channel power measurements (in dB) from the same environment, and $q = [q_1 \cdots q_m]^{\text{T}}$ denote the vector of the corresponding positions.

**Lemma 2.1 (See [3] for proof)** *A Gaussian random vector,* $\Gamma(r) = [\Gamma(r_1) \cdots \Gamma(r_k)]^{\text{T}} \sim \mathcal{N}\left(\overline{\Gamma}(r), \Sigma(r)\right)$ *can best characterize the vector of channel power (in the dB domain) when transmitting from unvisited locations* $r = [r_1 \cdots r_k]^{\text{T}}$*, with the mean and covariance matrix given by*

$$\begin{aligned}\overline{\Gamma}(r) &= \mathbb{E}\left\{\Gamma(r) \mid \Gamma_q, \hat{\vartheta}, \hat{\beta}_{SH}, \hat{\sigma}_{SH}, \hat{\sigma}_{MP}\right\} \\ &= G_r\hat{\vartheta} + \Xi_{r,q}\left(\Xi_q + \hat{\sigma}^2_{MP} I_m\right)^{-1}\left(\Gamma_q - G_q\hat{\vartheta}\right)\end{aligned}$$

*and*

$$\begin{aligned}\Sigma(r) &= \mathbb{E}\Big\{\left(\Gamma(r) - \overline{\Gamma}(r)\right)\left(\Gamma(r) - \overline{\Gamma}(r)\right)^{\text{T}} \mid \Gamma_q, \hat{\vartheta}, \hat{\beta}_{SH}, \hat{\sigma}_{SH}, \hat{\sigma}_{MP}\Big\} \\ &= \Xi_r + \hat{\sigma}^2_{MP} I_k - \Xi_{r,q}\left(\Xi_q + \hat{\sigma}^2_{MP} I_m\right)^{-1} \Xi^{\text{T}}_{r,q},\end{aligned}$$

*respectively, where* $G_r = [\mathbf{1}_k \quad -L_r]$*,* $G_q = [\mathbf{1}_m \quad -L_q]$*,* $\mathbf{1}_m$ *(*$\mathbf{1}_k$*) represents the m-dimensional (k - dimensional) vector of all ones,* $I_m$ *(*$I_k$*) represents the m-dimensional (k - dimensional) identity matrix,* $L_q = \left[10\log_{10}(\|q_1 - q_b\|)\ \cdots\ 10\log_{10}(\|q_m - q_b\|)\right]^{\text{T}}$*,* $L_r = \left[10\log_{10}(\|r_1 - q_b\|)\ \cdots\ 10\log_{10}(\|r_k - q_b\|)\right]^{\text{T}}$ *and* $q_b$ *is the position of the remote* *station. Furthermore,* $\Xi_q$*,* $\Xi_r$ *and* $\Xi_{r,q}$ *denote matrices with entries*

$$\begin{aligned}\left[\Xi_q\right]_{i_1,i_2} &= \hat{\sigma}_{SH}^2 e^{-\|q_{i_1}-q_{i_2}\|/\hat{\beta}_{SH}},\\ \left[\Xi_r\right]_{j_1,j_2} &= \hat{\sigma}_{SH}^2 e^{-\|r_{j_1}-r_{j_2}\|/\hat{\beta}_{SH}},\\ \left[\Xi_{r,q}\right]_{j_1,i_1} &= \hat{\sigma}_{SH}^2\, e^{-\|r_{j_1}-q_{i_1}\|/\hat{\beta}_{SH}},\end{aligned}$$

*where* $i_1, i_2 \in \{1, \cdots, m\}$*, and* $j_1, j_2 \in \{1, \cdots, k\}$*. Moreover,* $\vartheta$*,* $\beta_{SH}$*,* $\sigma_{SH}^2$ *and* $\sigma_{MP}^2$ *denote the path loss parameters, the decorrelation distance of shadowing, the power of shadowing (in dB) and the power of multipath (in dB) respectively. The* $\hat{}$ *symbol denotes the estimate of the corresponding parameter.*

The underlying parameters $\vartheta$, $\beta_{\mathrm{SH}}$, $\sigma_{\mathrm{SH}}$, and $\sigma_{\mathrm{MP}}$, can be estimated based on very few a priori measurements as well. See [3] for more details on the estimation of the underlying parameters and the performance of this framework with real data and in different environments.

# Chapter 3

# Energy Optimal Distributed Beamforming using Unmanned Vehicles

In this chapter, we show how a team of unmanned vehicles utilize their mobility to cooperatively enable connectivity to a remote station. We consider a team of unmanned vehicles that need to communicate to a remote node but are located in a poorly-connected area. The nodes are then interested in cooperatively generating a strong link to the remote node by utilizing their mobility. More specifically, they are tasked with distributed transmit beamforming (virtual antenna array placement and design), in order to cooperatively transmit information to a remote station in realistic communication environments. We are interested in an energy-aware operation, where the unmanned vehicles cooperatively transmit information, while minimizing the total energy consumption including both motion and communication energy costs. We are then interested in characterizing the optimal motion and communication strategies of the robots, including the optimization of the transmit power and locations of the robots. Fig. 1.2 shows an example of such a scenario, where the unmanned vehicles use their mobility to move to better locations

Parts of this chapter have appeared in our journal submission [72], ©[2017] IEEE.

for distributed beamforming.

The chapter is organized as follows. In Section 3.1, we introduce the motion and communication energy cost models and briefly review distributed transmit beamforming. In Section 3.2, we consider the scenario where the robots do not satisfy the reception quality requirement from their initial positions when employing distributed transmit beamforming. We are then interested in determining the optimum paths of the robots such that the reception requirement is met with minimum motion energy cost. As a starting point, we start with the case where the robots know the channel strength over the workspace. We show how to pose this problem as a multiple-choice knapsack problem, which can be solved optimally. We then extend our analysis to the scenario where the robots no longer have the knowledge of the channel strength. Instead, they use the channel prediction framework of Chapter 2 to probabilistically predict the channel quality at unvisited locations based on a small number of a priori channel samples. In Section 3.3, we then incorporate communication energy cost into our framework, i.e, we minimize the total energy cost (both motion and communication) while satisfying the reception quality requirement. We are then interested in the co-optimization of robotic paths and transmission powers for cooperative transmit beamforming. For the perfect channel knowledge case, we show how to obtain an $\epsilon$-suboptimal solution by solving a series of multiple-choice knapsack problems. We then extend our analysis to the case where the channel is not known, by utilizing probabilistic channel prediction. In Section 3.4, we confirm our proposed approach with extensive simulation results using channel parameters obtained from real measurements [73]. Our results indicate a considerable energy saving.

## 3.1 Problem Setup

In this section we first introduce our energy consumption models for both motion and communication. We then review distributed transmit beamforming and the corresponding power gains that it provides. Finally, we briefly summarize probabilistic modeling and prediction of wireless channels.

### 3.1.1 Motion Energy Model

In this chapter, we adopt a model where the motion energy consumption is proportional to the distance traveled, similar to the one adopted in [55,74]. Thus, motion energy $= K_M d$, where $d$ is the distance traveled by the robot and $K_M$ is a constant that depends on the environment (e.g., friction coefficient, terrain) and the mass of the vehicle. This model is a good match for wheeled robots (see [74] for discussion).

### 3.1.2 Communication Energy Model

We consider a generic model of communication rate of the form $\eta_1 W \log_2\left(1 + \eta_2 \frac{P_R}{N_0}\right)$, where $\eta_1, \eta_2 \leq 1$ are constants, $W$ is the available bandwidth, $P_R$ is the received power and $N_0$ is the noise power.[1] For capacity approaching codes (such as turbo codes and LDPC), the constants for a binary symmetric channel correspond to $\eta_1 = 1 - \varepsilon$ and $\eta_2 = 1$ where $\varepsilon$ is the multiplicative gap to capacity [75]. For an uncoded MQAM modulation scheme with a target bit error rate of $\text{BER}_{\text{th}}$, we obtain $\eta_1 = 1$ and $\eta_2 = 1.5/\ln(5\text{BER}_{\text{th}})$ [1]. The communication energy incurred in transmitting $n_{\text{bits}}$ bits of data can then be expressed

[1]Note that the communication rate is adaptive as it is a function of the received power.

as

$$\text{Communication Energy} = \underbrace{\frac{n_{\text{bits}}}{\eta_1 W \log_2\left(1+\eta_2 \frac{P_R}{N_0}\right)}}_{\text{time to transmit } n_{\text{bits}} \text{ bits}} P_0, \tag{3.1}$$

where $P_0$ is the transmit power.

### 3.1.3 Distributed Transmit Beamforming

Distributed transmit beamforming is a form of cooperative communication where several nodes that are distributed in a given space emulate a centralized antenna array [46]. The nodes simultaneously transmit the same message such that the signals combine constructively at the remote station. Channel state information (CSI), i.e., information about the channel, is required at the transmitters for the implementation of distributed transmit beamforming.

Consider $N$ robots in an environment. Let $h_i = \alpha_i e^{j\angle h_i}$ denote the complex baseband channel from robot $i$ to the remote station with $\alpha_i$ denoting the channel amplitude. Ideally, node $i$ transmits $\omega_i s_b(t)$ where $\omega_i = \rho_i e^{-j\angle h_i}$ is the complex beamforming weight and $s_b(t)$ is the complex baseband signal to be transmitted. As can be seen, setting $\angle\omega_i = -\angle h_i$ is the crucial step in obtaining a constructive interference and thus beamforming gains. The received signal is then $(\sum_{i=1}^{N} h_i w_i)s_b(t) = \sum_{i=1}^{N}(\alpha_i\rho_i)s_b(t)$ resulting in the received SNR of $\frac{P_0(\sum_{i=1}^{N}\alpha_i\rho_i)^2}{N_0}$ where the transmit power of robot $i$ is $\rho_i^2 P_0$. Constraining $\rho_i \leq 1$ imposes a maximum power of $P_0$ on each node. We stress here the difference from the traditional centralized transmit beamforming where a total transmit power is enforced, i.e. $\sum_{i=1}^{N}\rho_i^2 \leq 1$. However, in distributed beamforming, the nodes are separated and have their own power supply. We thus impose individual power constraints instead. Note that the position of node $i$ affects $\alpha_i$, the corresponding channel amplitude, and therefore the overall received SNR. Thus, by properly designing robotic paths and transmit power ($\rho_i$), using unmanned vehicles can significantly improve distributed transmit beamforming, as we shall see in this chapter.

## 3.2 Motion Energy-Aware Cooperative Robotic Beamforming

Consider the case where the robots are distributed over the space such that the required cooperative received SNR is not satisfied. The robots thus need to move to new positions that satisfy the cooperative connectivity requirement while minimizing the motion energy consumption. We start by looking at the case of perfect channel knowledge (i.e., the robots know the uplink channel quality for transmission from any unvisited location), and show how this problem can be optimally solved by posing it as a multiple-choice knapsack problem. We then extend our analysis to the stochastic case where the nodes predict the channel based on a small number of a priori channel samples, as discussed in Section 2.2. In this section we focus on motion energy minimization, assuming a non-adaptive communication transmit power case.

### 3.2.1 Perfect Channel Knowledge

The perfect channel knowledge assumption would be a good approximation for environments where path loss is dominant and channel has a low variance around path loss. In our case, this serves as a starting point for our analysis, which will then be extended to the general case of an unknown channel. Consider $N$ robots in a workspace $\mathcal{W} \subseteq \mathbb{R}^2$. Let $d_i(r_i) = ||r_i - r_i^0||_2$ be the distance traveled by robot $i$ with $r_i^0$ and $r_i$ denoting the initial and final position respectively. Let $h(r_i) = \alpha(r_i)e^{j\angle h(r_i)}$ be the uplink channel from position $r_i$ to the remote station with $\alpha(r_i)$ denoting the channel amplitude.

Since communication cost is not penalized in this setup, the optimal thing for the nodes would be to maximize the SNR at the remote station, subject to the individual power constraints. We then set $\rho_i = 1$, which corresponds to each node transmitting at the maximum allowed power, and the complex beamforming weight as $\omega_i = e^{-j\angle h(r_i)}$, for the $i^{\text{th}}$ node. The received signal power, after beamforming, is then given by $P_R = P_0 \left(\sum_{i=1}^{N} \alpha(r_i)\right)^2$.

A Quality of Service (QoS) requirement, such as a target bit error rate, would result in a minimum required received power at the remote station, which we denote as $P_{R,\text{th}}$. We then need $P_R = P_0 \left(\sum_{i=1}^{N} \alpha(r_i)\right)^2 \geq P_{R,\text{th}}$ or equivalently $\sum_{i=1}^{N} \alpha(r_i) \geq \sqrt{\frac{P_{R,\text{th}}}{P_0}}$, which results in the following optimization problem,

$$\begin{aligned} \underset{r}{\text{minimize}} \quad & KM \sum_i d_i(r_i) \\ \text{subject to} \quad & \sum_i \alpha(r_i) \geq \alpha_{R,\text{th}} \\ & r_i \in \mathcal{N}(r_i^0),\ i = 1, \cdots, N, \end{aligned} \tag{3.2}$$

where $\alpha_{R,\text{th}} = \sqrt{\frac{P_{R,\text{th}}}{P_0}}$, $\mathcal{N}(r_i^0) \subseteq \mathcal{W}$ is the neighborhood around $r_i^0$ that the $i^{\text{th}}$ node is constrained to stay in, and $r = [r_1 \cdots r_N]^{\text{T}}$ are the final positions of the robots.

### Optimal Solution

We next show how to repose the optimization problem of (3.2) as a multiple-choice knapsack problem, which is a well studied problem in the computer science literature and can be solved optimally [76]. We first discretize our workspace $\mathcal{W}$ into $M$ cells with centers $\bar{r}_j \in \mathcal{W}$, for $j \in \{1, \cdots, M\}$. The motion cost is then given by $J_{\text{MEMP}}(\{z_{ij}\}) =$

$K_M \sum_{i=1}^{N} \sum_{j\in\mathcal{N}_i} d_i(\overline{r}_j) z_{ij}$ and the optimization problem of (3.2) can be reformulated as

$$\begin{aligned} \underset{\{z_{ij}\}}{\text{minimize}} \quad & J_{\text{MEMP}}(\{z_{ij}\}) \\ \text{subject to} \quad & \sum_i \sum_{j\in\mathcal{N}_i} \alpha(\overline{r}_j) z_{ij} \geq \alpha_{R,\text{th}} \\ & \sum_{j\in\mathcal{N}_i} z_{ij} = 1,\ z_{ij} \in \{0,1\},\ \forall\, j \in \mathcal{N}_i,\ \forall\, i, \end{aligned} \tag{3.3}$$

where $d_i(\overline{r}_j)$ is the distance to cell $j$ for robot $i$, $\alpha(\overline{r}_j)$ is the channel amplitude from cell $j$ to the remote station and $\mathcal{N}_i \subseteq \{1, \cdots, M\}$ is the set of cells present in $\mathcal{N}(r_i^0)$. A value of $z_{ij} = 1$ implies that robot $i$ chooses to move to cell $j$. We refer to this problem as the Motion Energy Minimization Problem (MEMP), with the optimal value of $J_{\text{MEMP}}^{\text{OPT}}$.

**Lemma 3.1** *MEMP of (3.3) can be posed as a multiple-choice knapsack problem (MCKP).*

*Proof:* Define $\{\pi_{ij}\}$ and $\{\varpi_j\}$ as

$$\begin{aligned} \pi_{ij} &= \max_{k\in\mathcal{N}_i} d_i(\overline{r}_k) - d_i(\overline{r}_j) \\ \varpi_j &= \max_k \alpha(\overline{r}_k) - \alpha(\overline{r}_j). \end{aligned}$$

We have

$$\begin{aligned} \sum_{i=1}^{N} \sum_{j\in\mathcal{N}_i} \pi_{ij} z_{ij} &= \sum_{i=1}^{N} \max_{k\in\mathcal{N}_i} d_i(\overline{r}_k) \sum_{j\in\mathcal{N}_i} z_{ij} - \sum_{i=1}^{N} \sum_{j\in\mathcal{N}_i} d_i(\overline{r}_j) z_{ij} \\ &= \sum_{i=1}^{N} \max_{k\in\mathcal{N}_i} d_i(\overline{r}_k) - \sum_{i=1}^{N} \sum_{j\in\mathcal{N}_i} d_i(\overline{r}_j) z_{ij}, \end{aligned}$$

where the second equality follows for any feasible solution since $\sum_{j=1}^{M} z_{ij} = 1$. Similarly, $\sum_{i=1}^{N} \sum_{j\in\mathcal{N}_i} \varpi_j z_{ij} = N \max_k \alpha(\overline{r}_k) - \sum_i \sum_{j\in\mathcal{N}_i} \alpha(\overline{r}_j) z_{ij}$. MEMP of (3.3) can then be posed as

$$\begin{aligned} \underset{\{z_{ij}\}}{\text{maximize}} \quad & \sum_i \sum_{j\in\mathcal{N}_i} \pi_{ij} z_{ij} \\ \text{subject to} \quad & \sum_i \sum_{j\in\mathcal{N}_i} \varpi_j z_{ij} \leq c_{\mathrm{KP}} \\ & \sum_{j\in\mathcal{N}_i} z_{ij} = 1,\ z_{ij} \in \{0,1\},\ \forall\, j \in \mathcal{N}_i,\ \forall\, i, \end{aligned} \tag{3.4}$$

where $c_{\mathrm{KP}} = N \max_k \alpha(\overline{r}_k) - \alpha_{R,\mathrm{th}}$. ■

Equation (3.4) is the standard form of the multiple-choice knapsack problem (MCKP). Although MCKP is NP-hard, the true solution can be efficiently found for several cases that arise in practice [76]. In this chapter, we thus utilize the minimal algorithm developed by Pisinger [76] to optimally solve the resulting MCKP.

**Remark 3.1** *Let $J_{MCKP}^{OPT}$ denote the optimal value of the objective function of (3.4). The optimal values of the two formulations (3.3) and (3.4) are then related as follows:*

$$J_{MEMP}^{OPT} = K_M\Big(\sum_i \max_{k\in\mathcal{N}_i} d_i(\overline{r}_k) - J_{MCKP}^{OPT}\Big).$$

### 3.2.2 Probabilistic Channel Prediction

In realistic scenarios, the uplink channel values in transmission from unvisited locations may not be known to the robots a priori. We next consider this realistic case. The robots then utilize the stochastic prediction approach of Section 2.2 to predict the channel strength when transmitting from an unvisited location, using a small number of a priori channel samples in the same environment.[2] Optimization of path planning for

[2] To predict the channel in transmission from any unvisited location to the remote station, the robots only need a small number of a priori channel measurements, $\Gamma_q$, from which they can estimate the channel parameters ($\hat{\vartheta}$, $\hat{\beta}_{\mathrm{SH}}$, $\hat{\sigma}_{\mathrm{SH}}$, $\hat{\sigma}_{\mathrm{MP}}$) [3]. If time division duplexing (TDD) is not utilized, the remote station can broadcast $\Gamma_q$ to the unmanned vehicles during the operation, using a feedback channel. Alternatively, if we have TDD, then the robots can directly utilize a few a priori downlink channel samples to obtain $\Gamma_q$.

cooperative beamforming can then be posed as follows in this case,

$$\begin{aligned} \underset{r}{\text{minimize}} \quad & K_M \sum_i d_i(r_i) \\ \text{subject to} \quad & \Pr\Big(\sum_i \alpha(r_i) < \alpha_{R,\text{th}}\Big) < \Pr_{\text{out}} \\ & r_i \in \mathcal{N}(r_i^0),\ i = 1, \cdots, N, \end{aligned} \tag{3.5}$$

where Pr(.) denotes the probability of the argument, $\alpha(r_i) = \sqrt{\Gamma_{\text{lin}}(r_i)}$ is the random variable that represents the received channel amplitude when the $i^{\text{th}}$ node transmits from $r_i$, and $\Pr_{\text{out}}$ is the maximum tolerable outage probability.

As discussed in Chapter 2, the predicted channel power (in dB) when transmitting from unvisited locations $r = [r_1 \cdots r_N]^{\mathrm{T}} \in \mathcal{W}^N$ can be represented as a Gaussian random vector $\Gamma(r) \sim \mathcal{N}(\overline{\Gamma}(r), \Sigma(r))$, where $\overline{\Gamma}(r)$ and $\Sigma(r)$ are the estimated mean and covariance matrix respectively. Thus, the channel amplitude at locations $r$, $\alpha(r) = [\alpha(r_1) \cdots \alpha(r_N)]^{\mathrm{T}} = \Big[\sqrt{\Gamma_{\text{lin}}(r_1)} \cdots \sqrt{\Gamma_{\text{lin}}(r_N)}\Big]^{\mathrm{T}}$ is a lognormal random vector, i.e., $[20 \log_{10} \alpha(r_1) \cdots 20 \log_{10} \alpha(r_N)]^{\mathrm{T}} \sim \mathcal{N}(\overline{\Gamma}(r), \Sigma(r))$.

$\sum_{i=1}^{N} \alpha(r_i)$ is then the sum of lognormal random variables. As established in the literature, the lognormal distribution is a good approximation for the distribution of the sum of lognormal random variables [77]. Let $\alpha_{\text{sum}}$ with distribution $20 \log_{10} \alpha_{\text{sum}} \sim \mathcal{N}(\overline{\Gamma}_{\text{sum}}, \sigma^2_{\text{sum}})$ denote the lognormal random variable approximating $\sum_{i=1}^{N} \alpha(r_i)$. $\overline{\Gamma}_{\text{sum}}$ and $\sigma^2_{\text{sum}}$ can be found, based on $\overline{\Gamma}(r)$ and $\Sigma(r)$, by using the extended Fenton-Wilkinson (F-W) method [77]. The details are given in Appendix A.1.

The outage probability inequality in (3.5) can then be expressed as

$$\overline{\Gamma}_{\text{sum}} + \sigma_{\text{sum}} Q^{-1} (1 - \Pr_{\text{out}}) \geq 20 \log_{10}(\alpha_{R,\text{th}}),$$

where $Q(.)$ denotes the $Q$ function. Equation (3.5) can then be posed as

$$\begin{aligned} \underset{r}{\text{minimize}} \quad & K_M \sum_i d_i(r_i) \\ \text{subject to} \quad & \overline{\Gamma}_{\text{sum}} + \sigma_{\text{sum}} Q^{-1}\left(1 - \text{Pr}_{\text{out}}\right) \geq 20 \log_{10} \alpha_{R,\text{th}} \\ & r_i \in \mathcal{N}(r_i^0),\ i = 1, \cdots, N. \end{aligned} \tag{3.6}$$

We refer to this as the Motion Energy Stochastic Setting (MESS) minimization problem. The optimization problem (3.6) can then be solved by using existing optimization toolboxes. We next propose an alternative approach for the case of stochastic channel knowledge, based on our proposed MEMP approach of (3.3) of Section 3.2.1.

### Approximation using analysis of MEMP of (3.3)

In Section 3.2.1, we showed how the motion energy-aware optimization problem can be solved for the case of perfect channel knowledge. That analysis and the corresponding solution can be used to find an approximate solution for the stochastic case, as we show next. As introduced earlier, the channel power in dB, $20 \log_{10} \alpha(\overline{r}_j)$, has the distribution $20 \log_{10} \alpha(\overline{r}_j) \sim \mathcal{N}\left(\overline{\Gamma}(\overline{r}_j), \sigma^2(\overline{r}_j)\right)$, where $\overline{r}_j$ is the $j^{\text{th}}$ cell, as defined in Section 3.2.1, and $\overline{\Gamma}(\overline{r}_j)$ and $\sigma^2(\overline{r}_j) = \Sigma(\overline{r}_j)$ are obtained by evaluating Lemma 2.1 at $\overline{r}_j$ (scalar). Consider $\tilde{\alpha}(\overline{r}_j)$ such that $20 \log_{10} \tilde{\alpha}(\overline{r}_j) = \overline{\Gamma}(\overline{r}_j) - \zeta\sigma(\overline{r}_j)$ for some constant $\zeta \geq 0$. $\tilde{\alpha}(\overline{r}_j)$ provides a conservative estimate of the channel amplitude. We then approximate $\alpha(\overline{r}_j)$ by $\tilde{\alpha}(\overline{r}_j)$ in (3.3), which results in the following optimization problem:

$$\begin{aligned} \underset{\{z_{ij}\}}{\text{minimize}} \quad & J_{\text{MEMP}}(\{z_{ij}\}) \\ \text{subject to} \quad & \sum_i \sum_{j \in \mathcal{N}_i} \tilde{\alpha}(\overline{r}_j) z_{ij} \geq \alpha_{R,\text{th}} \\ & \sum_{j \in \mathcal{N}_i} z_{ij} = 1,\ z_{ij} \in \{0, 1\},\ \forall\, j \in \mathcal{N}_i,\ \forall\, i. \end{aligned} \tag{3.7}$$

Equation (3.7) can then be efficiently solved using the proposed approach of Section 3.2.1 for MEMP of (3.3). We next relate the optimization problem of (3.7) to the original optimization problem of (3.5) by finding a bound on the probability that the obtained solution satisfies the inequality constraint of (3.3). We first need the following lemma.

**Lemma 3.2** $20\log_{10}\alpha(\overline{r}_{j_i})$*s are positively correlated as a result of the exponential correlation, where* $j_i$ *is the cell chosen by robot* $i$*. We then have* $Pr\left(20\log_{10}\alpha(\overline{r}_{j_i}) \geq \xi_i,\ \forall i\right) \geq \prod_{i=1}^{N} Pr\left(20\log_{10}\alpha(\overline{r}_{j_i}) \geq \xi_i\right)$*, for any* $\xi_i \in \mathbb{R}$.

*Proof:* See [78]. ■

**Lemma 3.3** *Let* $\{z_{ij}\}$ *be the solution of (3.7), and let* $j_i$ *be such that* $z_{ij_i} = 1$*. Then the probability that this solution results in an outage in (3.3) is bounded as follows:* $Pr\left(\sum_{i=1}^{N}\alpha(\overline{r}_{j_i}) < \alpha_{R,th}\right) < 1 - \left[Q\left(-\zeta\right)\right]^{N}$.

*Proof:* The probability of successful transmission is $\Pr\left(\sum_{i=1}^{N}\alpha(\overline{r}_{j_i}) \geq \alpha_{R,\text{th}}\right) \geq \Pr\left(\sum_{i=1}^{N}\alpha(\overline{r}_{j_i}) \geq \sum_{i=1}^{N}\tilde{\alpha}(\overline{r}_{j_i})\right)$ since $\sum_{i=1}^{N}\tilde{\alpha}(\overline{r}_{j_i}) \geq \alpha_{R,\text{th}}$ for a feasible solution of (3.7). Further,

$$\begin{aligned}
\Pr\left(\sum_{i=1}^{N}\alpha(\overline{r}_{j_i}) \geq \sum_{i=1}^{N}\tilde{\alpha}(\overline{r}_{j_i})\right) &\geq \Pr\left(20\log_{10}\alpha(\overline{r}_{j_i}) \geq 20\log_{10}\tilde{\alpha}(\overline{r}_{j_i}),\ \forall i\right) \\
&\geq \prod_{i=1}^{N}\Pr\left(\frac{20\log_{10}\alpha(\overline{r}_{j_i}) - \overline{\Gamma}(\overline{r}_{j_i})}{\sigma(\overline{r}_{j_i})} \geq -\zeta\right) \\
&= \left[Q\left(-\zeta\right)\right]^{N}
\end{aligned}$$

where the second inequality follows from Lemma 3.2. ■

## 3.3 Energy-Aware Cooperative Robotic Beamforming

In this section, we extend our results of Section 3.2 to include the communication energy cost as well, i.e., we are interested in finding the most energy efficient way (considering both motion and communication) for the robots to cooperatively transmit the data to a remote station. The robots need to determine new locations for transmission as well as the transmission powers such that they minimize the total energy consumption while satisfying the cooperative connectivity requirement. As in Section 3.2, we start with the scenario of perfect channel knowledge, for which we obtain an $\epsilon$-suboptimal solution by showing that solving our problem is equivalent to solving a series of multiple-choice knapsack problems. We then extend our analysis to the stochastic setting with probabilistic channel prediction and incorporate channel uncertainty into our formulation.

### 3.3.1 Perfect Channel Knowledge

In this case, the robots perform distributed transmit beamforming with complex weights $\omega_i = \rho_i e^{-\angle h(r_i)}$, if node $i$ moves to $r_i$, where $0 \leq \rho_i \leq 1$ and $h(r_i)$ is as described in Section 3.1.3. The received power is then given as $P_R = P_0 \left(\sum_{i=1}^{N} \alpha(r_i)\rho_i\right)^2$. As can be seen from Section 3.1.2, imposing a minimum transmission rate requirement results in a minimum required received power which we denote by $P_{R,\text{th}}$ for a given $\eta_1, \eta_2 \leq 1$. For instance, in the case of uncoded MQAM, a bit error rate requirement of $\text{BER}_{\text{th}}$, results in $\eta_1 = 1$ and $\eta_2 = 1.5/\ln(5\text{BER}_{\text{th}})$, and a minimum spectral efficiency requirement would translate to a minimum required received power $P_{R,\text{th}}$ . Imposing this results in $P_R = P_0 \left(\sum_{i=1}^{N} \alpha(r_i)\rho_i\right)^2 \geq P_{R,\text{th}}$ or equivalently $\sum_{i=1}^{N} \alpha(r_i)\rho_i \geq \sqrt{\frac{P_{R,\text{th}}}{P_0}} = \alpha_{R,\text{th}}$, with

$\rho_i^2 P_0$ denoting the transmit power of robot $i$.

The total energy cost is then given as

$$J_{\mathrm{TE}}(r) = K_M \sum_{i=1}^{N} d_i(r_i) + \frac{n_{\mathrm{bits}} P_0 \sum_{i=1}^{N} \rho_i^2}{\eta_1 W \log_2 \left(1 + \eta_2 P_0 \frac{\left(\sum_{i=1}^{N} \alpha(r_i)\rho_i\right)^2}{N_0}\right)}$$

and the resulting optimization problem can be expressed as

$$\begin{aligned} \underset{r,\rho}{\text{minimize}} \quad & J_{\mathrm{TE}}(r) \\ \text{subject to} \quad & \sum_i \alpha(r_i)\rho_i \geq \alpha_{R,\mathrm{th}} \\ & 0 \leq \rho_i \leq 1,\ r_i \in \mathcal{N}(r_i^0),\ i = 1, \cdots, N, \end{aligned} \tag{3.8}$$

where $\rho = [\rho_1 \cdots \rho_N]^{\mathrm{T}}$, $n_{\mathrm{bits}}$ is the number of bits to be transmitted to the remote station and $\mathcal{N}(r_i^0) \subseteq \mathcal{W}$ is the neighborhood around $r_i^0$ that robot $i$ is constrained to be in.

As before, we first discretize the workspace $\mathcal{W}$ into $M$ cells with centers $\overline{r}_j \in \mathcal{W}, \quad \text{for } j \in \{1, \cdots, M\}$. The optimization problem (3.8) can then be reformulated as

$$\begin{aligned} \underset{\{z_{ij}\},\rho}{\text{minimize}} \quad & J_{\mathrm{TEMP}}(\{z_{ij}\}, \rho) \\ \text{subject to} \quad & \sum_i \sum_{j \in \mathcal{N}_i} \alpha(\overline{r}_j)\rho_i z_{ij} \geq \alpha_{R,\mathrm{th}} \\ & 0 \leq \rho_i \leq 1,\ i = 1, \cdots, N \\ & \sum_{j \in \mathcal{N}_i} z_{ij} = 1,\ z_{ij} \in \{0,1\},\ \forall\, j \in \mathcal{N}_i,\ \forall\, i, \end{aligned} \tag{3.9}$$

where $J_{\mathrm{TEMP}}(\{z_{ij}\}, \rho) = K_M \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_i} d_i(\overline{r}_j) z_{ij} + \frac{n_{\mathrm{bits}} P_0 \sum_{i=1}^{N} \rho_i^2}{\eta_1 W \log_2 \left(1 + \eta_2 P_0 \frac{\left(\sum_i \sum_j \alpha(\overline{r}_j)\rho_i z_{ij}\right)^2}{N_0}\right)}$, $d_i(\overline{r}_j)$ is the distance to cell $j$ for robot $i$, $\alpha(\overline{r}_j)$ is the channel amplitude when transmitting from cell $j$ to the remote station, and $\mathcal{N}_i \subseteq \{1, \cdots, M\}$ is the set of cells present in

$\mathcal{N}(r_i^0)$. A value of $z_{ij} = 1$ implies that robot $i$ moves to cell $j$. We refer to this problem as the Total Energy Minimization Problem (TEMP), with the optimal value of $J_{\mathrm{TEMP}}^{\mathrm{OPT}}$.

We begin by characterizing properties of the optimal communication strategy as well as of the optimal solution ($\{z_{ij}\}$,$\rho$) of TEMP of (3.9). This is one of the key intermediate steps that allows us to pose our problem as a series of multiple choice knapsack problems.

In the following lemma, we show that the inequality for the cooperative connectivity requirement in TEMP of (3.9) is satisfied with equality in the optimal solution.

**Lemma 3.4** *Let ($\{z_{ij}^{OPT}\}, \rho^{OPT}$) be an optimal solution of TEMP of (3.9). Let $j_i^{OPT}$ be such that $z_{ij_i^{OPT}}^{OPT} = 1$. Then the solution satisfies $\sum_{i=1}^{N} \alpha(\bar{r}_{j_i^{OPT}})\rho_i^{OPT} = \alpha_{R,th}$.*

*Proof:* See Appendix A.2 for the proof. ■

Next, consider the case where the positions of the robots are fixed and the only objective is to minimize the total transmit power (not energy) while satisfying the cooperative connectivity requirement. Lemma 3.5 characterizes the optimal solution of this case, as follows.

**Lemma 3.5** *Consider the following optimization problem:*

$$\begin{aligned} \underset{\rho}{\textit{minimize}} \quad & \sum_i \rho_i^2 \\ \textit{subject to} \quad & \sum_i \alpha_i \rho_i \geq \alpha_{R,th} \\ & 0 \leq \rho_i \leq 1,\ i = 1, \cdots, N. \end{aligned} \tag{3.10}$$

*The optimal solution for (3.10) is $\rho_i = \min\{\lambda\alpha_i, 1\}$ where $\lambda > 0$ is such that $\sum_i \min\{\lambda\alpha_i, 1\}\alpha_i = \alpha_{R,th}$.*[3]

*Proof:* See Appendix A.3 for the proof. ■

[3] *In this case, $\alpha_i$ can be interpreted as the fixed channel amplitude from robot $i$ to the remote station.*

Using Lemma 3.4 and 3.5 as building blocks, we next characterize the optimal communication strategy of TEMP of (3.9) given the final optimal positions of the robots.

**Lemma 3.6** *Let* $(\{z_{ij}^{OPT}\},\ \rho^{OPT})$ *be an optimal solution of TEMP of (3.9). Let* $j_i^{OPT}$ *be such that* $z_{ij_i^{OPT}}^{OPT} = 1$. *Then* $\rho_i^{OPT} = \min\{\lambda\alpha(\overline{r}_{j_i^{OPT}}), 1\}$ *where* $\lambda > 0$ *is such that* $\sum_{i=1}^{N} \min\{\lambda\alpha(\overline{r}_{j_i^{OPT}}), 1\}\alpha(\overline{r}_{j_i^{OPT}}) = \alpha_{R,th}$.

*Proof:* We prove this by contradiction. Assume that $\rho_i^{\mathrm{OPT}} \neq \rho_i^*$ where $\rho_i^* = \min\{\lambda\alpha(\overline{r}_{j_i^{\mathrm{OPT}}}), 1\}$, for $\lambda > 0$, such that $\sum_{i=1}^{N} \min\{\lambda\alpha(\overline{r}_{j_i^{\mathrm{OPT}}}), 1\}\alpha(\overline{r}_{j_i^{\mathrm{OPT}}}) = \alpha_{R,\mathrm{th}}$. Then, $(\{z_{ij}^{\mathrm{OPT}}\}, \rho^*)$ is a feasible solution of (3.9) since $\sum_{i=1}^{N} \alpha(\overline{r}_{j_i^{\mathrm{OPT}}})\rho_i^* = \alpha_{R,\mathrm{th}}$. The cost of the optimal solution then becomes

$$\begin{aligned}
J_{\mathrm{TEMP}}(\{z_{ij}^{\mathrm{OPT}}\}, \rho^{\mathrm{OPT}}) &= K_M \sum_{i=1}^{N} d_i(\overline{r}_{j_i^{\mathrm{OPT}}}) + K_C \sum_{i=1}^{N} (\rho_i^{\mathrm{OPT}})^2 \\
&> K_M \sum_{i=1}^{N} d_i(\overline{r}_{j_i^{\mathrm{OPT}}}) + K_C \sum_{i=1}^{N} (\rho_i^*)^2 \\
&= J_{\mathrm{TEMP}}(\{z_{ij}^{\mathrm{OPT}}\}, \rho^*),
\end{aligned}$$

following from Lemma 3.4 and Lemma 3.5, where $K_C = \frac{n_{\mathrm{bits}} P_0}{\eta_1 W \log_2\left(1+\eta_2 \frac{P_{R,\mathrm{th}}}{N_0}\right)}$. Thus, we have a contradiction, as we found a feasible solution with a lower cost. ∎

We next pose an optimization problem motivated by Lemma 3.6, the solution to which will help us obtain the solution to TEMP of (3.9). Let $K_C = \frac{n_{\mathrm{bits}} P_0}{\eta_1 W \log_2\left(1+\eta_2 \frac{P_{R,\mathrm{th}}}{N_0}\right)}$, as was defined in Lemma 3.6. Consider the following optimization problem,

$$\begin{aligned} \underset{\{z_{ij}\},\lambda}{\text{minimize}} \quad & J_\lambda(\{z_{ij}\},\lambda) \\ \text{subject to} \quad & \sum_i \sum_{j \in \mathcal{N}_i} \min\left\{\lambda\alpha(\overline{r}_j), 1\right\} \alpha(\overline{r}_j) z_{ij} \geq \alpha_{R,\text{th}} \\ & \sum_{j \in \mathcal{N}_i} z_{ij} = 1, \; z_{ij} \in \{0,1\}, \; \forall\, j \in \mathcal{N}_i, \; \forall\, i, \end{aligned} \tag{3.11}$$

where $J_\lambda(\{z_{ij}\},\lambda) = K_M \sum_i \sum_{j \in \mathcal{N}_i} d_i(\overline{r}_j) z_{ij} + K_C \sum_i \sum_{j \in \mathcal{N}_i} \left[\min\{\lambda\alpha(\overline{r}_j), 1\}\right]^2 z_{ij}$. We next show how to obtain an optimal solution for TEMP of (3.9) from an optimal solution for (3.11).

**Lemma 3.7** *Let $(\{z^*_{ij}\}, \lambda^*)$ be an optimal solution of (3.11). Let $j^*_i$ be such that $z^*_{ij^*_i} = 1$ and let $\rho^*_i = \min\{\lambda^*\alpha(\overline{r}_{j^*_i}), 1\}$. Then $(\{z^*_{ij}\}, \rho^*)$ is an optimal solution of TEMP of (3.9).*

*Proof:* Without loss of generality, let $\lambda^*$ be such that $\lambda^*\alpha(\overline{r}_{j^*_i}) \leq 1$ for some $i$. We first show that the connectivity requirement inequality in (3.11) is satisfied with equality for the optimal solution, i.e., $\sum_{i=1}^N \min\{\lambda^*\alpha(\overline{r}_{j^*_i}), 1\}\alpha(\overline{r}_{j^*_i}) = \alpha_{R,\text{th}}$. We show this by contradiction. Assume otherwise, i.e., $\sum_{i=1}^N \min\{\lambda^*\alpha(\overline{r}_{j^*_i}), 1\}\alpha(\overline{r}_{j^*_i}) > \alpha_{R,\text{th}}$. Consider $\lambda$ such that $\sum_{i=1}^N \min\{\lambda\alpha(\overline{r}_{j^*_i}), 1\}\alpha(\overline{r}_{j^*_i}) = \alpha_{R,\text{th}}$. Clearly we have $\lambda < \lambda^*$, which implies $\left[\min\{\lambda\alpha(\overline{r}_{j^*_i}), 1\}\right]^2 < \left[\min\{\lambda^*\alpha(\overline{r}_{j^*_i}), 1\}\right]^2$ for some $i$. Hence $J_\lambda(\{z^*_{ij}\}, \lambda) < J_\lambda(\{z^*_{ij}\}, \lambda^*)$, resulting in a contradiction. Thus $\sum_{i=1}^N \min\{\lambda^*\alpha(\overline{r}_{j^*_i}), 1\}\alpha(\overline{r}_{j^*_i}) = \alpha_{R,\text{th}}$.

Next, we show via contradiction that $(\{z^*_{ij}\},\ \rho^*)$, obtained from an optimal solution of (3.11), is an optimal solution of TEMP of (3.9). Assume $(\{z^*_{ij}\},\ \rho^*)$ is not an optimal solution of (3.9) and let $(z^{\text{OPT}}_{ij}, \rho^{\text{OPT}})$ be an optimal solution instead. From Lemma 3.6, we have that $\rho^{\text{OPT}}_i = \min\{\lambda^{\text{OPT}}\alpha(\overline{r}_{j^{\text{OPT}}_i}), 1\}$, where $\lambda^{\text{OPT}} > 0$ is such that

$\sum_{i=1}^{N} \min\{\lambda\alpha(\overline{r}_{j_i^{\mathrm{OPT}}}), 1\}\alpha(\overline{r}_{j_i^{\mathrm{OPT}}}) = \alpha_{R,\mathrm{th}}$. Then, we have

$$\begin{aligned} J_{\mathrm{TEMP}}(\{z_{ij}^{\mathrm{OPT}}\}, \rho^{\mathrm{OPT}}) &= K_M \sum_{i=1}^{N} d_i(\overline{r}_{j_i^{\mathrm{OPT}}}) + K_C \sum_{i=1}^{N} \Big[\min\{\lambda^{\mathrm{OPT}}\alpha(\overline{r}_{j_i^{\mathrm{OPT}}}), 1\}\Big]^2 \\ &= J_\lambda(\{z_{ij}^{\mathrm{OPT}}\}, \lambda^{\mathrm{OPT}}), \end{aligned}$$

and similarly $J_{\mathrm{TEMP}}(\{z_{ij}^*\}, \rho^*) = J_\lambda(\{z_{ij}^*\}, \lambda^*)$. As a result, $J_{\mathrm{TEMP}}(\{z_{ij}^*\}, \rho^*) > J_{\mathrm{TEMP}}(\{z_{ij}^{\mathrm{OPT}}\}, \rho^{\mathrm{OPT}})$ implies that $J_\lambda(\{z_{ij}^*\}, \lambda^*) > J_\lambda(\{z_{ij}^{\mathrm{OPT}}\}, \lambda^{\mathrm{OPT}})$. Thus, $(\{z_{ij}^{\mathrm{OPT}}\}, \lambda^{\mathrm{OPT}})$ is a feasible solution for (3.11) with a lower cost than $(\{z_{ij}^*\}, \lambda^*)$, resulting in a contradiction. ■

### $\epsilon$-Suboptimal Solution

In this subsection we pose a series of multiple-choice knapsack problems and relate their solution to TEMP of (3.9) to obtain an $\epsilon$-suboptimal solution. In this context, $\epsilon$ is a positive variable that determines how close to the optimal solution we can get. Basically, for each fixed value of $\lambda$, we have a multiple-choice knapsack problem, as can be seen from (3.11), which we can solve optimally. We then discretize $\lambda$ uniformly with $\epsilon$ determining the corresponding resolution. Let $\lambda_k = k\epsilon_1/\alpha_{\max}$ for $k \in \{k_{lb} - 1, \cdots, k_{ub} - 1\}$ and $\lambda_{k_{ub}} = 1/\alpha_{\min}^0$, where $\epsilon_1 = \frac{\epsilon}{2N\left(\alpha_{\max}/\alpha_{\min}^0\right)}$, $\alpha_{\max} = \max_{j\in\{1,\cdots,M\}} \alpha(\overline{r}_j)$ and $\alpha_{\min}^0 = \min\{\alpha(r_i^0) : i = 1, \cdots, N\}$ denotes the minimum channel amplitude among the initial positions of the robots. Furthermore, $k_{lb} = \left\lceil \frac{1}{\epsilon_1} \frac{\alpha_{R,\mathrm{th}}}{N\alpha_{\max}} \right\rceil = \left\lceil \frac{2}{\epsilon} \frac{\alpha_{R,\mathrm{th}}}{\alpha_{\min}^0} \right\rceil$ and $k_{ub} = \lceil \frac{1}{\epsilon_1} \frac{\alpha_{\max}}{\alpha_{\min}^0} \rceil = \left\lceil \frac{2N}{\epsilon} \left(\frac{\alpha_{\max}}{\alpha_{\min}^0}\right)^2 \right\rceil$ determine the range of $\lambda$, as explained next. Since $\sum_{i=1}^{N} \min\{\lambda\alpha(\overline{r}_{j_i}), 1\}\alpha(\overline{r}_{j_i}) \le \sum_{i=1}^{N} (\lambda_{k_{lb}-1}\alpha_{\max})\alpha_{\max} = \left(\left\lceil \frac{1}{\epsilon_1} \frac{\alpha_{R,\mathrm{th}}}{N\alpha_{\max}} \right\rceil - 1\right) \epsilon_1 N\alpha_{\max} < \alpha_{R,\mathrm{th}}$, $\lambda \le \lambda_{k_{lb}-1}$ could not be a feasible solution of (3.11). Moreover, an optimal solution $(\{z_{ij}^*\}, \lambda^*)$ would not involve a robot incurring motion energy to get to a location with a worse channel amplitude, resulting in $\alpha(\overline{r}_{j_i^*}) \ge \alpha_{\min}^0\ \forall i$. Since $\lambda_{k_{ub}}\alpha_{\min}^0 = 1$, we have that if $\lambda^* > \lambda_{k_{ub}}$ is an optimal solution, then $\lambda_{k_{ub}}$ is also an optimal solution. Thus, we need to only consider $\lambda \in (\lambda_{k_{lb}-1}, \lambda_{k_{ub}}]$ in (3.11), which results in the following optimization problem for each $k \in \{k_{lb}, \cdots, k_{ub}\}$:

$$\begin{aligned} \underset{\{z_{ij}\}}{\text{minimize}} \quad & J_{\lambda,k}(\{z_{ij}\}) \\ \text{subject to} \quad & \sum_i \sum_{j \in \mathcal{N}_i} [\min\{\lambda_k \alpha(\bar{r}_j), 1\}]\, \alpha(\bar{r}_j) z_{ij} \geq \alpha_{R,\text{th}} \\ & \sum_{j \in \mathcal{N}_i} z_{ij} = 1,\ z_{ij} \in \{0,1\},\ \forall\, j \in \mathcal{N}_i,\ \forall\, i, \end{aligned} \tag{3.12}$$

where $J_{\lambda,k}(\{z_{ij}\}) = \sum_i \sum_{j \in \mathcal{N}_i} (K_M d_i(\bar{r}_j) + K_C [\min\{\lambda_k \alpha(\bar{r}_j), 1\}]^2) z_{ij}$, with the optimum value of $J^{\text{OPT}}_{\lambda,k}$. This optimization problem can be solved similar to (3.3) by posing it as a knapsack problem through a change of variable as shown in Section 3.2.

Let $J_{\lambda,\min} = \min_{k \in \{k_{lb},\cdots,k_{ub}\}}\{J^{\text{OPT}}_{\lambda,k}\}$. In order to find $J_{\lambda,\min}$, we need to solve $k_{ub} - k_{lb} + 1 \leq \left\lceil \frac{2N}{\epsilon} \frac{\alpha_{\max}}{\alpha^0_{\min}} \left( \frac{\alpha_{\max}}{\alpha^0_{\min}} - \frac{\alpha_{R,\text{th}}}{N\alpha_{\max}} \right) \right\rceil + 1$ multiple choice knapsack problems. As can be seen, the number of knapsack problems to be solved grows linearly with $N$ and $\frac{1}{\epsilon}$. In the following theorem, we show how we can get arbitrarily close to the optimal solution by solving this set of knapsack problems.

**Theorem 3.1** *Let $m = \arg\min_{k \in \{k_{lb},\cdots,k_{ub}\}}\{J^{OPT}_{\lambda,k}\}$. Let $\{z^*_{ij}\}$ be a solution of (3.12) when $k = m$, and $j^*_i$ be such that $z^*_{ij^*_i} = 1$. Consider a $\lambda^*$ such that $\lambda^* \leq \lambda_m$ and $\sum_{i=1}^N \left[\min\{\lambda^* \alpha(\bar{r}_{j^*_i}), 1\}\right] \alpha(\bar{r}_{j^*_i}) = \alpha_{R,th}$. Further, set $\rho^*_i = \min\{\lambda^* \alpha(\bar{r}_{j^*_i}), 1\}$. Then, $(\{z^*_{ij}\}, \rho^*)$ is a feasible solution of TEMP of (3.9) and*

$$J_{TEMP}(\{z^*_{ij}\}, \rho^*) \leq J^{OPT}_{TEMP} + \epsilon \kappa_C. \tag{3.13}$$

*Proof:* It is straightforward to see that $(\{z^*_{ij}\}, \rho^*)$ is a feasible solution of TEMP of (3.9). Moreover, $J_{\text{TEMP}}(\{z^*_{ij}\}, \rho^*) = J_\lambda(\{z^*_{ij}\}, \lambda^*) \leq J_\lambda(\{z^*_{ij}\}, \lambda_m) = J^{\text{OPT}}_{\lambda,m} = J_{\lambda,\min}$ since

$J_\lambda(.)$ is a non-decreasing function of $\lambda$.

Let $(\{z_{ij}^{\text{OPT}}\}, \lambda^{\text{OPT}})$ be an optimal solution of (3.11). From Lemma 3.7 we have that $J_{\text{TEMP}}^{\text{OPT}} = J_\lambda(\{z_{ij}^{\text{OPT}}\}, \lambda^{\text{OPT}})$. Let $\lambda^{\text{OPT}} \in (\lambda_{k_{lb}-1}, \lambda_{k_{ub}}]$, as we established earlier. Thus, there exists a $k \in \{k_{lb}, \cdots, k_{ub}\}$ such that $\lambda^{\text{OPT}} \in (\lambda_{k-1}, \lambda_k]$. We then have

$$\begin{aligned}
J_{\text{TEMP}}(\{z_{ij}^*\}, \rho^*) \leq J_{\lambda,\min} \leq J_{\lambda,k}^{\text{OPT}} &\leq J_\lambda(\{z_{ij}^{\text{OPT}}\}, \lambda_k) \\
&= J_\lambda(\{z_{ij}^{\text{OPT}}\}, \lambda^{\text{OPT}}) + \left(J_\lambda(\{z_{ij}^{\text{OPT}}\}, \lambda_k) - J_\lambda(\{z_{ij}^{\text{OPT}}\}, \lambda^{\text{OPT}})\right) \\
&\leq J_{\text{TEMP}}^{\text{OPT}} + K_C \sum_{i=1}^{N} \left(\lambda_k^2 - (\lambda^{\text{OPT}})^2\right) \left(\alpha(\overline{r}_{j_i^{\text{OPT}}})\right)^2 \\
&\leq J_{\text{TEMP}}^{\text{OPT}} + K_C \sum_{i=1}^{N} \left(\lambda_k - \lambda_{k-1}\right)\left(\lambda_k + \lambda_k\right) \left(\alpha(\overline{r}_{j_i^{\text{OPT}}})\right)^2 \\
&\leq J_{\text{TEMP}}^{\text{OPT}} + 2\frac{\epsilon_1}{\alpha_{\max}} \lambda_{k_{ub}} K_C \sum_{i=1}^{N} \left(\alpha(\overline{r}_{j_i^{\text{OPT}}})\right)^2 \\
&\leq J_{\text{TEMP}}^{\text{OPT}} + 2N\epsilon_1 \frac{\alpha_{\max}}{\alpha_{\min}^0} K_C \\
&= J_{\text{TEMP}}^{\text{OPT}} + \epsilon K_C.
\end{aligned}$$

■

**Remark 3.2** *$K_C$ is the communication energy cost of a single robot when it transmits at maximum power and the robots satisfy the cooperative connectivity requirement with equality.*

**Remark 3.3** *Solving TEMP of (3.9) through a brute-force search of space is infeasible even for moderately small values of the number of robots ($N$). For instance, if $M$ is the number of points in the discretized workspace and if we represent each $\rho_i$ by $k$ bits, then the computational complexity of an exhaustive search is $M^N 2^{kN}$. On the other hand, with our proposed $\epsilon$-suboptimal solution, the number of multiple-choice knapsack problems to solve grows linearly with $N$ and $\frac{1}{\epsilon}$. We note that while we can solve (3.8) with an existing* *solver, there is no guarantee that the solver will find the global optimum since the objective function is non-convex. Theorem 3.1 then allows us to get arbitrarily close to the optimal solution with a low computational complexity.*

### 3.3.2 Probabilistic Channel Prediction

As discussed earlier, in realistic scenarios, the unmanned vehicles do not know the uplink channel when transmitting from unvisited locations. As such, they will probabilistically predict the channel based on a small number of a priori measurements in the same environment, as summarized in Section 2.2. The energy-aware (both motion and communication) cooperative beamforming problem (3.8), can then be extended to the following in this stochastic setting:

$$\begin{aligned} \underset{r,\rho}{\text{minimize}} \quad & J_{\text{TE,ST}}(r,\rho) \\ \text{subject to} \quad & \Pr\Big(\sum_i \alpha(r_i)\rho_i < \alpha_{R,\text{th}}\Big) < \Pr_{\text{out}} \\ & 0 \le \rho_i \le 1,\ r_i \in \mathcal{N}(r_i^0),\ i = 1, \cdots, N, \end{aligned} \tag{3.14}$$

where $J_{\text{TE,ST}}(r,\rho) = K_M \sum_{i=1}^{N} d_i(r_i) + \mathbb{E}\Big[\frac{n_{\text{bits}} P_0 \sum_i \rho_i^2}{\eta_1 W \log_2\left(1+\eta_2 \frac{P_0 (\sum_i \alpha(r_i)\rho_i)^2}{N_0}\right)}\Big]$, with $r = [r_1 \cdots r_N]^{\text{T}}$ and $\rho = [\rho_1 \cdots \rho_N]^{\text{T}}$ as optimization variables and $\mathbb{E}(.)$ representing the average of the argument. In this case, the average is taken over $\alpha(r_i),\ \forall i$. The vector $[\alpha(r_1)\rho_1 \cdots \alpha(r_N)\rho_N]^{\text{T}}$ is a lognormal random vector with distribution $[20\log_{10}(\alpha(r_1)\rho_1) \cdots 20\log_{10}(\alpha(r_N)\rho_N)]^{\text{T}} \sim \mathcal{N}\left(\rho_{\text{dB}} + \overline{\Gamma}(r), \Sigma(r)\right)$ where $\overline{\Gamma}(r)$ and $\Sigma(r)$ are the estimated mean and covariance matrix of the predicted channel power respectively, and $[\rho_{\text{dB}}]_i = 20\log_{10}\rho_i$. Let $\alpha_{\text{sum},\rho}$ with distribution $20\log_{10}\alpha_{\text{sum},\rho} \sim \mathcal{N}(\overline{\Gamma}_{\text{sum},\rho}, \sigma^2_{\text{sum},\rho})$ denote the lognormal random variable approximating $\sum_{i=1}^{N}\alpha(r_i)\rho_i$. $\overline{\Gamma}_{\text{sum},\rho}$ and $\sigma^2_{\text{sum},\rho}$ can be found, based on $\rho_{\text{dB}}$, $\overline{\Gamma}(r)$ and $\Sigma(r)$, by using the extended Fenton-Wilkinson method [77]. Similar to Section 3.2, the objective then becomes $J_{\text{TESS}}(r,\rho) = K_M \sum_i d_i(r_i) + \mathbb{E}\left[\frac{n_{\text{bits}} P_0}{\eta_1 W \log_2\left(1+\eta_2 \frac{P_0 \alpha^2_{\text{sum},\rho}}{N_0}\right)} \sum_i \rho_i^2\right]$, and our optimization problem can be rewritten as

$$\begin{aligned} \underset{r,\rho}{\text{minimize}} \quad & J_{\text{TESS}}(r,\rho) \\ \text{subject to} \quad & \overline{\Gamma}_{\text{sum},\rho} + \sigma_{\text{sum},\rho} Q^{-1}\left(1 - \text{Pr}_{\text{out}}\right) \geq 20 \log_{10} \alpha_{R,\text{th}} \\ & 0 \leq \rho_i \leq 1,\ i = 1, \cdots, N \\ & r_i \in \mathcal{N}(r_i^0),\ i = 1, \cdots, N, \end{aligned} \tag{3.15}$$

which can then be solved by using existing optimization toolboxes. We refer to this as the Total Energy Stochastic Setting (TESS) minimization problem.

### Approximation using analysis of TEMP of (3.9)

Similar to Lemma 3.3, the stochastic optimization problem of (3.14) can be approximately but efficiently solved using the solution to TEMP of (3.9), which is for the perfect channel knowledge case. As introduced earlier, the channel power in dB, $20\log_{10}\alpha(\overline{r}_j)$, has the distribution $20\log_{10}\alpha(\overline{r}_j) \sim \mathcal{N}\left(\overline{\Gamma}(\overline{r}_j), \sigma^2(\overline{r}_j)\right)$, where $\overline{r}_j$ is the $j^{\text{th}}$ cell, as defined in Section 3.2.1, and $\overline{\Gamma}(\overline{r}_j)$ and $\sigma^2(\overline{r}_j) = \Sigma(\overline{r}_j)$ are obtained by evaluating Lemma 2.1 at $\overline{r}_j$ (scalar). Consider $\tilde{\alpha}(\overline{r}_j)$ such that $20\log_{10}\tilde{\alpha}(\overline{r}_j) = \overline{\Gamma}(\overline{r}_j) - \zeta\sigma(\overline{r}_j)$ for some constant $\zeta \geq 0$. $\tilde{\alpha}(\overline{r}_j)$ provides a conservative estimate of the channel amplitude with a high probability. We then approximate $\alpha(\overline{r}_j)$ by $\tilde{\alpha}(\overline{r}_j)$ in (3.9), which results in the following optimization:

$$\begin{aligned}
\underset{\{z_{ij}\},\rho}{\text{minimize}} \quad & \tilde{J}_{\text{TESS}}(\{z_{ij}\},\rho) \\
\text{subject to} \quad & \sum_i \sum_{j\in\mathcal{N}_i} \tilde{\alpha}(\overline{r}_j)\rho_i z_{ij} \geq \alpha_{R,\text{th}} \\
& 0 \leq \rho_i \leq 1,\ i=1,\cdots,N \\
& \sum_{j\in\mathcal{N}_i} z_{ij} = 1,\ z_{ij}\in\{0,1\},\ \forall\, j\in\mathcal{N}_i,\ \forall\, i.
\end{aligned} \tag{3.16}$$

where $\tilde{J}_{\text{TESS}}(\{z_{ij}\},\rho) = K_M \sum_i \sum_{j\in\mathcal{N}_i} d_i(\overline{r}_j)z_{ij} + \frac{n_{\text{bits}}P_0\sum_i \rho_i^2}{\eta_1 W \log_2\left(1+\eta_2 P_0 \frac{\left(\sum_i \sum_{j\in\mathcal{N}_i}\tilde{\alpha}(\overline{r}_j)\rho_i z_{ij}\right)^2}{N_0}\right)}$. Equation (3.16) can then be efficiently solved using the proposed approach of Section 3.3.1 for TEMP. We next relate the optimization problem of (3.16) to the original optimization problem of (3.14) by finding a bound on the probability that the obtained solution satisfies the inequality constraint of (3.9).

**Lemma 3.8** *Let $(\{z_{ij}\},\rho)$ be the solution obtained when solving (3.16) using the proposed approach of Section 3.3.1, and let $j_i$ be such that $z_{ij_i}=1$. The probability that this solution results in an outage in (3.9) is bounded as follows:* $\Pr\left(\sum_i \alpha(\overline{r}_{j_i})\rho_i < \alpha_{R,th}\right) < 1-\left[Q\left(-\zeta\right)\right]^N$.

The proof is similar to the proof of Lemma 3.3.

## 3.4 Simulation Results

Consider a scenario where 6 robots are located in a 50 m × 50 m workspace with the remote station at the origin and initial positions as shown in Fig. 3.3. The channel is generated using the probabilistic channel model described in Section 2.1, with the following parameters that were obtained from real channel measurements in downtown

San Francisco [73] : $n_{\mathrm{PL}} = 4.4$, $\nu_{\mathrm{SH}} = 6.76$ and $\beta_{\mathrm{SH}} = 22.6$ m. Moreover, the multipath fading is taken to be uncorrelated Rician fading with the parameter $K_{\mathrm{ric}} = 3.9$. We consider a bandwidth of $W = 10$ MHz and the received noise power is taken to be a realistic value of $-100$ dBmW [79]. We consider uncoded MQAM modulation with a BER tolerance of $10^{-5}$ and a minimum spectral efficiency requirement (transmission rate divided by bandwidth) of[4] 4. This corresponds to $\eta_2 = 0.1515$ and a minimum received SNR requirement of 20 dB which, for the given noise power, corresponds to a received power requirement of $P_{R,\mathrm{th,dBm}} = -80$ dBmW. We take the maximum transmission power of a node to be $P_{0,\mathrm{dBm}} = 27$ dBmW [80], which results in $\alpha_{R,\mathrm{th},dB} = -53.5$ dB. The amount of data to be transferred is 800 bits/Hz. The robots are situated far enough from the remote station that they do not satisfy the received power requirement at their initial positions (see Fig. 3.3). The neighborhood $\mathcal{N}_i$, within which the final position of robot $i$ is constrained to lie in, is taken to be the entire workspace. The optimization problems of MEMP, MESS, TEMP and TESS, can be solved centrally by either one of the robots or by the remote station.

### 3.4.1 Perfect Channel Knowledge

We first analyze the trends of the motion energy-aware (MEMP) and the total energy-aware (TEMP) approaches as the communication load ($n_{\mathrm{bits}}/W$) varies in Figures 3.1, 3.2 and 3.3. TEMP of (3.9) is solved via the set of multiple-choice knapsack problems of Theorem 3.1 with $\epsilon = 0.05$. As shown in Theorem 3.1, the optimal value lies at most $0.05\kappa_C$ below the value obtained by solving the family of knapsack problems. This confidence bound is also shown in Fig. 3.1. It can be seen that the confidence bound is very close to the solution obtained by using Theorem 3.1 for solving TEMP, which confirms

[4]Note that the BER requirement is always satisfied, even if the minimum transmission rate requirement is not satisfied.

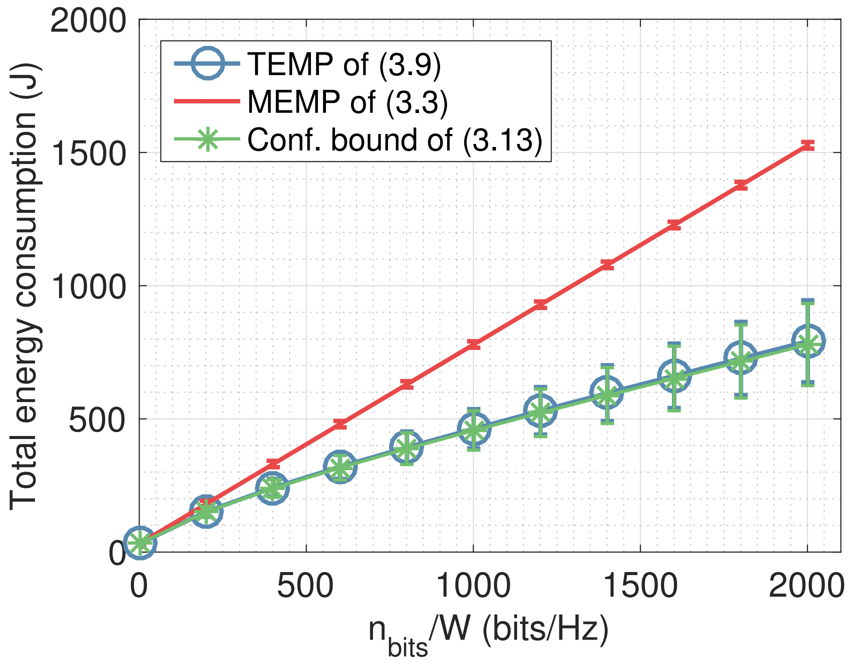


Figure 3.1: Total energy (sum of motion and communication) consumption of MEMP and TEMP for different communication loads for the case of perfect channel knowledge. TEMP provides a considerable energy saving, as expected. MEMP refers to the case where only motion energy is minimized while communication energy is also adapted and co-optimized in TEMP.

that Theorem 3.1 can get arbitrarily close to the optimal solution with a considerably low computational complexity. Each data point on the plots is obtained by averaging over 100 channels generated for the given set of channel parameters.

Fig. 3.1 shows the average total energy consumption of MEMP and TEMP. The figure also shows the corresponding error bars, representing the standard deviation of the total energy consumption for each data point. As can be seen, TEMP provides a significant energy saving as the communication load $n_{\text{bits}}/W$ increases. In TEMP, with increasing $n_{\text{bits}}/W$, the time for transmission increases as well, as a direct consequence of Lemma 3.4. Transmission power is thus penalized more and the robots travel larger distances to get to the locations with a better channel quality, allowing them to utilize a lower transmit power for communication. More specifically, Fig. 3.2 shows the total

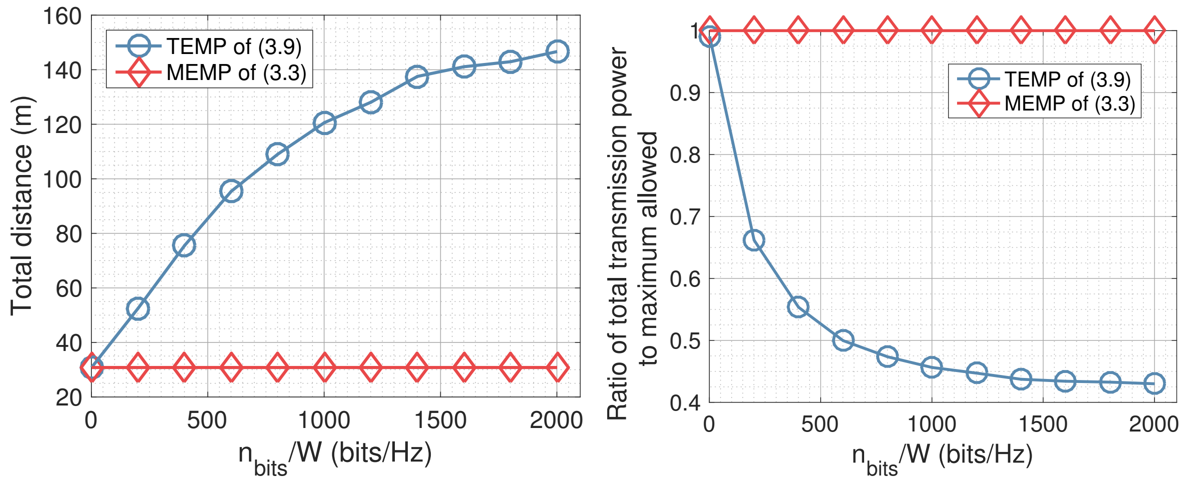


Figure 3.2: Total (left) distance traveled and (right) transmission power utilized by MEMP and TEMP as a function of communication load and for the case of perfect channel knowledge.

distance traveled as $n_{\text{bits}}/W$ varies. We can see that TEMP travels larger distances as $n_{\text{bits}}/W$ increases. Fig. 3.2 also shows how the total communication transmission power of TEMP decreases with increasing $n_{\text{bits}}/W$. This is due to the fact that in TEMP, by incurring more motion energy, the nodes can find spots with a better channel quality, resulting in a lower communication energy and a lower total energy consumption. Fig. 3.3 shows the behavior of the solution of MEMP and TEMP for communication loads of $n_{\text{bits}}/W = 100$ bits/Hz and $n_{\text{bits}}/W = 1500$ bits/Hz. The background color encodes the channel power to the remote station. The lighter (darker) areas correspond to regions with better (worse) channel quality. As expected, the TEMP solution moves larger distances in the high communication load case to get to locations with a better channel quality. This is as expected since communication energy is adaptive and penalized in this case, which has implications for path planning and motion energy consumption, allowing the robots to achieve a solution that is more efficient in total energy consumption (see Fig. 3.1).

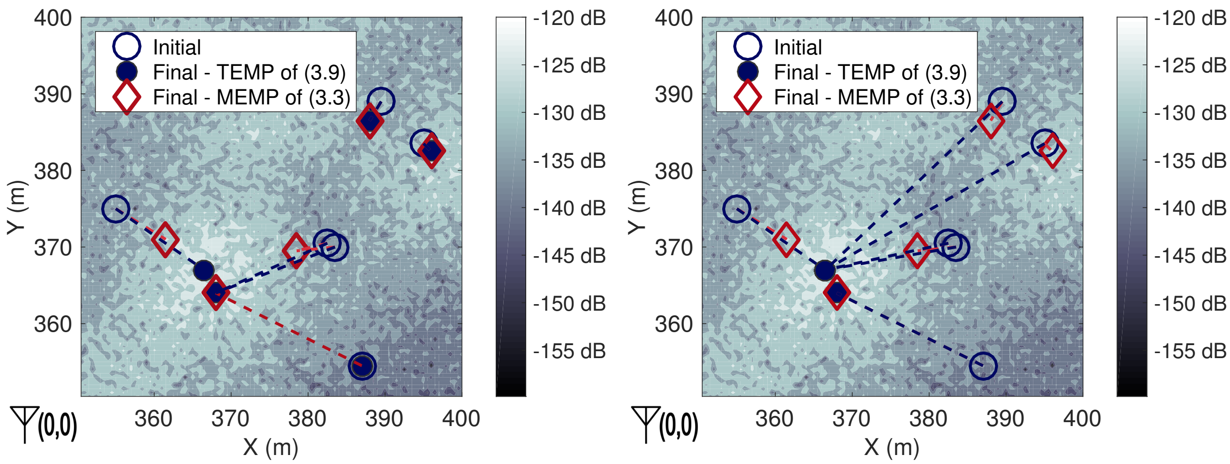


Figure 3.3: Solution of MEMP and TEMP for (left) low ($n_{\text{bits}}/W = 100$ bits/Hz) and (right) high ($n_{\text{bits}}/W = 1500$ bits/Hz) communication loads for the case of perfect channel knowledge. The background represents the uplink channel power with lighter (darker) regions corresponding to a better (worse) channel quality.

We next show the underlying trends of the motion energy-aware MEMP of (3.3) and the total energy-aware TEMP of (3.9) in Fig. 3.4, when varying the channel parameters for the case of perfect channel knowledge. For each plot, the channel parameter under consideration is varied while keeping the other parameters fixed at the values discussed earlier. Fig. 3.4 shows the energy consumption of MEMP and TEMP as a function of the channel parameters: the path loss exponent, shadowing power and shadowing decorrelation distance. Both MEMP and TEMP consume more energy as the path loss exponent increases. This is expected as an increase in the path loss exponent results in a worse channel quality, requiring the nodes to travel larger distances and/or increase their transmit power. Also, as can be seen, an increase in the shadowing power results in a decrease in the energy cost for both MEMP and TEMP. This is due to the fact that an increase in the shadowing power results in an increase in the spatial variation of the channel, allowing an unmanned vehicle to find a spot with a good channel quality with a smaller traveled distance. Moreover, as can been seen in Fig. 3.4, the total

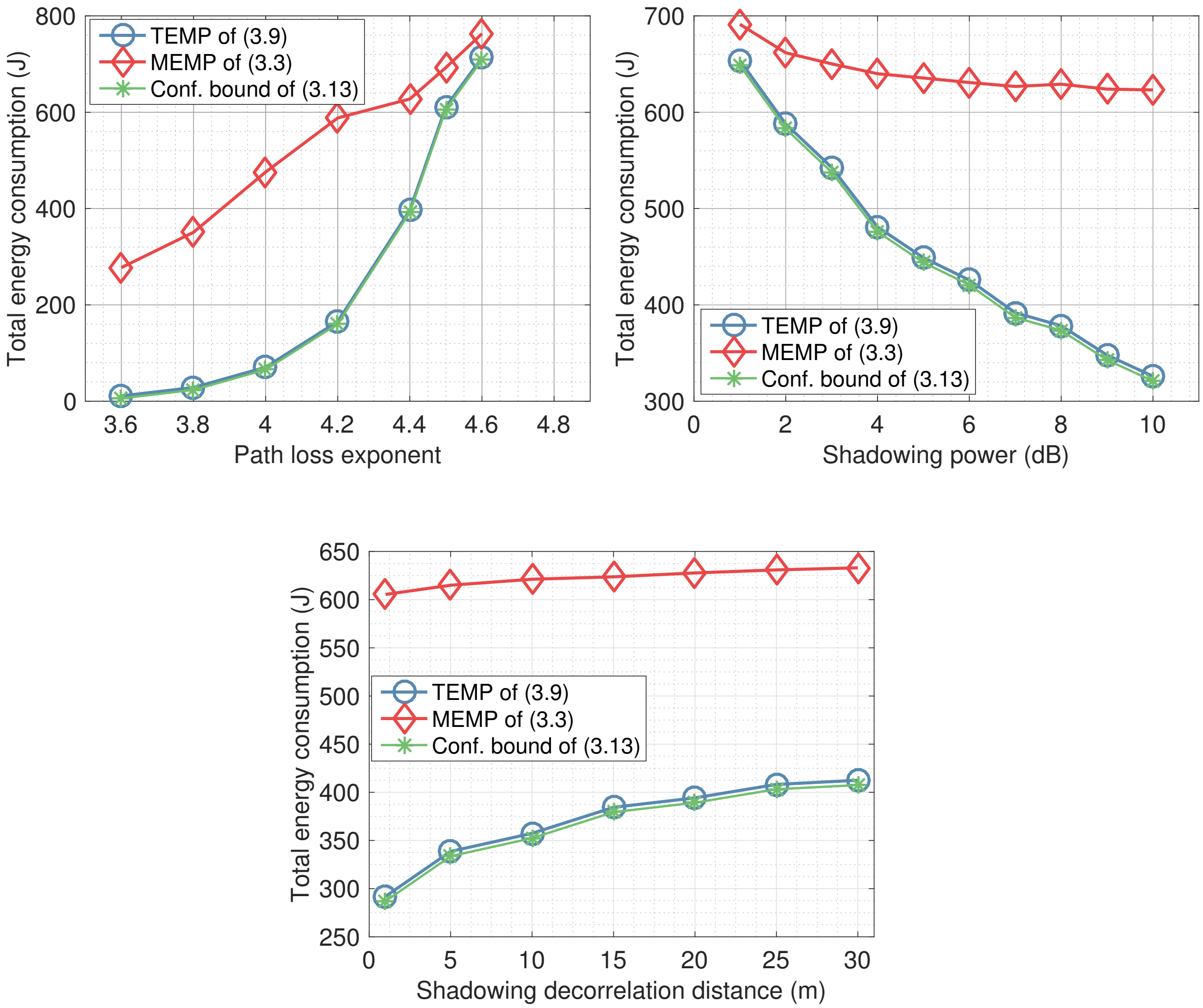


Figure 3.4: Total energy (sum of motion and communication) consumption of MEMP and TEMP for varying (top-left) path loss exponent, (top-right) shadowing power and (bottom) shadowing decorrelation distance, for the case of perfect channel knowledge. MEMP refers to the case where only motion energy is minimized while communication energy is also adapted and co-optimized in TEMP.

energy increases as we increase the decorrelation distance. This is due to the fact that an increase in the decorrelation distance reduces the spatial variation and a robot has to travel larger distances to find a location with a better channel quality.

### 3.4.2 Probabilistic Channel Prediction

We next consider the case where the channel is not known in the transmission from unvisited locations. We consider the workspace of Fig. 3.7. The robots are assumed to have 5% a priori channel measurements in this workspace. The robots then utilize the channel prediction framework of Section 2.2 for probabilistically predicting the channel at unvisited locations.[5] Channel and system parameters are as summarized earlier in this section, with $\mathrm{Pr}_{\mathrm{out}} = 0.2$. Fig. 3.5 shows the average total energy consumption as a function of $n_{\mathrm{bits}}/W$ for both MESS and TESS. The figure also shows the corresponding error bars which represents the standard deviation of the total energy consumption for each data point. MESS refers to the case where only motion energy is minimized, for the case of probabilistic channel prediction, while communication energy is also adapted and co-optimized in TESS. Curves marked by TESS and MESS denote the total energy consumption when the nodes move to the final locations and experience the true channel values. The label 'TESS predicted' in Fig. 3.5, on the other hand, is obtained from (3.15) using predicted channel values and the lognormal approximation. In other words, the predicted curve is what the nodes predict to consume while the TESS curve is the true consumption. As expected, we see a significant performance improvement when using the total energy-aware approach (TESS), as compared to the motion energy-aware approach (MESS), especially as the communication load increases. Fig. 3.6 shows the

[5]MESS and TESS are solved using MATLAB's *fmincon* solver. *fmincon* requires the objective and the constraints to be twice differentiable and is thus unable to handle uncorrelated multipath. We then assume an exponentially correlated multipath in the channel predictor with a very small decorrelation distance of $\beta_{\mathrm{MP}} = 0.033$ m, which has a negligible impact on the prediction performance. It should be noted that this is only for prediction purposes and that the real channel has an uncorrelated multipath.

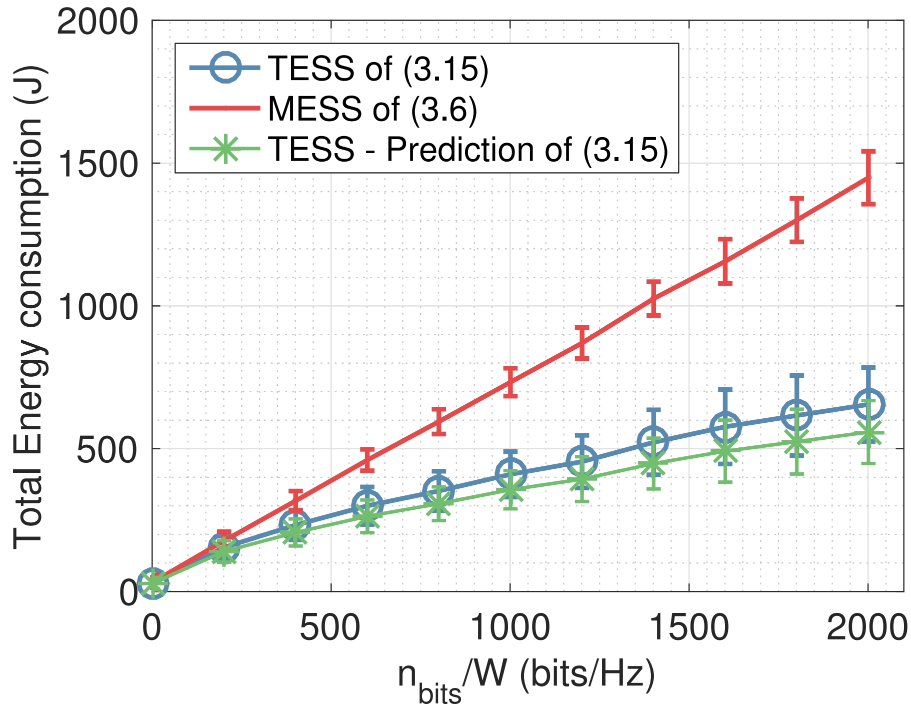


Figure 3.5: Energy consumption of MESS and TESS for different communication loads for the case of probabilistic channel prediction. TESS provides a considerable energy saving, as expected. MESS refers to the case where only motion energy is minimized for the case of probabilistic channel prediction while communication energy is also adapted and co-optimized in TESS.

total distance traveled by the robots for both MESS and TESS. Similar to the behavior of the perfect channel knowledge case, an increase in the communication load results in a larger penalization of the transmit power, and as a result TESS travels larger distances to get to locations with a better channel quality. Then TESS can use lower transmission powers, as can be seen in Fig. 3.6. Fig. 3.7 shows the behavior of MESS and TESS for communication loads of $n_{\text{bits}}/W = 100$ bits/Hz and $n_{\text{bits}}/W = 1500$ bits/Hz. The background color encodes the estimated channel power to the remote station. The lighter (darker) areas correspond to a better (worse) channel quality.

**Remark 3.4 (Computational complexity)** *In our simulations, the implementation is done in MATLAB except for the MCKP solver, which is in C, and is adapted from*

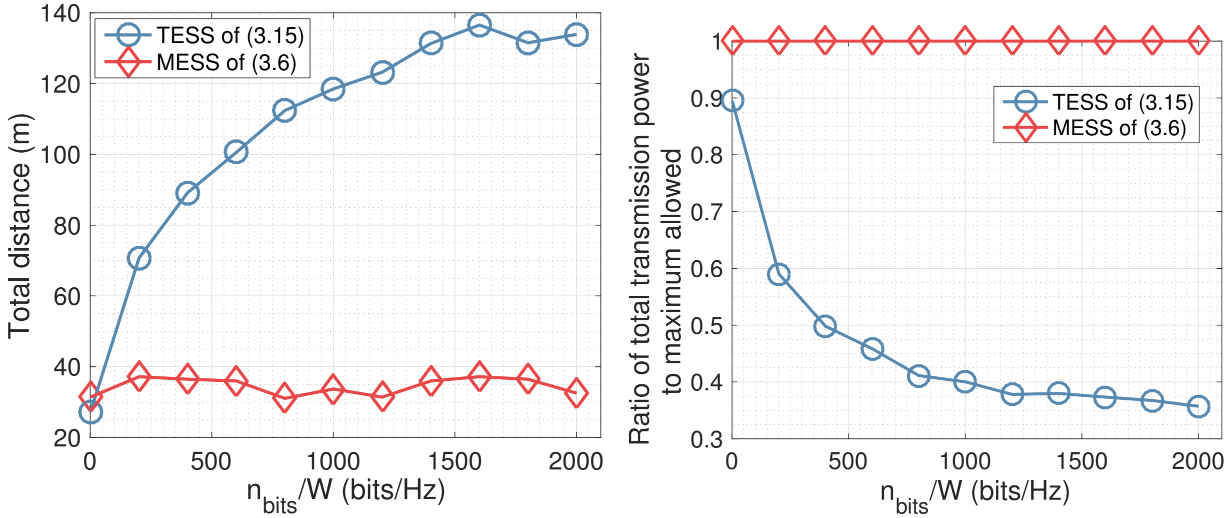


Figure 3.6: Total (left) distance traveled and (right) transmission power utilized by MESS and TESS for different communication loads for the case of probabilistic channel prediction.

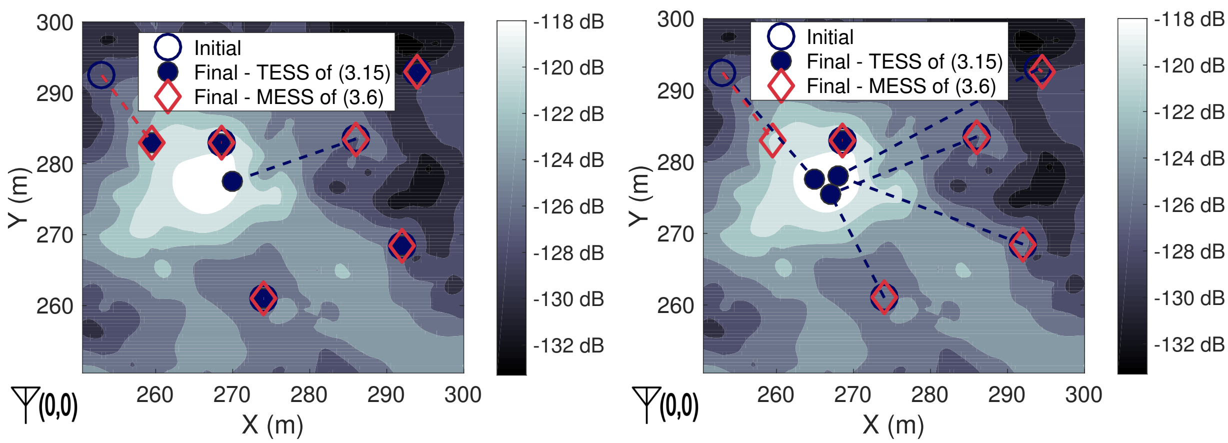


Figure 3.7: Solution of MESS and TESS for (left) low ($n_{\text{bits}}/W = 100$ bits/Hz) and (right) high ($n_{\text{bits}}/W = 1500$ bits/Hz) communication loads for the case of probabilistic channel prediction. The background represents the estimated channel power with lighter (darker) regions corresponding to a better (worse) channel quality.

| Scenario | TESS of (3.15) with target $\mathrm{Pr}_{\mathrm{out}} = 0.2$ | MESS of (3.6) with target $\mathrm{Pr}_{\mathrm{out}} = 0.2$ | Approx. of (3.16) with $\zeta = 0.1$ | Approx. of (3.7) with $\zeta = 0.1$ |
|---|---|---|---|---|
| Prob. of outage | 0.108 | 0.178 | 0.028 | 0.020 |

Table 3.1: Probability of outage for TESS and MESS as well as for the approximations of (3.16) and (3.7) of Sections 3.3.2 and 3.2.2. We can see that the target $\mathrm{Pr}_{\mathrm{out}}$ is satisfied for TESS and MESS.

*David Pisinger's implementation [76]. The simulations were run on a* 3.4 *GHz i7-3770 CPU. For the case of Fig. 3.3, solving MEMP took* 0.01 *s, and solving TEMP took* 25.2 *s and* 25.3 *s for the case of low ($n_{bits}/W = 100$ bits/Hz) and high ($n_{bits}/W = 1500$ bits/Hz) communication loads respectively. For the case of Fig. 3.7 of probabilistic channel knowledge, solving MESS took* 63.8 *s and* 53.2 *s for the case of low ($n_{bits}/W = 100$ bits/Hz) and high ($n_{bits}/W = 1500$ bits/Hz) communication loads respectively. Furthermore, solving TESS took* 122.7 *s and* 129.3 *s for the case of low and high communication loads respectively.*

Since the targeted outage probability was 0.2, some of the cases shown in Fig. 3.5 and 3.6 will result in outage, which means that the constraints of (3.9) and (3.3) will not be satisfied for TESS and MESS respectively. The first two columns of Table 3.1 show this outage probability, averaged over different communication loads, which is close to 0.2. In Sections 3.2.2 and 3.3.2, we proposed an alternative way for solving the probabilistic cases by choosing $20\log_{10}\tilde{\alpha}(\overline{r}_j) = \overline{\Gamma}(\overline{r}_j) - \zeta\sigma(\overline{r}_j)$ in (3.7) and (3.16) with $\zeta \geq 0$, and utilizing the corresponding deterministic solutions. Table 3.1 also shows the probability of outage for such a case with $\zeta = 0.1$, for both the total energy-aware and motion energy-aware cases. We can see that this approach provides a good performance, with a low outage probability, in this case. The total energy consumption is also similar to TESS and MESS in these results. The computational complexity of this approach is also low, as it takes advantage of our proposed theories of Sections 3.2.1 and 3.3.1. Choosing

$\zeta$ of $20 \log_{10} \tilde{\alpha}(\overline{r}_j) = \overline{\Gamma}(\overline{r}_j) - \zeta\sigma(\overline{r}_j)$ in a methodical way, in order to satisfy a target probability of outage, is a subject of future work.

In this chapter, we optimized only for the final location of the robots and did not consider the channel quality along the path traveled by the robot. Ideally, we would want to plan the entire path of the robot as opposed to just the final location. However, this is a very challenging problem as we now need to take into account the statistics of the channel along a path traveled by a robot. In the next chapter, we take the first step towards this: we characterize the statistics of the distance traveled by the mobile robot along a given path until it gets connected to a remote station.

# Chapter 4

# Statistics of the Distance Traveled until Connectivity for Unmanned Vehicles

In this chapter, we consider a scenario where a robot needs to establish connectivity with a remote station or another robot, as it moves along a predefined path. We are interested in answering the following question: what is the distance traveled by the robot along the path before it finds a connected spot? More specifically, we are interested in characterizing the statistics of the distance traveled along the path before it gets connected, in realistic channel environments experiencing path loss, shadowing and multipath effects. We refer to this problem as *the first passage distance (FPD) problem*, analogous to the concept of first passage time [83]. Figure 1.3 shows an example of such a scenario.

In this chapter, we mathematically characterize the probability density function (PDF) of the FPD as a function of the underlying channel parameters of the environment, such as shadowing, path loss, and multipath fading parameters. We do so for two cases: 1) when ignoring the multipath component (which could be of interest when the

Parts of this chapter have appeared in our conference submission [81], ©[2017] IEEE, and our submitted journal [82].

robot looks for an area of good connectivity as opposed to a single spot, or when multipath is negligible), and 2) when considering the multipath component. In both cases, we first develop an exact characterization of the statistics of the FPD for the setting with straight paths. We utilize tools from the stochastic equation literature to characterize the FPD while ignoring the multipath component, and develop a recursive characterization for the case when we include multipath. We then mathematically characterize a more general space of paths for which the analysis holds, based on properties of the path such as its curvature. Note that the PDF of the FPD can be directly computed via a high dimensional integration, as we will discuss in Section 4.3.1. However, the direct computation of this is infeasible for moderate distances. In this chapter, we utilize tools from the stochastic differential equation literature to significantly reduce this computational complexity. The derivations of the chapter can thus bring a foundational analytical understanding to the FPD and can significantly affect networked robotic operations. Note that the robot only needs to evaluate the five underlying channel parameters in order to calculate the PDF of FPD, using our derivations. As such, it can quickly evaluate it online after collecting a few online channel samples.

The chapter is organized as follows. In Section 4.1, we formally introduce the problem and briefly summarize the channel's underlying dynamics. In Section 4.2, we characterize the statistics of the distance traveled until connectivity while ignoring the multipath component. In Section 4.3, we characterize the statistics of the FPD while including the effect of multipath in the analysis. Finally, in Section 4.4, we validate our mathematical characterizations through extensive simulation with real channel parameters from downtown San Francisco.

## 4.1 Problem Setup

Consider a robot traveling along a given trajectory that needs to get connected to either a remote operator or another robot, as shown in Fig. 1.3. In order for the robot to successfully connect with the remote operator, the receptions need to satisfy a Quality of Service (QoS) requirement such as a target Bit Error Rate, which in turn results in a minimum required received Signal to Noise Ratio, or equivalently a minimum required channel power, given a fixed transmission power. We denote this minimum required received channel power as $\gamma_{\text{th}}$ in this chapter. This chapter then asks the following question: *What is the distance traveled by the robot along the path before it gets connected to the remote operator?* More specifically, we are interested in mathematically characterizing the probability density function (PDF) of this distance, for a given path, as a function of the underlying channel parameters, such as path loss, shadowing and multipath fading parameters, as well as the parameters of the path, such as its curvature.

### 4.1.1 Channel Model

As described in Chapter 2, the received channel power (in the dB domain) at location $q \in \mathbb{R}^2$ can be expressed as $\Gamma(q) = \gamma_{\text{PL}}(q) + \Gamma_{\text{SH}}(q) + \Gamma_{\text{MP}}(q)$ where $\gamma_{\text{PL}}(q) = K_{\text{dB}} - 10n_{\text{PL}} \log_{10} \|q\|$ is the distance-dependent path loss with $n_{\text{PL}}$ representing the path loss exponent, and $\Gamma_{\text{SH}}$ and $\Gamma_{\text{MP}}$ are random variables denoting the impact of shadowing and multipath respectively (in dB).

Consider the case where the robot is traveling along a path. Let $d$ be the distance traveled by the robot along this path. With a slight abuse of notation, in the rest of the chapter we let $\Gamma(d)$ represent the channel power when the robot has traveled distance $d$ along the path, as marked in Fig. 1.3. We thus have $\Gamma(d) = \gamma_{\text{PL}}(d) + \Gamma_{\text{SH}}(d) + \Gamma_{\text{MP}}(d)$.

## 4.2 Characterizing the FPD Without Considering Multipath

We start our analysis by ignoring the multipath and only considering the shadowing and path loss components of the channel, i.e., we want $\Gamma(d) = \gamma_{\text{PL}}(d) + \Gamma_{\text{SH}}(d)$ to be above $\gamma_{\text{th}}$. This assumption allows us to better analyze and understand the FPD, and paves the way towards our most general characterization of the next section, which includes multipath as well. Moreover, the analysis also has practical values of its own, and would be relevant to the case where the robot is interested in finding a general area of good connectivity as opposed to a single good spot. In this section, we will characterize the statistics of the distance traveled until connectivity for this scenario. We begin by analyzing straight paths in Section 4.2.1, where we utilize the stochastic differential equation literature [84] in our characterization. We then extend our analysis to a more general space of paths in Section 4.2.2.

### 4.2.1 Straight Paths: Stochastic Differential Equation Analysis

In this section, we characterize the PDF of the distance traveled until connectivity for straight-line paths. Consider a robot situated at a distance $d_{\text{src}}$ from a remote operator or from another robot to which it needs to be connected, and moving in the direction specified by the angle $\theta_{\text{src}}$, as shown in Fig. 4.1. The angle $\theta_{\text{src}}$ is measured clockwise with respect to the line segment connecting the remote operator and the robot, as can be seen in Fig. 4.1, and denotes the direction of travel chosen by the robot.

$\Gamma(d)$ represents the channel power when the robot is at distance $d$ along direction

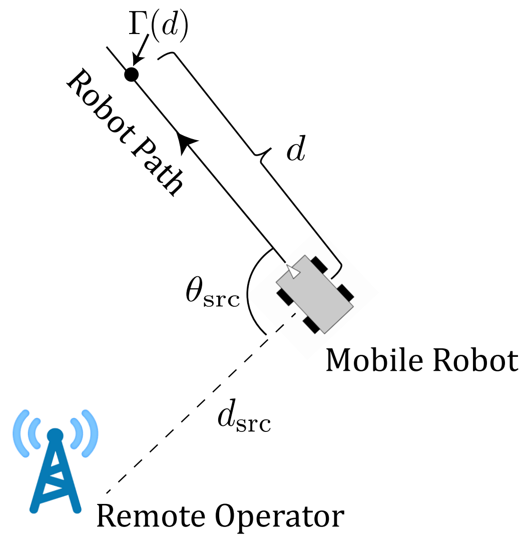


Figure 4.1: An example of the considered scenario for a straight path.

$\theta_{\rm src}$, as marked in Fig. 4.1. We thus have $\Gamma(d) = \gamma_{\rm PL}(d) + \Gamma_{\rm SH}(d)$, where

$$\gamma_{\rm PL}(d) = K_{\rm dB} - 5n_{\rm PL}\log_{10}(d_{\rm src}^2 + d^2 - 2d_{\rm src}d\cos\theta_{\rm src}), \tag{4.1}$$

and $\Gamma_{\rm SH}(d)$ is a zero mean Gaussian process with the spatial correlation of $\mathbb{E}\{\Gamma_{\rm SH}(b)\Gamma_{\rm SH}(d)\} = \sigma_{\rm SH}^2 e^{-(d-b)/\beta_{\rm SH}}$, with $d \geq b$. Note that $\Gamma(d)$ is also a function of $d_{\rm src}$ and $\theta_{\rm src}$. We drop $\Gamma(d)$'s dependency on them in the notation as the analysis of the chapter is carried out for a fixed $d_{\rm src}$ and $\theta_{\rm src}$.

As we shall see, $\Gamma_{\rm SH}(d)$ becomes an Ornstein-Uhlenbeck process, one of the most studied types of Gauss-Markov processes [84–87]. Ornstein-Uhlenbeck process appears in many practical scenarios, such as Brownian motion, financial stock markets, or neuronal firing [86], [87], and thus has been heavily studied in the literature. In this chapter, we shall utilize this rich literature [84], [88] to mathematically characterize the FPD to connectivity for a mobile robot.

We begin by summarizing the definitions of a Gaussian process and a Markov process.

**Definition 4.1 (Gaussian Process [89])** *A stochastic process* $\{X(t) : t \in \mathcal{T}\}$*, where* $\mathcal{T}$ *is an index set, is a Gaussian process, if any finite number of samples have a joint Gaussian distribution, i.e.,* $(X(t_1), X(t_2), \cdots, X(t_k))$ *is a Gaussian random vector for all* $t_1, \cdots, t_k \in \mathcal{T}$ *and for all* $k$.

A Gaussian process is completely specified by its mean function $m_X(t) = \mathbb{E}[X(t)]$ and its covariance function $\Sigma_X(t_1, t_2) = \mathbb{E}\{[X(t_1) - m_X(t_1)][X(t_2) - m_X(t_2)]\}$. We use the notation $X \sim \mathcal{GP}(m_X, \Sigma_X)$ to denote the underlying process.

**Definition 4.2 (Markov Process [90])** *A process* $X(t)$ *is Markov if*

$$\Pr(X(t_n) \leq x_n | X(t_{n-1}), \cdots, X(t_1)) = \Pr(X(t_n) \leq x_n | X(t_{n-1})),$$

*for all* $n$ *and for all* $t_n \geq t_{n-1} \geq \cdots \geq t_1$*, where* $\Pr(.)$ *denotes the probability of the argument.*

**Definition 4.3 (Gauss-Markov Process [91])** *A stochastic process is Gauss-Markov if it satisfies the requirements of both a Gaussian process and a Markov process.*

We next state a lemma that shows when a Gaussian process is also Markov, which we shall utilize to prove that the channel shadowing power $\Gamma_{\mathrm{SH}}(d)$ is Gauss-Markov.

**Lemma 4.1** *A Gaussian process* $X \sim \mathcal{GP}(m_X, \Sigma_X)$ *is Markov if and only if*

$$\Sigma_X(t_1, t_3) = \frac{\Sigma_X(t_1, t_2)\Sigma_X(t_2, t_3)}{\Sigma_X(t_2, t_2)},$$

*for all* $t_3 \geq t_2 \geq t_1$.

*Proof:* See [92] for the proof. ■

**Corollary 4.1** *The channel shadowing power $\Gamma_{SH}(d)$ and the channel power $\Gamma(d)$ are Gauss-Markov processes.*

*Proof:* $\Gamma_{\mathrm{SH}} \sim \mathcal{GP}(0, \Sigma_{\Gamma_{\mathrm{SH}}})$ is a Gaussian process with zero mean and covariance function $\Sigma_{\Gamma_{\mathrm{SH}}}(b,d) = \sigma_{\mathrm{SH}}^2 e^{-(d-b)/\beta_{\mathrm{SH}}}$. This covariance function satisfies $\Sigma_{\Gamma_{\mathrm{SH}}}(b,t)\Sigma_{\Gamma_{\mathrm{SH}}}(t,d)/\Sigma_{\Gamma_{\mathrm{SH}}}(t,t) = \sigma_{\mathrm{SH}}^2 e^{-(d-t)-(t-b)/\beta_{\mathrm{SH}}} = \Sigma_{\Gamma_{\mathrm{SH}}}(b,d)$, for $b \geq t \geq d$, which concludes the proof for $\Gamma_{\mathrm{SH}}(d)$ using Lemma 4.1. The channel power $\Gamma(d)$ is the sum of $\Gamma_{\mathrm{SH}}(d)$ and a mean function (path loss function $\gamma_{\mathrm{PL}}(d)$). Thus, the channel power is also a Gauss-Markov process with distribution $\Gamma \sim \mathcal{GP}(\gamma_{\mathrm{PL}}, \Sigma_{\Gamma_{\mathrm{SH}}})$. ■

**Remark 4.1 (see [84])** *The Ornstein-Uhlenbeck process $O \sim \mathcal{GP}(0, \Sigma_O)$ is a Gauss-Markov process with the covariance function $\Sigma_O(b,d) = \sigma^2 e^{-(d-b)/\beta}$, where $\sigma \geq 0$ and $\beta \geq 0$ are constants. Thus, we can see that $\Gamma_{SH}(d)$ is an Ornstein-Uhlenbeck process.*

In order to gain more insight into the stochastic process $\Gamma(d)$, we next discuss the transition PDF $f(\gamma, d|\eta, b) = \frac{\partial}{\partial\gamma}\Pr\left(\Gamma(d) < \gamma | \Gamma(b) = \eta\right)$, where $d \geq b$, as well as the stochastic differential equation governing $\Gamma(d)$, both of which we shall subsequently use in our characterization of the PDF of the FPD.

### The Underlying Stochastic Differential Equation

The transition PDF $f(\gamma, d|\eta, b)$ characterizes the distribution of $\Gamma(d)$ given $\Gamma(b) = \eta$. This is a normal density characterized by a mean and variance of (see 10.5 of [93])

$$\begin{aligned} \mathbb{E}\left[\Gamma(d)|\Gamma(b) = \eta\right] &= \gamma_{\mathrm{PL}}(d) + e^{-(d-b)/\beta_{\mathrm{SH}}}(\eta - \gamma_{\mathrm{PL}}(b)) \\ \mathrm{Var}\left[\Gamma(d)|\Gamma(b) = \eta\right] &= \sigma_{\mathrm{SH}}^2(1 - e^{-2(d-b)/\beta_{\mathrm{SH}}}). \end{aligned} \tag{4.2}$$

The transition PDF explicitly shows the spatial dependence of the channel power $\Gamma(d)$. As stated in [88], $f(\gamma, d|\eta, b)$ satisfies the partial differential equation known as the forward

Fokker-Planck equation:[1]

$$\frac{\partial}{\partial d} f(\gamma, d|\eta, b) = -\frac{\partial}{\partial \gamma}\left[A_{\mathrm{FP}}(\gamma, d) f(\gamma, d|\eta, b)\right] + \frac{1}{2}\frac{\partial^2}{\partial \gamma^2}\left[B_{\mathrm{FP}} f(\gamma, d|\eta, b)\right], \tag{4.3}$$

with the associated initial condition of $f(\gamma, b|\eta, b) = \delta(\gamma - \eta)$, where $A_{\mathrm{FP}}(\gamma, d) = \gamma'_{\mathrm{PL}}(d) - (\gamma - \gamma_{\mathrm{PL}}(d))/\beta_{\mathrm{SH}}$, $B_{\mathrm{FP}} = (2\sigma^2_{\mathrm{SH}})/\beta_{\mathrm{SH}}$ and $\gamma_{\mathrm{PL}}(d)$ is as stated in (4.1), with its derivative:

$$\gamma'_{\mathrm{PL}}(d) = -10 n_{\mathrm{PL}} \log_{10}(e) \frac{d - d_{\mathrm{src}} \cos\theta_{\mathrm{src}}}{d^2_{\mathrm{src}} + d^2 - 2 d_{\mathrm{src}} d \cos\theta_{\mathrm{src}}}.$$

The Fokker-Planck equation shows the evolution of the probability density $f(\gamma, d|\eta, b)$ with the traveled distance $d$ given $\Gamma(b) = \eta$.

Moreover, as shown in [84], the channel power $\Gamma(d)$ can be represented as a stochastic differential equation:[2]

$$\mathrm{d}\Gamma(d) = A_{\mathrm{FP}}(\Gamma, d)\mathrm{d}d + \sqrt{B_{\mathrm{FP}}}\mathrm{d}\overline{W}(d), \tag{4.4}$$

where $\overline{W}(d)$ is the Wiener process and $A_{\mathrm{FP}}(\gamma, d)$ and $B_{\mathrm{FP}}$ are as defined before.

**Remark 4.2** *In (4.3) and (4.4), $A_{FP}(\gamma, d)$ and $B_{FP}$ are known as the drift and the diffusion components respectively. The drift $A_{FP}(\gamma, d) = \gamma'_{PL}(d) - (\gamma - \gamma_{PL}(d))/\beta_{SH}$ is a pull towards the mean, and the diffusion component $B_{FP} = (2\sigma^2_{SH})/\beta_{SH}$ is a function of the shadowing variance and the decorrelation distance. Then, in an increment $\Delta d$, we can think of the channel power spatially evolving with a deterministic rate $A_{FP}(\gamma, d)$, in addition to a random Gaussian term with the variance $B_{FP}\Delta d$.*

Next, we utilize our established lemmas to derive the PDF of the FPD.

[1] The Fokker-Planck equation of [88] is stated for a general Gauss-Markov process. Here we adapted it for our specific Gauss-Markov process $\Gamma(d)$.

[2] [84] provides the stochastic differential equation for the Ornstein-Uhlenbeck process, from which we can easily obtain 4.4.

**First Passage Distance**

Consider the random variable $\mathcal{D}_{\gamma_0} = \inf_{d \geq 0}\{d : \Gamma(d) \geq \gamma_{\text{th}} | \Gamma(0) = \gamma_0 < \gamma_{\text{th}}\}$. This denotes the FPD of the process $\Gamma(d)$ to the connectivity threshold $\gamma_{\text{th}}$, with the initial value $\Gamma(0) = \gamma_0 < \gamma_{\text{th}}$. Further, let $g[d|\gamma_0] = \frac{\partial}{\partial d}\text{Pr}\left(\mathcal{D}_{\gamma_0} < d\right)$ represent the PDF of the FPD. In the following theorem, we characterize this PDF.

**Theorem 4.1** *The PDF of FPD $g[d|\gamma_0]$ satisfies the following non-singular second-kind Volterra integral equation:*

$$g[d|\gamma_0] = -2\Psi[d|\gamma_0, 0] + 2\int_0^d g[b|\gamma_0]\Psi[d|\gamma_{th}, b]\mathrm{d}b, \tag{4.5}$$

*where $\gamma_0 < \gamma_{th}$ and*

$$\Psi[d|\eta, b] = \Bigg\{ -\frac{1}{2}\frac{\mathrm{d}\gamma_{PL}(d)}{\mathrm{d}d} - \frac{\gamma_{th} - \gamma_{PL}(d)}{2\beta_{SH}}\frac{1 + e^{-2(d-b)/\beta_{SH}}}{1 - e^{-2(d-b)/\beta_{SH}}} + \frac{\eta - \gamma_{PL}(b)}{\beta_{SH}}\frac{e^{-(d-b)/\beta_{SH}}}{1 - e^{-2(d-b)/\beta_{SH}}}\Bigg\} f(\gamma_{th}, d|\eta, b). \tag{4.6}$$

*Proof:* The proof is based on the fact that $\Gamma(d)$ is a Gauss-Markov process and utilizes the Fokker-Planck equation (4.3). The details are then adapted from Theorem 3.1 of [88] to our particular Gauss-Markov process. ■

$\mathcal{D}_{\gamma_0}$ represents the FPD for a given initial value of $\Gamma(0) = \gamma_0$. In many scenarios, we are instead interested in characterizing the FPD for the initial state $\Gamma(0)$ being a random variable bounded from above by $\gamma_{\text{th}}$, i.e., we are interested in characterizing the FPD when the starting position is not connected. This is known as the upcrossing FPD in the general first passage literature [88]. We next extend our analysis to derive the PDF of the upcrossing FPD. Let the random variable $\mathcal{D}^{(\epsilon)}_{\Gamma_0} = \inf_{d \geq 0}\{d : \Gamma(d) \geq \gamma_{\text{th}} | \Gamma(0) < \gamma_{\text{th}} - \epsilon\}$ denote the $\epsilon$-upcrossing FPD of $\Gamma(d)$ to the boundary $\gamma_{\text{th}}$ given that the initial state satisfies $\Gamma(0) < \gamma_{\text{th}} - \epsilon$, where $\epsilon > 0$ is a fixed real number. The $\epsilon$-upcrossing FPD, $\mathcal{D}_{\Gamma_0}^{(\epsilon)}$, can be characterized as follows:

$$\Pr\left(\mathcal{D}_{\Gamma_0}^{(\epsilon)} < d\right) = \int_{-\infty}^{\gamma_{\text{th}}-\epsilon} \Pr\left(\mathcal{D}_{\gamma_0} < d\right) \zeta_\epsilon(\gamma_0)\mathrm{d}\gamma_0,$$

where $\mathcal{D}_{\gamma_0}$ is the FPD given the initial value $\Gamma(0) = \gamma_0 < \gamma_{\text{th}}$, as defined earlier, and

$$\zeta_\epsilon(\gamma_0) = \begin{cases} \frac{f(\gamma_0,0)}{\Pr(\Gamma(0)<\gamma_{\text{th}}-\epsilon)}, & \gamma_0 < \gamma_{\text{th}} - \epsilon \\ 0, & \gamma_0 \geq \gamma_{\text{th}} - \epsilon \end{cases},$$

is the PDF of $\Gamma(0)|\Gamma(0) < \gamma_{\text{th}} - \epsilon$ with $f(\gamma, d)$ denoting the PDF of $\Gamma(d)$. Moreover, the $\epsilon$-upcrossing FPD density $g_u^{(\epsilon)}[d] = \frac{\partial}{\partial d}\mathrm{P}(\mathcal{D}_{\Gamma_0}^{(\epsilon)} < d)$ is similarly related to the FPD density $g[d|\gamma_0]$ as follows: $g_u^{(\epsilon)}[d] = \int_{-\infty}^{\gamma_{\text{th}}-\epsilon} g[d|\gamma_0]\zeta_\epsilon(\gamma_0)\mathrm{d}\gamma_0$.

**Remark 4.3** *Note that we have required $\epsilon > 0$. This is due to the fact that the mathematical tools we shall utilize are not well-defined for $\gamma_0 = \gamma_{th}$. However, $\epsilon$ can be chosen arbitrarily small.*

In the following theorem, we derive an expression for $g_u^{(\epsilon)}[d]$, the PDF of the $\epsilon$-upcrossing FPD.

**Theorem 4.2** *The PDF of the $\epsilon$-upcrossing FPD, $g_u^{(\epsilon)}[d]$, satisfies the following non-singular second-kind Volterra integral equation:*

$$g_u^{(\epsilon)}[d] = -2\Psi_u^{(\epsilon)}[d] + 2\int_0^d g_u^{(\epsilon)}[b]\Psi[d|\gamma_{th}, b]\mathrm{d}b, \tag{4.7}$$

*where* $\Psi[d|\eta, b]$ *is as defined in (4.6),*

$$\Psi_u^{(\epsilon)}[d] = \frac{1}{2\Pr(\Gamma(0) < \gamma_{th} - \epsilon)} \left\{ \frac{-2\sigma_{SH}^2}{\beta_{SH}} e^{-d/\beta_{SH}} f(\gamma_{th} - \epsilon, 0) f[\gamma_{th}, d|\gamma_{th} - \epsilon, 0] \right.$$
$$\left. + \frac{1}{2} f(\gamma_{th}, d)(1 + \mathrm{Erf}[\Upsilon_\epsilon(d)]) \left( -\frac{\mathrm{d}\gamma_{PL}(d)}{\mathrm{d}d} - \frac{1}{\beta_{SH}} \left[\gamma_{th} - \gamma_{PL}(d)\right] \right) \right\},$$

*with* $\mathrm{Erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} \mathrm{d}t$ *representing the error function, and*

$$\Upsilon_\epsilon(d) = \frac{\gamma_{th} - \epsilon - \gamma_{PL}(0) - e^{-d/\beta_{SH}} (\gamma_{th} - \gamma_{PL}(d))}{\sqrt{2\sigma_{SH}^2 (1 - e^{-2d/\beta_{SH}})}}.$$

*Proof:* The proof is obtained by adapting Theorem 5.3 of [88] to our particular Gauss-Markov process form. ■

In terms of implementation, the functions $\Psi[d|\eta, b]$ and $\Psi_u^\epsilon[d]$ in Theorem 4.1 and Theorem 4.2 can be easily computed. The PDF of the FPD ($g[d|\gamma_0]$) and the PDF of the $\epsilon$-upcrossing FPD ($g_u^{(\epsilon)}[d]$) can then be computed from the integral equations (4.5) and (4.7) respectively. In particular, Simpson rule provides the basis for an efficient iterative algorithm for evaluating these integrals (See Section 4 of [88]).

**Remark 4.4 (Computational complexity)** *The direct computation of* $g_u^{(\epsilon)}[d]$ *involves a high dimension integration, as we will discuss in Section 4.3.1. For a discretized path of* $N$ *steps, this direct computation would have a computational cost exponential in* $N$*, i.e.* $O(NM^N)$ *for some constant* $M$*. In contrast, the computation cost of* $g_u^{(\epsilon)}[d]$ *using Theorem 4.2 is* $O(N^2)$*. Moreover, Theorem 4.2 is also an elegant characterization of the* $\epsilon$*-upcrossing FPD that can be utilized for analysis and design of robotic operations.*

### 4.2.2 Approximately-Markovian Paths

In this section, we characterize the space of paths (beyond straight paths) that results in approximately-Markovian processes. As we saw in Section 4.2.1, the channel shadowing component along a straight line is a Gauss-Markov process. This allowed us to characterize the statistics of the distance to connectivity for a mobile robot traveling along a straight path. A general non-straight path is not Markovian since the covariance function $\Sigma_{\Gamma_{\mathrm{SH}}}(b, d)$ does not satisfy Lemma 4.1. In this section, we characterize the space of paths for which the channel shadowing power along the path is approximately a Gauss-Markov process. This allows us to immediately apply the stochastic differential equation analysis of Section 4.2.1 to characterize the statistics of the distance until connectivity for these paths.

Consider the scenario in Fig. 4.2 (top), where we have discretized the path with $\Gamma_{\mathrm{SH},-0}$ denoting the shadowing power at the current location and $\Gamma_{\mathrm{SH},-1}, \Gamma_{\mathrm{SH},-2}, \cdots$ indicating the channel shadowing power at previously-visited points.[3] In Section 4.2.2, we saw that a Gauss-Markov process satisfies the Fokker-Planck equation of (3), which provides us with the result of Theorem 4.2. The Fokker-Planck equation in turn requires the property that $p(\gamma_{\mathrm{SH},-0}|\gamma_{\mathrm{SH},-1}, \gamma_{\mathrm{SH},-2}, \cdots) = p(\gamma_{\mathrm{SH},-0}|\gamma_{\mathrm{SH},-1})$ for its derivation (through the Chapman-Kolmogorov equation [84]). Thus, we say a path is approximately-Markovian, if at every point on the path, we have that $p(\gamma_{\mathrm{SH},-0}|\gamma_{\mathrm{SH},-1}, \gamma_{\mathrm{SH},-2}, \cdots)$ is close to $p(\gamma_{\mathrm{SH},-0}|\gamma_{\mathrm{SH},-1})$. We will characterize this closeness precisely in Section 4.2.2 using the Kullback-Leibler (KL) divergence metric.

Our key insight is that the approximate Markovian nature is related to the curvature of a path, which is a measure of how much the path curves, i.e., how much it deviates from a straight line. For instance, a straight line has a curvature of 0. Thus, we would

[3]Note that the discretization step size of the path must be small for the derivations of Theorem 4.2 to be valid.

expect that paths with small enough curvature would result in approximately-Markovian processes. We will precisely characterize what we mean by this in Section 4.2.2.

We first describe an outline of our approach for characterizing the space of approximately-Markovian paths. At every point on the path, instead of checking for the conditional distribution given all the past points on the path, which is cumbersome, we consider all past points on the path within a certain distance of the current point, i.e., within a ball centered at the current point. In other words, to check the approximately-Markovian property, we evaluate $p(\gamma_{\text{SH},-0}|\gamma_{\text{SH},-1}, \gamma_{\text{SH},-2}, \cdots, \gamma_{\text{SH},-n})$ instead of $p(\gamma_{\text{SH},-0}|\gamma_{\text{SH},-1}, \gamma_{\text{SH},-2}, \cdots)$. Fig. 4.2 (top) shows an illustration of this. This makes sense since the shadowing component has an exponential correlation function. Thus, if the radius of the ball is large enough, the points outside of the ball will have a negligible impact on the estimate at the center of the ball. We will characterize this radius in Section 4.2.2. Thus, our strategy is to roll a ball along the path, as shown in Fig. 4.2 (bottom), and to check if the approximate Markovian property holds at each point along the path. We then characterize two conditions that can ensure that a path will be approximately-Markovian. The first is that, at any point on the path, if we travel backward along the path it should not loop either within the ball or such that it re-enters the ball. We refer to such looping as $d_{\text{th}}$-looping ($d_{\text{th}}$ being the radius of the ball), and examples of this are shown in Fig. 4.3. Equivalently, a path is called $d_{\text{th}}$-loop-free if there is no $d_{\text{th}}$-looping. The second condition is that the maximum curvature of the path should be smaller than a certain bound, which will be characterized later in Section 4.2.2. If the $d_{\text{th}}$-loop-free condition is satisfied, then the only part of the path that lies within the ball would lie in the shaded region of Fig. 4.4, and if the maximum curvature of the path is small enough, then the path will be approximately-Markovian. We will formulate this precisely in Section 4.2.2.

We start by mathematically characterizing the $d_{\text{th}}$-looping condition in detail.

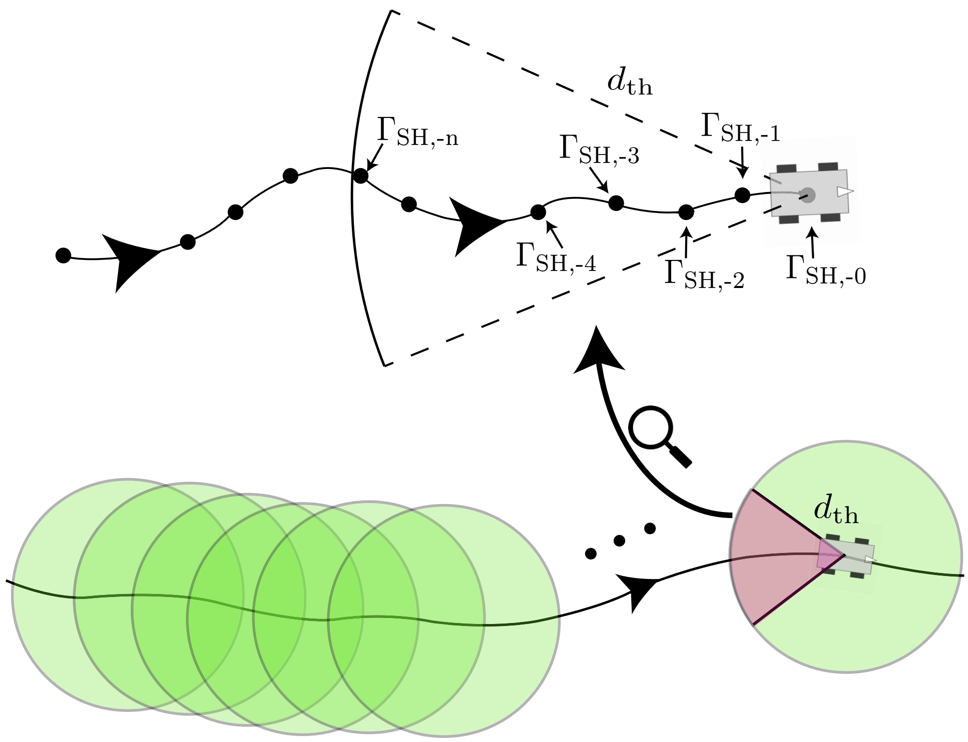


Figure 4.2: (bottom) A ball with radius $d_{\mathrm{th}}$ rolling along the path, where we check for approximate Markovianity within each ball, and (top) the discretized path and the corresponding channel shadowing power values within a ball.

### $d_{\mathbf{th}}$-Loop-Free Constraint

We define $d_{\text{th}}$-loop-free paths as paths where neither of the two following scenarios occurs at any point on the path. The first is when traveling backward along a path, the path loops within the ball itself. More precisely, when traveling backward along the path, let the initial direction of travel be along the negative x-axis. We say that the path loops within the ball if at any point (still inside the ball), the direction of travel has a component along the positive x-axis (e.g., Fig. 4.3 (left)). The second scenario is when the path re-enters the ball once it leaves it. These two scenarios, which we collectively refer to as $d_{\text{th}}$-looping, are illustrated in Fig. 4.3. Such $d_{\text{th}}$-looping behavior can possibly invalidate the approximate Markovian nature of the path.

We next relate the $d_{\text{th}}$-loop-free condition to the curvature of the path. We first review the precise definition of curvature.

**Definition 4.4 (Curvature [94])** *The curvature of a planar path $r(b) = (x(b), y(b))$ parameterized by arc-length is defined as*

$$\kappa(b) = \|\upsilon_T'(b)\|,$$

*where $\upsilon_T(b)$ is the unit tangent vector at $b$.*

When traveling backward along a path, consider the segment of the path inside the ball, before the path exits the ball. Let $r_{\text{ball}}$ refer to this segment, as shown in Fig. 4.4. Moreover, let $d_{r_{\text{ball}}}$ refer to its length. The following lemma characterizes some important properties of $r_{\text{ball}}$.

**Lemma 4.2** *For a path with maximum curvature $\kappa$ and a ball with radius $d_{th}$, the path segment $r_{ball}$ satisfies the following properties:*

1. *$r_{ball}$ lies within the shaded region of Fig. 4.4 where the boundary of the region corresponds to circular arcs with curvature $\kappa$.*

2. *If $\kappa < 1/d_{th}$, $r_{ball}$ cannot loop within the ball (see Fig. 4.3 (left) for an example of looping within the ball).*

3. *The length of the segment $r_{ball}$ satisfies*

$$d_{r_{ball}} < \frac{1}{\kappa} \sin^{-1} (\kappa \times d_{th}) .$$

*Proof:* See Appendix B.1 for the proof. ■

Then, a sufficient condition for a $d_{\mathrm{th}}$-loop-free path is given as follows.

**Lemma 4.3 ($d_{\mathrm{th}}$-loop-free path)** *Consider a planar path $r(b) = (x(b), y(b))$ parameterized by arc length, i.e., $b$ denotes the arc length. Let $\kappa$ be the maximum curvature of the path. The path is $d_{th}$-loop-free if it satisfies $\kappa < 1/d_{th}$ and*

$$\|r(b) - r(b-d)\| > d_{th},$$

*for $d > \frac{1}{\kappa} \sin^{-1} (\kappa d_{th})$ and for all $b$.*

*Proof:* From Lemma 4.2, we know that if $\kappa < 1/d_{\mathrm{th}}$, the path cannot loop within the ball, preventing the condition of Fig. 4.3 (left). Moreover, from Lemma 4.2, it can easily be confirmed that $\|r(b) - r(b-d)\|$ for $d > \frac{1}{\kappa} \sin^{-1} (\kappa d_{\mathrm{th}})$ is the euclidean distance from the center to a point on the part of the path that has left the ball. Thus, if $\|r(b) - r(b-d)\| > d_{\mathrm{th}}$, for $d > \frac{1}{\kappa} \sin^{-1} (\kappa d_{\mathrm{th}})$ the path cannot re-enter the ball (i.e., scenario of Fig. 4.3 (right) is not possible). ■

**Remark 4.5** *Any path can be reparameterized by arc length. Details on this can be found in [95].*

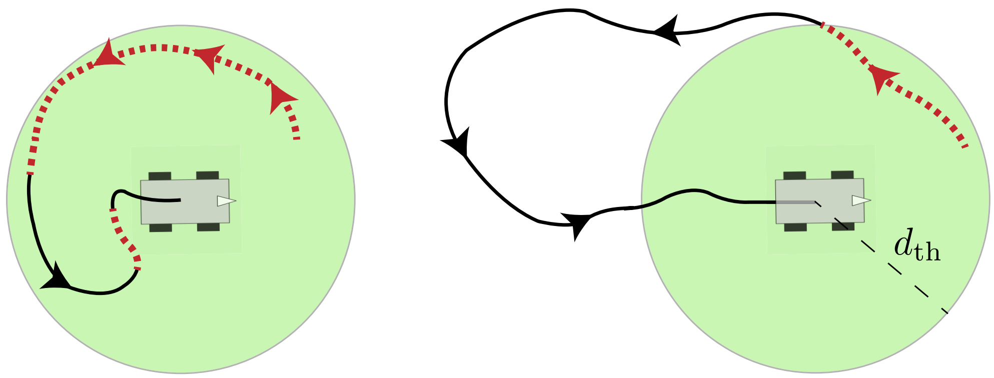


Figure 4.3: Two scenarios of $d_{\text{th}}$-looping: (left) path loops within the ball and (right) path loops back to re-enter the ball. The parts causing the loop in either scenario is denoted by the dashed red line.

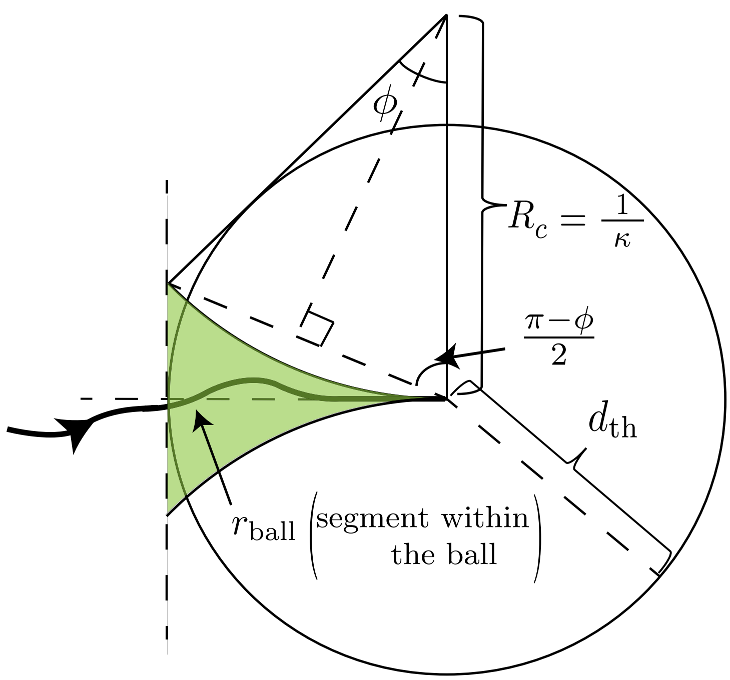


Figure 4.4: A path of maximum curvature $\kappa$ would lie within the shaded area. A sample such path is shown.

We next characterize the similarity or dissimilarity between the true distribution $p(\gamma_{\text{SH},-0}|\gamma_{\text{SH},-1},\cdots,\gamma_{\text{SH},-n})$ and its Markov approximation $p(\gamma_{\text{SH},-0}|\gamma_{\text{SH},-1})$ using the KL divergence metric. We then utilize this to obtain sufficient conditions on the ball radius and the curvature of a path for the approximate Markovian nature to hold.

**Approximately-Markovian: KL Divergence metric**

Consider a path as shown in Fig. 4.2 (top). Let $\Gamma_{\text{SH},-0}$ be the channel shadowing power at the current location and $\Gamma_{\text{SH},-1},\cdots,\Gamma_{\text{SH},-n}$ be the channel shadowing power at the past $n$ locations along the path. From Section 4.1.1, we know that $\Gamma_{\text{SH},-0},\cdots,\Gamma_{\text{SH},-n}$ are jointly Gaussian random variables. The distribution of $\Gamma_{\text{SH},-0}|\Gamma_{\text{SH},-1},\cdots,\Gamma_{\text{SH},-n}$ is then given as $\mathcal{N}(m,\sigma^2)$, where

$$m = \Sigma_{0,1:n}^T \Sigma_{1:n}^{-1} \Gamma_{\text{SH},-1:n}, \tag{4.8}$$

$$\sigma^2 = \sigma_{\text{SH}}^2 - \Sigma_{0,1:n}^T \Sigma_{1:n}^{-1} \Sigma_{0,1:n}, \tag{4.9}$$

with $\Gamma_{\text{SH},-1:n} = [\Gamma_{\text{SH},-1},\cdots,\Gamma_{\text{SH},-n}]^T$, $\Sigma_{0,1:n} = \mathbb{E}[\Gamma_{\text{SH},-0}\Gamma_{\text{SH},-1:n}]$ and $\Sigma_{1:n} = \mathbb{E}[\Gamma_{\text{SH},-1:n}\Gamma_{\text{SH},-1:n}^T]$ (see 10.5 of [93]). Moreover, $\mathbb{E}[\Gamma_{\text{SH},-i}\Gamma_{\text{SH},-j}] = \sigma_{\text{SH}}^2 e^{-\|q_i - q_j\|/\beta_{\text{SH}}}$, where $q_i \in \mathbb{R}^2$ is the location corresponding to $\Gamma_{\text{SH},-i}$. Let $c = \Sigma_{1:n}^{-1}\Sigma_{0,1:n}$ denote the coefficients of the mean. We then have $m = c^T \Gamma_{\text{SH},-1:n} = c_1 \Gamma_{\text{SH},-1} + \cdots + c_n \Gamma_{\text{SH},-n}$.

We want to approximate this distribution with the Markovian distribution $\Gamma_{\text{SH},-0}|\Gamma_{\text{SH},-1} \sim \mathcal{N}(\hat{m},\hat{\sigma}^2)$ where $\hat{m} = \varrho \Gamma_{\text{SH},-1}$ and $\hat{\sigma}^2 = \sigma_{\text{SH}}^2(1-\varrho^2)$, with $\varrho = e^{-\Delta d/\beta_{\text{SH}}}$, and $\Delta d$ being the step size of the path. We first characterize the difference between the means, given as $\Delta m = m - \hat{m} = \Delta c^T \Gamma_{\text{SH},-1:n}$, where $\Delta c = [c_1 - \varrho, c_2, \cdots, c_n]^T$. $\Delta m$ is thus a zero-mean Gaussian random variable $\mathcal{N}(0,\sigma_{\Delta m}^2)$, where

$$\sigma_{\Delta m}^2 = \Delta c^T \Sigma_{1:n} \Delta c. \tag{4.10}$$

We will compare how close the true distribution and its approximation are using the KL divergence metric. We first review the definition of KL divergence.

**Definition 4.5 (KL Divergence [96])** *The KL divergence between two distributions* $p(x)$ *and* $\tilde{p}(x)$ *is defined as*

$$KL = \int p(x) \log_e \frac{p(x)}{\tilde{p}(x)} dx.$$

KL divergence is a measure of the distance between two distributions [96]. We will utilize this as a measure of the goodness of the approximation: the smaller the KL divergence, the better the approximation. The following lemma gives us the expression for this KL divergence.

**Lemma 4.4** *The KL divergence between* $\mathcal{N}(m, \sigma^2)$ *and its approximation* $\mathcal{N}(\hat{m}, \hat{\sigma}^2)$ *is given as*

$$KL = \frac{\sigma_{\Delta m}^2}{2\hat{\sigma}^2}\chi_1^2 + \frac{1}{2}\left(\frac{\sigma^2}{\hat{\sigma}^2} - 1 - \log_e \frac{\sigma^2}{\hat{\sigma}^2}\right), \tag{4.11}$$

*where* $\chi_1^2 = (m - \hat{m})^2/\sigma_{\Delta m}^2$.

*Proof:* See [97] for the proof. ■

Since $m$ and $\hat{m}$ are functions of $\Gamma_{\mathrm{SH},-1}, \cdots, \Gamma_{\mathrm{SH},-n}$, they are random variables. Thus, $\chi_1^2$ becomes a Chi-squared random variable with one degree of freedom since $(m - \hat{m}) \sim (0, \sigma_{\Delta m}^2)$ [98], and the KL divergence of (4.11) becomes a random variable. More specifically, from (4.11), we know that the KL divergence is a scaled Chi-squared random variable with an offset term. We use the mean $m_{KL}$ and the standard deviation $\sigma_{KL}$ of the KL divergence to capture the deviation of the Markov approximation from the true distribution. The smaller these values are, the better the approximation is. In our approach, we set maximum tolerable values for the mean and the standard deviation as

$\epsilon_m$ and $\epsilon_\sigma$ respectively. Then, we say that the distribution is approximately-Markovian for the parameters $\epsilon_m$ and $\epsilon_\sigma$ if we satisfy $m_{KL} \leq \epsilon_m$ and $\sigma_{KL} \leq \epsilon_\sigma$.

We next consider the setting with 3 points in space, as shown in Fig. 4.5 (top), where we have the current point ($\Gamma_{\text{SH},-0}$), the previous point ($\Gamma_{\text{SH},-1}$) and a general point in space ($\Gamma_{\text{SH},r}$). We are interested in mathematically characterizing the impact of $\Gamma_{\text{SH},r}$ on the estimate at the current point, i.e., how good an approximation $\Gamma_{\text{SH},-0}|\Gamma_{\text{SH},-1} \sim \mathcal{N}(\hat{m}, \hat{\sigma}^2)$ is for the true distribution $\Gamma_{\text{SH},-0}|\Gamma_{\text{SH},-1}, \Gamma_{\text{SH},r} \sim \mathcal{N}(m, \sigma^2)$. As we shall see, we will utilize this analysis in such a way that it serves as a good proxy for the general $n$ point analysis. Specifically, we will utilize it to obtain bounds on the ball radius as well as on the maximum allowed curvature of a path in Section 4.2.2 and Section 4.2.2 respectively. Let $d_1 = \|q_0 - q_1\|$, $d_r = \|q_0 - q_r\|$, and $d_{1r} = \|q_1 - q_r\|$, as shown in Fig. 4.5 (top), where $q_r$ is the location of the general point. Moreover, $d_1 = \Delta d$.

The following lemma characterizes the mean and standard deviation of the KL divergence between the true distribution and its approximation for the 3 point analysis.

**Lemma 4.5** *The mean and standard deviation of the KL divergence between the true distribution $\mathcal{N}(m, \sigma^2)$ and its approximation $\mathcal{N}(\hat{m}, \hat{\sigma}^2)$ for the 3 point analysis of Fig. 4.5 (top) is given as*

$$m_{KL} = -\frac{1}{2} \log_e \left( 1 - \frac{\sigma^2_{\Delta m}}{\hat{\sigma}^2} \right),$$
$$\sigma_{KL} = \frac{\sigma^2_{\Delta m}}{\sqrt{2}\hat{\sigma}^2},$$

*where*

$$\sigma^2_{\Delta m} = \sigma^2_{SH} \frac{\left(e^{-d_r/\beta_{SH}} - e^{-(d_1+d_{1r})/\beta_{SH}}\right)^2}{1 - e^{-2d_{1r}/\beta_{SH}}}.$$

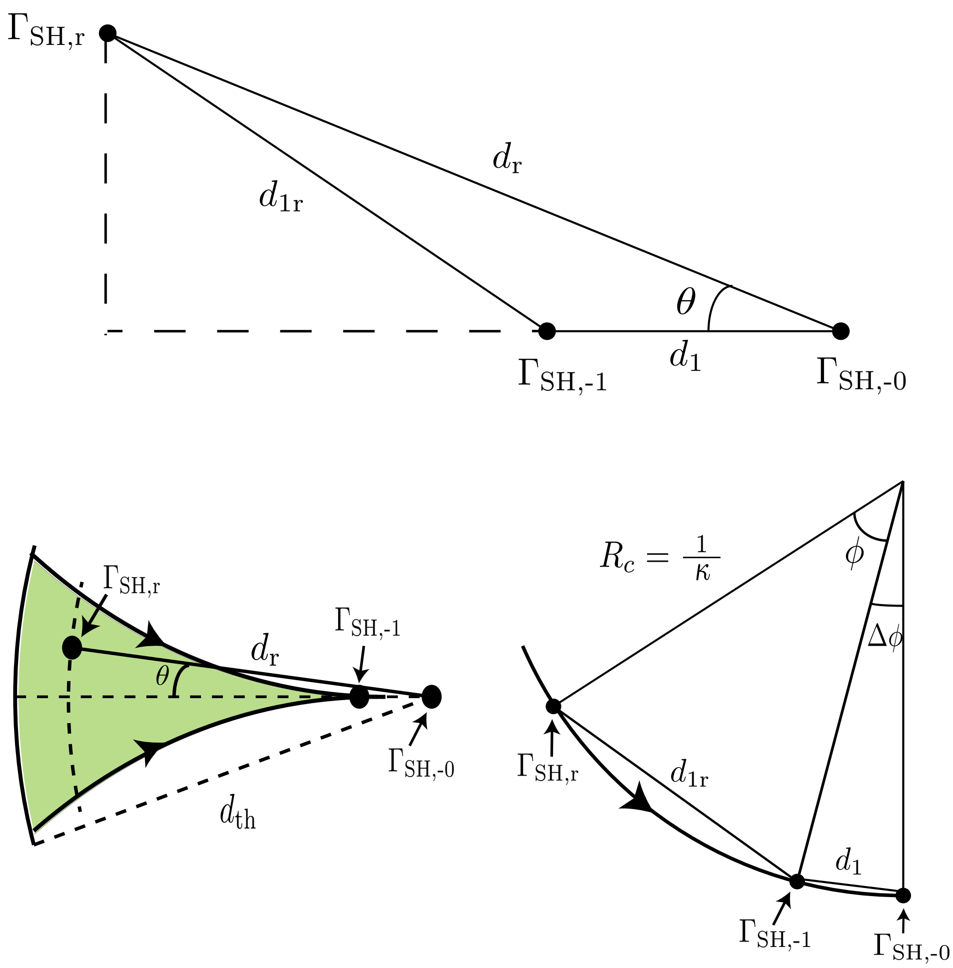


Figure 4.5: 3 points analysis (top) for a general case, (bottom left) for a path with maximum curvature $\kappa$ that satisfies $\kappa < 1/d_{\mathrm{th}}$, and (bottom right) along a path with a constant curvature.

*Proof:* See Appendix B.2 for the proof. ■

Any point on the path can belong to three possible regions: 1) the shaded region within the ball of Fig. 4.4, 2) within the ball but outside the shaded region, and 3) outside the ball. If the path is $d_{\mathrm{th}}$-loop-free, then no point of the path lies within region 2 (i.e., within the ball but outside the shaded region). We next characterize the minimum ball radius and the maximum allowed curvature of a path such that the impact of any point ($\Gamma_{\mathrm{SH},r}$) in region 1 and 3 on the estimate at the center of the ball is negligible.

### Ball Radius

We next utilize our analysis to determine the ball radius $d_{\text{th}}$. We wish to select the minimum $d_{\text{th}}$ such that the impact of any point outside the ball on the approximation is within the tolerable KL divergence parameters $\epsilon_m$ and $\epsilon_\sigma$, i.e., the KL divergence between the true and the approximating distribution (in the 3 point analysis) satisfies $m_{KL} \leq \epsilon_m$ and $\sigma_{KL} \leq \epsilon_\sigma$.

The following lemma characterizes what the minimum ball radius $d_{\text{th}}$ should be.

**Lemma 4.6** *The minimum ball radius $d_{th}$ such that any point outside the ball satisfies the maximum tolerable KL divergence parameters $\epsilon_m$ and $\epsilon_\sigma$ for the 3 point analysis, is given by*

$$d_{th} = \frac{\beta_{SH}}{2} \log_e \left( \varrho^2 + \frac{1 - \varrho^2}{\epsilon_d} \right),$$

*where $\varrho = e^{-\Delta d / \beta_{SH}}$ and $\epsilon_d = \min\left\{1 - e^{-2\epsilon_m}, \sqrt{2}\epsilon_\sigma\right\}$.*

*Proof:* See Appendix B.3 for the proof. ■

### Curvature Constraint

We next utilize the 3 point analysis to determine the maximum curvature of a path such that it is approximately-Markovian, i.e., it satisfies the KL divergence constraint $m_{KL} \leq \epsilon_m$ and $\sigma_{KL} \leq \epsilon_\sigma$.

Consider the scenario in Fig. 4.5 (bottom left). For a given maximum curvature $\kappa$, any valid point of the path must lie within the shaded region of the figure, where the boundary corresponds to circular paths with curvature $\kappa$. We wish to find the maximum allowed curvature such that the impact of any point within the shaded region on the approximation is within the tolerable KL divergence parameters $\epsilon_m$ and $\epsilon_\sigma$, i.e., the KL divergence between the true and the approximating distribution (in the 3 point analysis) satisfies $m_{KL} \leq \epsilon_m$ and $\sigma_{KL} \leq \epsilon_\sigma$. The following lemma characterizes this maximum allowed curvature as the solution of an optimization problem.

**Lemma 4.7** *The maximum allowed curvature $\kappa_{th}$ such that any past point on the path within the ball of radius $d_{th}$ satisfies the maximum tolerable KL divergence parameters $\epsilon_m$ and $\epsilon_\sigma$ for the 3 point analysis, is the solution to the following optimization problem:*

$$\begin{aligned} &\textit{maximize} && \kappa \\ &\textit{subject to} && \max_{\phi: 0<\phi\leq\psi_{cons}(\kappa)} \psi_{opt}(\kappa,\phi) \leq \epsilon_d \\ & && \kappa < 1/d_{th}, \end{aligned} \tag{4.12}$$

*where*

$$\psi_{opt}(\kappa,\phi) = \frac{\left(e^{-\frac{2}{\kappa\beta_{SH}}\sin(\frac{\phi+\Delta\phi}{2})} - \varrho e^{-\frac{2}{\kappa\beta_{SH}}\sin(\frac{\phi}{2})}\right)^2}{(1-e^{-\frac{4}{\kappa\beta_{SH}}\sin(\frac{\phi}{2})})(1-\varrho^2)},$$

*and* $\psi_{cons}(\kappa) = 2\sin^{-1}(\frac{\kappa d_{th}}{2}) - \Delta\phi$, $\Delta\phi = 2\sin^{-1}(\frac{\kappa\Delta d}{2})$, $\varrho = e^{-\Delta d/\beta_{SH}}$, $\epsilon_d = \min\left\{1-e^{-2\epsilon_m}, \sqrt{2}\epsilon_\sigma\right\}$.

*Proof:* See Appendix B.4 for the proof. ■

**Remark 4.6** *Ideally, we would have preferred to use the KL divergence between the approximation and the true distribution where we condition on all the past points on the path within the ball radius, as opposed to using just the point with the maximal impact. However, such an analysis does not lend itself to a neat characterization of the maximum allowed curvature. Through simulations, we have seen that the 3 point analysis, as described in Lemma 4.7, serves as a good proxy for the n past points case on a circular path (which has a maximum curvature everywhere for a given $\kappa$). For instance, for parameters*

*$\kappa = 1/15$, $\Delta d = 0.1$ and $\beta_{SH} = 5$ m, the KL divergence mean and standard deviation when considering all the past points of the path within the ball are $m_{KL} = 6 \times 10^{-7}$ and $\sigma_{KL} = 9 \times 10^{-7}$ respectively. This is comparable to the values $m_{KL} = 3 \times 10^{-7}$ and $\sigma_{KL} = 5 \times 10^{-7}$ obtained for the 3 point analysis from Lemma 4.7.*

Finally, we put together all our results to provide sufficient conditions for an approximately-Markovian path.

**Lemma 4.8 (Approximately-Markovian Path)** *Let $r(b) = (x(b), y(b))$ be a path parameterized by its arc length. The path is approximately-Markovian for given maximum tolerable KL divergence parameters $\epsilon_m$ and $\epsilon_\sigma$ for the 3 point analysis, if it satisfies the following conditions:*

1. *$r(b)$ is $d_{th}$-loop-free for ball radius $d_{th}$ (as characterized by Lemma 4.3),*
2. *curvature $\kappa(b) < \kappa_{th}$ for all $b$,*

*where $d_{th}$ and $\kappa_{th}$ are obtained from Lemma 4.6 and Lemma 4.7 respectively.*

Consider a given path. For a given $\epsilon_m$ and $\epsilon_\sigma$, we can check if the path satisfies the conditions of Lemma 4.8. If it does, we can then directly use the results of Section 4.2.1 to obtain the PDF of the FPD for the path. Note that even if the path does not satisfy the conditions, the path may still be approximately-Markovian as the conditions of Lemma 4.8 are sufficient conditions.

## 4.3 Characterizing FPD Considering Multipath

The previous section analyzed the FPD to the connectivity threshold when the multipath component was ignored. In this section, we show how to derive the FPD density in the presence of the multipath fading component, and for the most general channel model of $\Gamma(d) = \gamma_{\mathrm{PL}}(d) + \Gamma_{\mathrm{SH}}(d) + \Gamma_{\mathrm{MP}}(d)$. We begin by analyzing straight paths in Section 4.3.1, where we derive the PDF of the FPD using a recursive formulation. We then extend our analysis to a larger space of paths in Section 4.3.2.

### 4.3.1 Straight Paths: A Recursive Characterization

We first characterize the PDF of the distance traveled until connectivity for straight paths. We consider the scenario described in Section 4.2.1, where a robot situated at a distance $d_{\mathrm{src}}$ from a remote operator to which it needs to be connected, moves in a straight path in the direction specified by the angle $\theta_{\mathrm{src}}$, as shown in Fig. 4.1. $\Gamma(d)$ represents the channel power when the robot is at distance $d$ along direction $\theta_{\mathrm{src}}$, as marked in Fig. 4.1.

Recall that we define connectivity as the event where $\Gamma(d) \geq \gamma_{\mathrm{th}}$. The connectivity requirement is then given as $\Gamma(d) = \gamma_{\mathrm{PL}}(d) + \Gamma_{\mathrm{SH}}(d) + \Gamma_{\mathrm{MP}}(d) \geq \gamma_{\mathrm{th}}$, considering all the channel components. In this case, the approach of Section 4.2.1 is not applicable anymore as we no longer deal with a Markov process. Even if the multipath component was taken to be a Gauss-Markov process (which could be a valid model for some environments [71]), the resultant channel power would not be Markovian, as can be verified from Lemma 4.1. In this section, we assume that the robot measures the channel along the chosen straight path in discrete steps of size $\Delta d$. We assume that $\Delta d$ is such that the multipath random variable is uncorrelated at the distance $\Delta d$ apart (this is a realistic assumption as multipath decorrelates fast [3]). We then index the channel power and shadowing components accordingly, i.e., let $\Gamma_k = \Gamma(k\Delta d)$ and $\Gamma_{\mathrm{SH},k} = \Gamma_{\mathrm{SH}}(k\Delta d)$. The probability of failure of connectivity at the end of $N$ steps (given the initial failure of connectivity) can then be written as

$$\Pr\left(\Gamma_1, \Gamma_2, \cdots, \Gamma_N < \gamma_{\text{th}} | \Gamma_0 < \gamma_{\text{th}}\right) = \int_{\gamma_1, \cdots, \gamma_N < \gamma_{\text{th}}} \cdots \int p(\gamma_1, \cdots, \gamma_N | \Gamma_0 < \gamma_{\text{th}}) \mathrm{d}\gamma_1 \cdots \mathrm{d}\gamma_N, \tag{4.13}$$

where $p(\gamma_1, \cdots, \gamma_N | \Gamma_0 < \gamma_{\text{th}})$ is the conditional joint density function of $\Gamma_1, \cdots, \Gamma_N$. Consider the computation of this integral, which is an integration in an $N$ dimensional space. If we discretize the domain of $\Gamma_k$ into $M$ parts, then a direct computation of the FPD for upto $N$ steps would have a computational complexity of $O(NM^N)$, which is infeasible for high values of $M$ and $N$. Instead, we show how this can be solved efficiently through a recursive integral computation in $O(NM\log(M))$. In contrast, our previously proposed dynamic programming approach of [81] had a computational complexity of $O(N^2M^2)$.

As mentioned before, the robot measures the channel in discrete steps of size $\Delta d$. Let $d_k = k\Delta d$ denote the distance when $k$ steps are taken. Then, it can be shown, using (4.2), that the shadowing component is an autoregressive AR(1) process, the continuous analogue of which is the Ornstein-Uhlenbeck process (note that the shadowing component is Markovian):

$$\Gamma_{\text{SH},k+1} = \varrho\Gamma_{\text{SH},k} + \sigma_{\text{SH}}\sqrt{1-\varrho^2}Z_k,$$

where $\varrho = e^{-\Delta d/\beta_{\text{SH}}}$ and $Z_k$ are i.i.d. with a standard normal distribution. The conditional random variable $\Gamma_{\text{SH},k+1}|\gamma_{\text{SH},k}$ is thus a Gaussian random variable with mean $\varrho\gamma_{\text{SH},k}$ and variance $\sigma_{\text{SH}}^2(1-\varrho^2)$.

Note that the desired probability of (4.13) can be expressed as

$$\Pr\left(\Gamma_1, \cdots, \Gamma_N < \gamma_{\text{th}} | \Gamma_0 < \gamma_{\text{th}}\right) = \frac{\Pr\left(\Gamma_0, \cdots, \Gamma_N < \gamma_{\text{th}}\right)}{\Pr\left(\Gamma_0 < \gamma_{\text{th}}\right)}. \tag{4.14}$$

We next show how to compute $\Pr(\Gamma_0, \Gamma_1, \cdots \Gamma_N < \gamma_{\text{th}})$ via a recursive characterization. This is inspired in part by the calculation of orthant probabilities for auto-regressive sequences in [99]. Define the set of functions $\Omega_k$, as follows:

$$\Omega_k(\gamma_{\text{SH},k}) = \int_{\gamma_{\text{MP},k}=-\infty}^{\gamma_{\text{th}}-\gamma_{\text{PL}}(d_k)-\gamma_{\text{SH},k}} \underset{S_{k-1}}{\int \cdots \int} p(\gamma_{\text{SH},0}, \gamma_{\text{MP},0}, \cdots, \gamma_{\text{SH},k}, \gamma_{\text{MP},k})$$
$$\times \, \mathrm{d}\gamma_{\text{SH},0} \mathrm{d}\gamma_{\text{MP},0} \cdots \mathrm{d}\gamma_{\text{SH},k-1} \mathrm{d}\gamma_{\text{MP},k-1} \mathrm{d}\gamma_{\text{MP},k} \tag{4.15}$$

where $S_{k-1} = \cap_{i=0}^{k-1}\{\gamma_{\text{SH},i}, \gamma_{\text{MP},i} : \gamma_{\text{PL}}(d_i) + \gamma_{\text{SH},i} + \gamma_{\text{MP},i} < \gamma_{\text{th}}\}$ and $p(\gamma_{\text{SH},0}, \gamma_{\text{MP},0}, \cdots, \gamma_{\text{SH},k}, \gamma_{\text{MP},k})$ is the joint density of $\Gamma_{\text{SH},0}, \Gamma_{\text{MP},0}, \cdots, \Gamma_{\text{SH},k}, \Gamma_{\text{MP},k}$. Note that

$$\Pr\left(\Gamma_0, \Gamma_1, \cdots, \Gamma_N < \gamma_{\text{th}}\right) = \underset{S^N}{\int \cdots \int} p(\gamma_{\text{SH},0}, \gamma_{\text{MP},0}, \cdots, \gamma_{\text{SH},N}, \gamma_{\text{MP},N})$$
$$\times \, \mathrm{d}\gamma_{\text{SH},0} \mathrm{d}\gamma_{\text{MP},0} \cdots \mathrm{d}\gamma_{\text{SH},N} \mathrm{d}\gamma_{\text{MP},N}$$
$$= \int_{\gamma_{\text{SH},N}=-\infty}^{\infty} \Omega_N(\gamma_{\text{SH},N}) \mathrm{d}\gamma_{\text{SH},N}. \tag{4.16}$$

In the following lemma we show how to compute $\Omega_k(\gamma_{\text{SH},k})$ recursively.

**Lemma 4.9** *The functions $\Omega_k$, for $k = 1, \cdots, N$, of (4.15) can be computed by the recursion:*

$$\Omega_{k+1}(\gamma_{SH,k+1}) = F_{MP}(\gamma_{th} - \gamma_{PL}(d_{k+1}) - \gamma_{SH,k+1}) \frac{1}{\varrho} \int_{u=-\infty}^{\infty} \varphi\left(\frac{\gamma_{SH,k+1} - u}{\sigma_{SH}\sqrt{1-\varrho}}\right) \Omega_k(\frac{u}{\varrho}) \mathrm{d}u,$$

*initialized with*

$$\Omega_0(\gamma_{SH,0}) = F_{MP}(\gamma_{th} - \gamma_{PL}(0) - \gamma_{SH,0})\varphi\left(\frac{\gamma_{SH,0}}{\sigma_{SH}}\right),$$

*where $F_{MP}(.)$ is the CDF of the multipath random variable $\Gamma_{MP}$ and $\varphi(.)$ is the standard Gaussian density function.*

*Proof:* It can be seen that this clearly holds for $k = 0$:

$$\begin{aligned}\Omega_0(\gamma_{\mathrm{SH},0}) &= \int_{\gamma_{\mathrm{MP},k}=-\infty}^{\gamma_{\mathrm{th}}-\gamma_{\mathrm{PL}}(d_0)-\gamma_{\mathrm{SH},0}} p(\gamma_{\mathrm{SH},0},\gamma_{\mathrm{MP},0})\mathrm{d}\gamma_{\mathrm{MP},0}\\ &= F_{\mathrm{MP}}(\gamma_{\mathrm{th}} - \gamma_{\mathrm{PL}}(0) - \gamma_{\mathrm{SH},0})\varphi\left(\frac{\gamma_{\mathrm{SH},0}}{\sigma_{\mathrm{SH}}}\right).\end{aligned}$$

Next, $\Omega_{k+1}(\gamma_{\mathrm{SH},k+1})$ can be expanded as

$$\begin{aligned}\Omega_{k+1}(\gamma_{\mathrm{SH},k+1}) &= \int_{-\infty}^{\gamma_{\mathrm{th,MP},k+1}} \underset{S_k}{\int\cdots\int} p(\gamma_{\mathrm{SH},0},\gamma_{\mathrm{MP},0},\cdots,\gamma_{\mathrm{SH},k+1},\gamma_{\mathrm{MP},k+1})\\ &\qquad\times \mathrm{d}\gamma_{\mathrm{SH},0}\mathrm{d}\gamma_{\mathrm{MP},0}\cdots\mathrm{d}\gamma_{\mathrm{SH},k}\mathrm{d}\gamma_{\mathrm{MP},k}\mathrm{d}\gamma_{\mathrm{MP},k+1}\\ &= \int_{-\infty}^{\gamma_{\mathrm{th,MP},k+1}} p(\gamma_{\mathrm{MP},k+1})\mathrm{d}\gamma_{\mathrm{MP},k+1}\int_{-\infty}^{\infty} p(\gamma_{\mathrm{SH},k+1}|\gamma_{\mathrm{SH},k})\\ &\quad\times\int_{-\infty}^{\gamma_{\mathrm{th,MP},k}} \underset{S_{k-1}}{\int\cdots\int} p(\gamma_{\mathrm{SH},0},\gamma_{\mathrm{MP},0},\cdots,\gamma_{\mathrm{SH},k},\gamma_{\mathrm{MP},k})\\ &\qquad\times \mathrm{d}\gamma_{\mathrm{SH},0}\mathrm{d}\gamma_{\mathrm{MP},0}\cdots\mathrm{d}\gamma_{\mathrm{SH},k-1}\mathrm{d}\gamma_{\mathrm{MP},k-1}\mathrm{d}\gamma_{\mathrm{MP},k}\\ &= F_{\mathrm{MP}}(\gamma_{\mathrm{th,MP},k+1})\int_{-\infty}^{\infty}\varphi\left(\frac{\gamma_{\mathrm{SH},k+1}-\varrho\gamma_{\mathrm{SH},k}}{\sigma_{\mathrm{SH}}\sqrt{1-\varrho}}\right)\Omega_k(\gamma_{\mathrm{SH},k})\mathrm{d}\gamma_{\mathrm{SH},k}\\ &= \frac{F_{\mathrm{MP}}(\gamma_{\mathrm{th,MP},k+1})}{\varrho}\int_{u=-\infty}^{\infty}\varphi\left(\frac{\gamma_{\mathrm{SH},k+1}-u}{\sigma_{\mathrm{SH}}\sqrt{1-\varrho}}\right)\Omega_k(\frac{u}{\varrho})\mathrm{d}u,\end{aligned}$$

where $\gamma_{\mathrm{th,MP},k} = \gamma_{\mathrm{th}} - \gamma_{\mathrm{PL}}(d) - \gamma_{\mathrm{SH},k}$. ■

**Remark 4.7** *Note that the recursive integral in Lemma 4.9 is in the form of a convolution. This can be computed efficiently using the Fast Fourier transform.*

Using Lemma 4.9, we can compute $\Pr(\Gamma_0, \Gamma_1, \cdots, \Gamma_N < \gamma_{\text{th}})$ as shown in (4.16), which in turn is used to compute $\Pr(\Gamma_1, \cdots \Gamma_N < \gamma_{\text{th}} | \Gamma_0 < \gamma_{\text{th}})$ via (4.14).

Next, we use this result to calculate the FPD probability. Let $\mathcal{K} = \min_{k=1,2,\cdots} \{k : \Gamma_k \geq \gamma_{\text{th}}, \Gamma_0 < \gamma_{\text{th}}\}$ be the random variable which denotes the upcrossing first passage step to connectivity given that $\Gamma_0$ is restricted to lie below $\gamma_{\text{th}}$. Then,

$$\begin{aligned}\Pr(\mathcal{K} = k) &= \Pr(\Gamma_1, \cdots \Gamma_{k-1} < \gamma_{\text{th}}, \Gamma_k \geq \gamma_{\text{th}} | \Gamma_0 < \gamma_{\text{th}}) \\ &= \Pr(\Gamma_1, \cdots \Gamma_{k-1} < \gamma_{\text{th}} | \Gamma_0 < \gamma_{\text{th}}) - \Pr(\Gamma_1, \cdots \Gamma_k < \gamma_{\text{th}} | \Gamma_0 < \gamma_{\text{th}}),\end{aligned}$$

where both terms on the right hand side can be obtained from our recursive characterization using Lemma 4.9.

### 4.3.2 Approximately-Markovian Paths

In this section, we characterize the space of paths (beyond straight paths) for which we can characterize the statistics of the distance traveled until connectivity. As we saw in Section 4.3.1, the recursive characterization of Lemma 4.9 depends on the channel shadowing power being a Markov process. Specifically, the proof of Lemma 4.9 requires that $p(\gamma_{\text{SH},-0} | \gamma_{\text{SH},-1}, \gamma_{\text{SH},-2}, \cdots) = p(\gamma_{\text{SH},-0} | \gamma_{\text{SH},-1})$, where $\Gamma_{\text{SH},-0}$ is the shadowing power at the current location and $\Gamma_{\text{SH},-1}, \Gamma_{\text{SH},-2}, \cdots$ are the channel shadowing power at previously visited points, as shown in Fig. 4.2 (top). We can then directly use the tools and strategies developed in Section 4.2.2 to characterize the space of paths that are approximately-Markovian. We then obtain the statistics of the FPD for these paths using Lemma 4.9.

**Remark 4.8 (Computational complexity)** *A natural question that arises is: why not use the results of Section 4.3.1 to tackle the case without considering multipath of Section 4.2.1? We next address this. As discussed in Section 4.3.1, the computation cost of Lemma 4.9 for upto $N$ steps is $O(NM\log(M))$. In contrast, the computational cost of Theorem 4.2 for the case without considering multipath, for upto $N$ steps, is $O(N^2)$. Since $M >> N$, the stochastic differential equation approach is more computationally efficient. Moreover, the characterization of the $\epsilon$-upcrossing FPD of Section 4.2.1 can be used for analytical purposes.*

## 4.4 Numerical Results based on Real Channel Data

In this section, we validate the derivations of Sections 4.2 and 4.3 in a simulation environment with real channel parameters. We also highlight interesting trends of the FPD statistics as a function of the channel parameters. The channel is generated using the channel model described in Section 4.1.1, with parameters obtained from real channel measurements in downtown San Francisco [73]: $n_{\mathrm{PL}} = 4.2$, $\sigma^2_{\mathrm{SH}} = 8.41$ and $\beta_{\mathrm{SH}} = 12.92$ m. We impose a minimum required received SNR of 20 dB, the noise power is taken to be a realistic $-100$ dBmW, and the transmit power is taken to be 30 dBmW, which results in a channel power connectivity threshold of $\gamma_{\mathrm{th}} = -110$ dB. We furthermore take the upcrossing FDP constant to be $\epsilon = 0.1$ in the simulation results.

We consider a discretization step size of $\Delta d = 0.03$ m. Let the maximum tolerable KL divergence parameters be $\epsilon_m = 0.001$ and $\epsilon_\sigma = 0.001$. Then, the ball radius $d_{\mathrm{th}} = 9.5$ m and the maximum allowed curvature $\kappa_{\mathrm{th}} = 0.104$ satisfy Lemma 4.6 and Lemma 4.7 respectively. We will demonstrate the efficacy of our proposed approaches through two different paths that satisfy these constraints and are thus approximately-Markovian: 1) an archimedian spiral with equation $r_d = 11 + 5e^{\theta}$, and 2) a logarithmic spiral with equation $r_d = 11e^{0.5\theta}$, where both equations are in polar coordinates $(r_d, \theta)$. Figure 4.6 (top) shows the path and the curvature along the path of the archimedian spiral, while Fig. 4.7 (top) shows the path and the curvature along the path of the logarithmic spiral. The remote station is located at the origin as denoted in figures 4.6 (top) and 4.7 (top).

### 4.4.1 Results Without Considering Multipath

We first consider the case without multipath. Figure 4.6 (middle) shows the PDF and CDF of the upcrossing FPD for the archimedian path. Figure 4.7 (middle) shows the PDF and CDF of the upcrossing FPD for the logarithmic path. We can see that, for both paths, our theoretical derivations match the true statistics obtained via Monte Carlo simulations very well.

When considering the FPD to connectivity, it should be noted that $\mathcal{D}^{\epsilon}_{\Gamma_0}$ may not be a proper random variable, i.e.[4], $\Pr(\mathcal{D}^{\epsilon}_{\Gamma_0} < \infty) < 1$. To see how this may arise, consider the case when the robot moves away from the remote operator. In this case, it is reasonable to expect that with a finite probability the robot will never be connected to the remote operator, especially if it started far from the remote operator to begin with. Formally, let $P_{\text{conn},\infty} = \Pr(\mathcal{D}^{\epsilon}_{\Gamma_0} < \infty)$ denote the probability of connectivity as $d \to \infty$, i.e., the event that the robot will eventually be connected. Then, there could be cases where $P_{\text{conn},\infty} < 1$. Thus, to fully characterize the first passage statistics and embrace this possibility, we further consider $P_{\text{conn},\infty}$ as well.

Consider a robot moving along a straight path as shown in Fig. 4.1b. The robot travels along this path until it gets connected or until it travels a 1000 m. Figure 4.8 shows the contour plots of the probability of connectivity $P_{\text{conn},\infty}$ and the expected distance traveled, as a function of $d_{\text{src}}$ and $\theta_{\text{src}}$. These plots are representative of the connection statistics for a given space and given set of channel parameters. As expected, with an

[4]This issue also arises in the general literature on first passage [100].

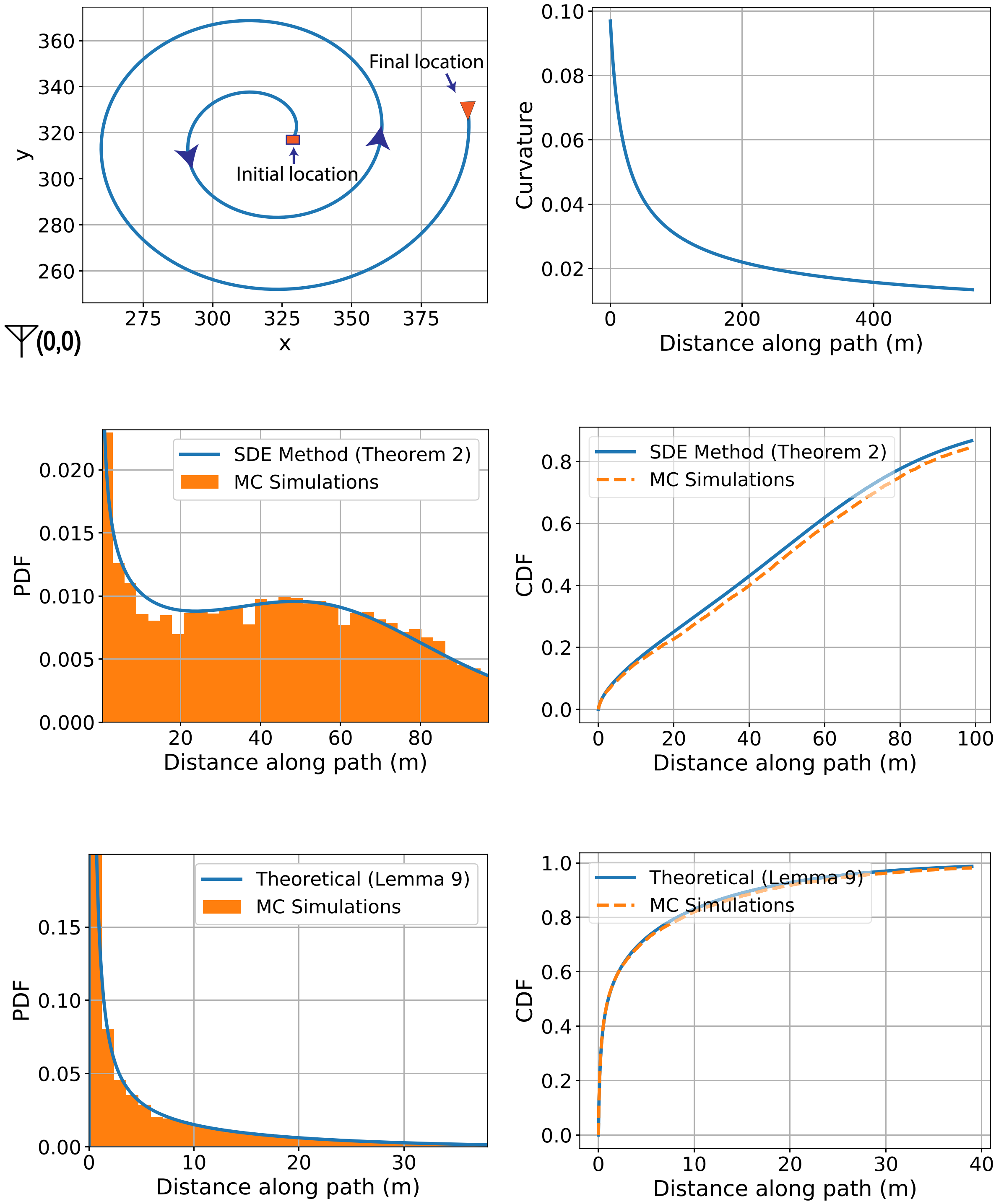


Figure 4.6: (top) Archimedian spiral as the path of the robot and curvature along the archimedian spiral, (middle) PDF and CDF of upcrossing FPD without considering multipath, and (bottom) PDF and CDF of upcrossing FPD when including multipath.

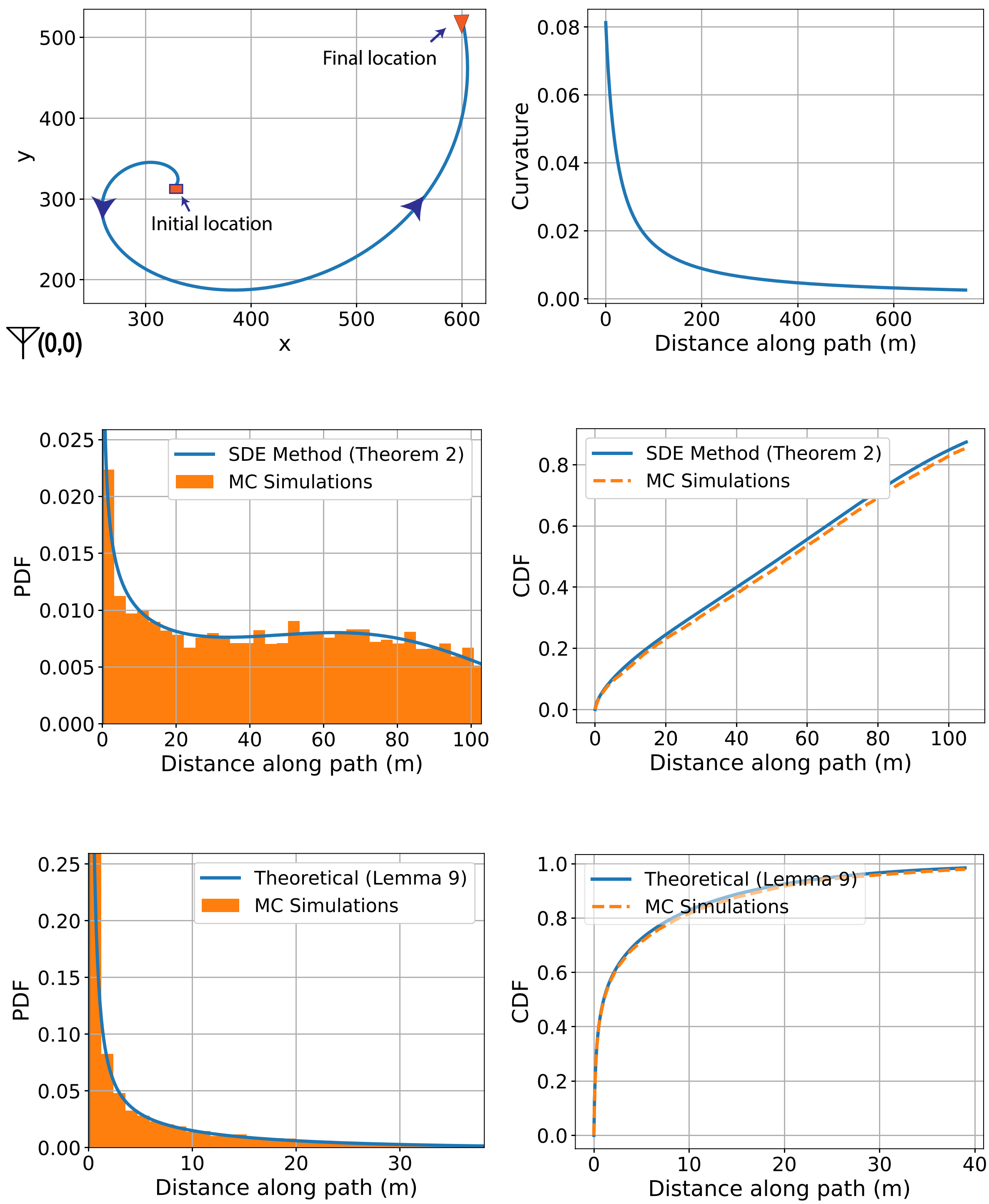


Figure 4.7: (top) Logarithmic spiral as the path of the robot and curvature along the logarithmic spiral, (middle) PDF and CDF of upcrossing FPD without considering multipath, and (bottom) PDF and CDF of upcrossing FPD when including multipath.

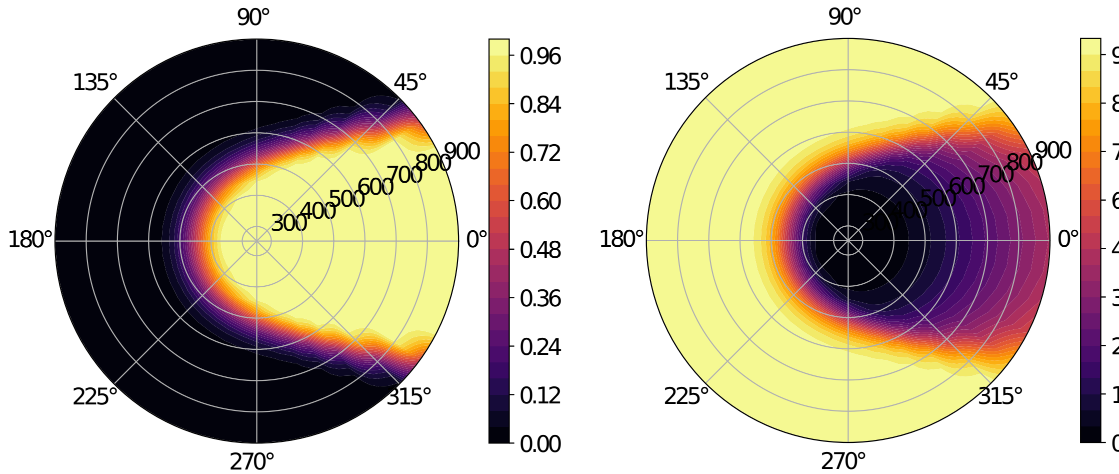


Figure 4.8: Polar contour plot of (left) eventual probability of connectivity $P_{\text{conn},\infty}$, and (right) expected distance traveled without considering multipath, as a function of $d_{\text{src}}$ and $\theta_{\text{src}}$.

increase in $\theta_{\text{src}}$, i.e., angling away from the remote operator, $P_{\text{conn},\infty}$ decreases (i.e., there is a higher non-zero probability that the robot will never get connected) and the expected distance traveled by the robot increases. Note that for every $d_{\text{src}}$, there is a positive angle $\phi > 0$ for which for angle $\theta_{\text{src}} \leq \phi$, connectivity is a sure event, i.e., $P_{\text{conn},\infty} = 1$. Furthermore, as $d_{\text{src}}$ increases, $\phi$ decreases as can be seen from Fig. 4.8, i.e., the robot needs to stay closer to $\theta_{\text{src}} = 0$ to guarantee $P_{\text{conn},\infty} = 1$. Moreover, as expected, there is an increase in the expected distance traveled by the robot with increasing distance from the source $d_{\text{src}}$.

### 4.4.2 Results When Including Multipath

Next, consider the case where multipath of the environment can not be neglected. We then simulate the multipath fading as an uncorrelated Rician random variable. Rician distribution is a common distribution for characterizing multipath [70] and is given by

$$f_{\text{ric}}(z) = (1 + K_{\text{ric}})e^{-K_{\text{ric}}-(1+K_{\text{ric}})z} I_{B,0}\left(2\sqrt{zK_{\text{ric}}(1 + K_{\text{ric}})}\right),$$

where $I_{B,0}(.)$ is the modified $0^{\text{th}}$ order Bessel function and the parameter $K_{\text{ric}}$ is the ratio of the power in the line of sight component to the power in the non-line of sight components of the channel. We use the rician parameter $K_{\text{ric}} = 1.59$, which we obtain from the real channel measurements in downtown San Francisco. We further assume that the multipath component gets uncorrelated at our discretization interval of 0.03 m, which is a reasonable assumption in many cases [3].

Figure 4.6 (bottom) shows the PDF and CDF of the upcrossing FPD for the archimedian path. Figure 4.7 (bottom) shows the PDF and CDF of the upcrossing FPD for the logarithmic path. The histogram obtained via Monte Carlo simulations is also plotted for comparison. It can be seen that in the case of both paths our derivations match the true statistics very well.

Finally, different environments will have different underlying channel parameters. Thus, we next consider the impact of the underlying channel parameters on the FPD. Figure 4.9 (top) shows the expected distance traveled as a function of the shadowing decorrelation distance ($\beta_{\text{SH}}$) and the shadowing variance ($\sigma^2_{\text{SH}}$) respectively when $d_{\text{src}} =$ 550 m and $\theta_{\text{src}} = 0$ rad, along a straight path. Increasing the shadowing power directly increases the spatial variance of the channel power. Thus, with a higher probability, $\Gamma(d)$ stumbles upon the connectivity threshold earlier, resulting in a smaller FPD, as can be seen. An increase in the decorrelation distance, on the other hand, implies a greater spatial correlation of the channel power and decreases the spatial variation. Thus, we observe that the expected traveled distance increases when increasing the decorrelation distance. Figure 4.9 (bottom) shows the expected distance until connectivity as a function of $K_{\text{ric}}$ of multipath. For large values of $K_{\text{ric}}$, the line of sight component dominates and results in a more deterministic multipath term. Decreasing $K_{\text{ric}}$, on the other hand, results in an increase in the variance of the multipath component, thus increasing the randomness of the channel. Thus as $K_{\text{ric}}$ decreases, $\Gamma(d)$ would cross the connectivity

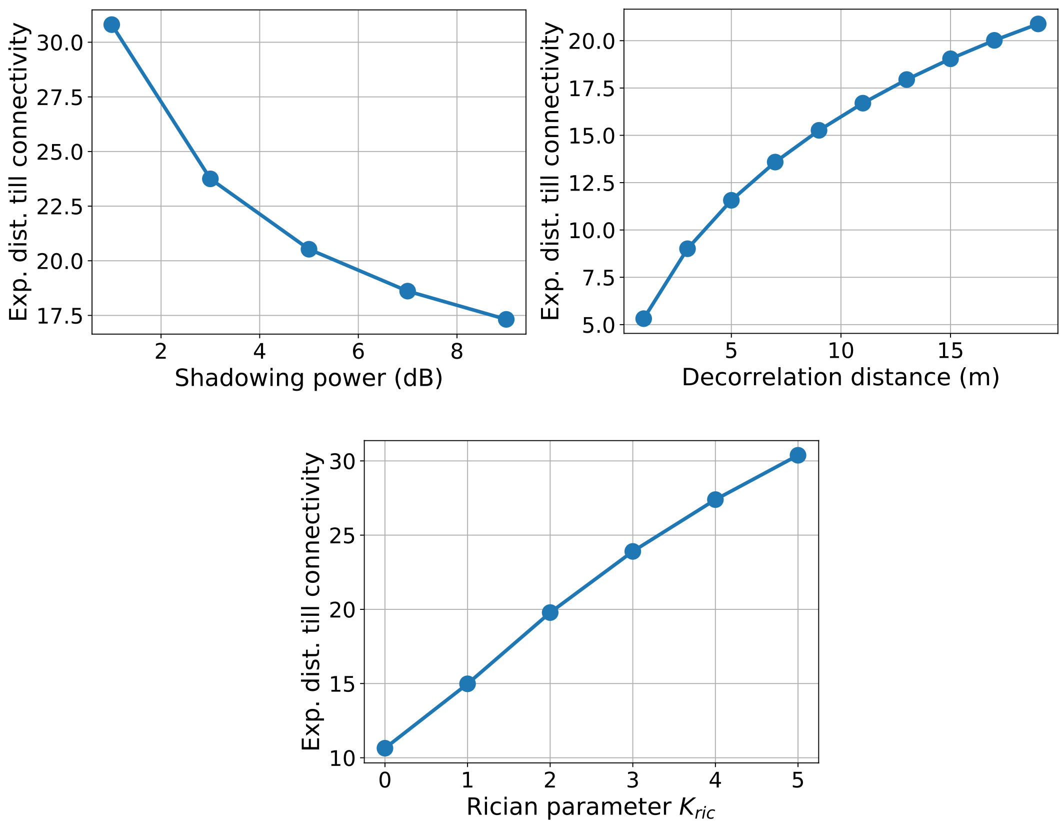


Figure 4.9: Expected distance until connectivity (with multipath) as a function of the (top left) shadowing power, (top right) shadowing decorrelation distance and (bottom) rician parameter $K_{\text{ric}}$, for the case of a straight path with $d_{\text{src}} = 550$ m and $\theta_{\text{src}} = 0$ rad.

threshold earlier with a higher probability (due to the increase in channel randomness), resulting in a smaller expected distance traveled.

In this chapter, we characterized the distance traveled by a mobile robot until it gets connected *along a given path*. In the next chapter, we actively *plan the path* of the robot, such that we minimize the expected traveled distance until the robot gets connected.

# Chapter 5

# Path Planning for Minimizing the Expected Cost until Success/Connectivity

In this chapter, we consider the scenario where an unmanned vehicle needs to get connected to a remote station (or another robot). More specifically, we consider the case where an unmanned vehicle is not connected in its initial location and needs to incur motion energy to find a connected spot. We are then interested in designing an energy efficient path for the robot that can guarantee connectivity in realistic channel environments experiencing path loss, shadowing, and multipath fading. More precisely, we consider the problem of planning the path of a robot in order to find a connected spot while minimizing the expected traveled distance.

We pose this in the framework of a general path planning problem of a robot on a graph with edge costs, and where each node has a Boolean value of success or failure (with respect to some task) with a given probability. The objective is to plan a path for the robot on the graph that minimizes the expected cost until success. As discussed in Section 1.3, this general problem has several possible applications. In the rest of

Parts of this chapter have appeared in our conference submission [49], ©[2017] IEEE, and our submitted journal [101].

the chapter we will primarily analyze and obtain results for the general graph-theoretic problem. We will apply the results we obtain to the specific scenario of path planning for a connectivity seeking robot in Section 5.5.2, where the robot uses the probabilistic prediction framework of Section 2.2 to evaluate the probability of connectivity success at each node on the graph, and then uses our proposed framework of this section to design a minimum average cost path to a connected spot.

In this chapter, it is our goal to bring a foundational understanding to this general graph-theoretic problem. We start by showing that the problem of interest, i.e., minimizing the expected cost until success, can be posed as an infinite horizon Markov Decision Process (MDP) and solved optimally, but with an exponential space complexity. We then formally prove its NP-hardness. To address the space complexity, we then propose an asymptotically $\epsilon$-suboptimal (i.e., within $\epsilon$ of the optimal solution value) path planner for this problem, using a game-theoretic framework. We further show how it is possible to solve this problem very quickly by proposing two sub-optimal but non-myopic approaches. Our proposed approaches provide a variety of tools that can be suitable for applications with different needs.

The rest of the chapter is organized as follows. In Section 5.1, we formally introduce the problem of interest and show how to optimally solve it by formulating it in an infinite horizon MDP framework as a stochastic shortest path (SSP) problem. As we shall see, however, the state space requirement for this formulation is exponential in the number of nodes in the graph. In Section 5.2, we formally prove our problem to be NP-hard, demonstrating that the exponential complexity result of the MDP formulation is not specific to it. In Section 5.3, we propose an asymptotically $\epsilon$-suboptimal path planner and in Section 5.4 we propose two suboptimal but non-myopic and fast path planners to tackle the problem. Finally, in Section 5.5, we confirm the efficiency of our approaches with numerical results in two different scenarios: a rover on Mars searching for an object for scientific studies, and a robot looking for a connected spot to a remote station (with real data from downtown San Francisco). Our numerical results show a considerable performance improvement over existing state-of-the-art approaches.

## 5.1 Problem Formulation

In this section, we formally define the problem of interest, which we refer to as the Min-Exp-Cost-Path problem. We next show that we can find the optimal solution of Min-Exp-Cost-Path by formulating it as an infinite horizon MDP with an absorbing state, a formulation known in the stochastic dynamic programming literature as the *stochastic shortest path* problem [102]. However, we show that this results in a state space requirement that is exponential in the number of nodes of the graph, implying that it is only feasible for small graphs and not scalable when increasing the size of the graph.

### 5.1.1 Min-Exp-Cost-Path Problem

Consider an undirected connected finite graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ denotes the set of nodes and $\mathcal{E}$ denotes the set of edges. Let $p_v \in [0, 1]$ be the probability of success at node $v \in \mathcal{V}$ and let $l_{uv} > 0$ denote the cost of traversing edge $(u, v) \in \mathcal{E}$. We assume that the success or failure of a node is independent of the success or failure of the other nodes in the graph. Let $v_s \in \mathcal{V}$ denote the starting node. The objective is to produce a path starting from node $v_s$ that *minimizes the expected cost incurred until success*. In other words, the average cost until success on the optimal path is smaller than the average cost on any other possible path on the graph. Note that the robot may only traverse part of the entire path produced by its planning, as its planning is based on a probabilistic prior knowledge and success may occur at any node along the path.

For the expected cost until success of a path to be well defined, the probability of failure after traversing the entire path must be 0. This implies that the final node of the path must be one where success is guaranteed, i.e., a $v$ such that $p_v = 1$. We call such a node a *terminal* node and let $T = \{v \in \mathcal{V} : p_v = 1\}$ denote the set of terminal nodes. We assume that the set $T$ is non-empty in this subsection. We refer to this as the *Min-Exp-Cost-Path* problem. Fig. 5.1 shows a toy example along with a feasible solution path. In Section 5.2.1, we will extend our discussion to the setting when the the set $T$ is empty.

We next characterize the expected cost for paths where nodes are not revisited, i.e., simple paths, and then generalize it to all possible paths. Let the path, $\mathcal{P} = (v_1, v_2, \cdots, v_m = v_t)$, be a sequence of $m$ nodes such that no node is revisited, i.e., $v_i \neq v_j,\ \forall i \neq j$, and which ends at a terminal node $v_t \in T$. Let $C(\mathcal{P}, i)$ represent the expected cost of the path from node $\mathcal{P}[i] = v_i$ onward. $C(\mathcal{P}, 1)$ is then given as

$$\begin{aligned}
C(\mathcal{P}, 1) &= p_{v_1} \times 0 + (1 - p_{v_1}) p_{v_2} l_{v_1 v_2} + \cdots \\
&\quad + \left[ \prod_{j \leq m-1} (1 - p_{v_j}) \right] p_{v_m} (l_{v_1 v_2} + \cdots + l_{v_{m-1} v_m}) \\
&= (1 - p_{v_1}) l_{v_1 v_2} + (1 - p_{v_1})(1 - p_{v_2}) l_{v_2 v_3} + \cdots \\
&\qquad\qquad + \left[ \prod_{j \leq m-1} (1 - p_{v_j}) \right] l_{v_{m-1} v_m} \\
&= \sum_{i=1}^{m-1} \left[ \prod_{j \leq i} (1 - p_{v_j}) \right] l_{v_i v_{i+1}}.
\end{aligned}$$

For a path which contains revisited nodes, the expected cost can then be given by

$$
\begin{aligned}
C(\mathcal{P},1) &= \sum_{i=1}^{m-1} \left[ \prod_{j\leq i: v_j \neq v_k, \forall k<j} (1-p_{v_j}) \right] l_{v_i v_{i+1}} \\
&= \sum_{e \in \mathcal{E}(\mathcal{P})} \left[ \prod_{v \in \mathcal{V}(\mathcal{P}_e)} (1-p_v) \right] l_e,
\end{aligned}
$$

where $\mathcal{E}(\mathcal{P})$ denotes the set of edges belonging to the path $\mathcal{P}$, and $\mathcal{V}(\mathcal{P}_e)$ denotes the set of vertices encountered along $\mathcal{P}$ until the edge $e \in \mathcal{E}(\mathcal{P})$. Note that $\mathcal{C}(\mathcal{P}, i)$ can be expressed recursively as

$$
C(\mathcal{P},i) = \begin{cases} (1-p_{v_i}) \left( l_{v_i v_{i+1}} + C(\mathcal{P}, i+1) \right), & \text{if } v_i \neq v_k, \forall k < i \\ l_{v_i v_{i+1}} + C(\mathcal{P}, i+1), & \text{else} \end{cases} . \tag{5.1}
$$

The Min-Exp-Cost-Path optimization can then be expressed as

$$
\begin{aligned}
\underset{\mathcal{P}}{\text{minimize}} \quad & C(\mathcal{P},1) \\
\text{subject to} \quad & \mathcal{P} \text{ is a path of } \mathcal{G} \\
& \mathcal{P}[1] = v_s \\
& \mathcal{P}[\text{end}] \in T.
\end{aligned} \tag{5.2}
$$

We next show how to optimally solve the Min-Exp-Cost-Path problem by formulating it as an infinite horizon MDP.

### 5.1.2 Optimal Solution via MDP Formulation

The stochastic shortest path problem (SSP) [102] is an infinite horizon MDP formulation, which is specified by a state space $S$, control/action constraint sets $A_s$ for $s \in S$, state transition probabilities $P_{ss'}(a_s) = \mathrm{P}\left(s_{k+1} = s' | s_k = s, a_k = a_s\right)$, an absorbing ter-

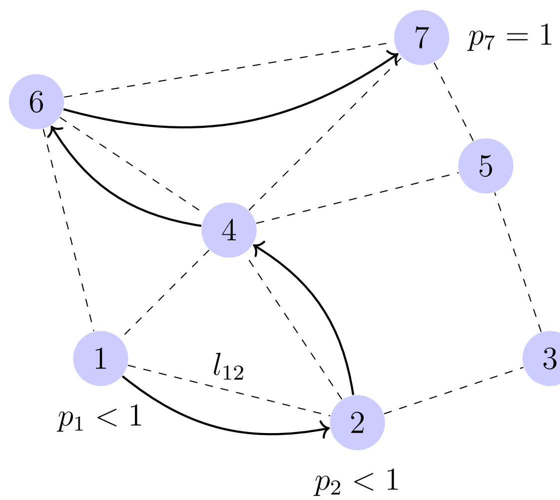


Figure 5.1: A toy example along with a feasible solution path starting from node 1.

minal state $s_t \in S$, and a cost function $g_{\text{cost}}(s, a_s)$ for $s \in S$ and $a_s \in A_s$. The goal is to obtain a policy that would lead to the terminal state $s_t$ with a probability 1 and with a minimum expected cost.

We next show that the Min-Exp-Cost-Path problem formulation of (5.2) can be posed in an SSP formulation. Utilizing the recursive expression of (5.1), we can see that the expected cost from a node, conditioned on the set of nodes already visited by the path can be expressed in terms of the expected cost from the neighboring node that the path visits next. Thus, the optimal path from a node can be expressed in terms of the optimal path from the neighboring node that the path visits next, conditioned on the set of nodes already visited. This motivates the use of a stochastic dynamic programming framework where a state is given by the current node as well as the set of nodes already visited.

More precisely, we formulate the SSP as follows. Let $\mathcal{V}' = \mathcal{V} \setminus T$ be the set of non-terminal nodes in the graph. A state of the MDP is given by $s = (v, H)$, where $v \in \mathcal{V}'$ is the current node and $H \subseteq \mathcal{V}'$ is the set of nodes already visited (keeping track of the history of the nodes visited), i.e., $u \in H$, if $u$ is visited. The state space is then given by $S = \{(v, H) : v \in \mathcal{V}', H \subseteq \mathcal{V}'\} \cup \{s_t\}$, where $s_t$ is the absorbing terminal state. In this setting, the state $s_t$ denotes the state of success. The actions/controls available at a given state is the set of neighbors of the current node, i.e., $A_s = \{u \in \mathcal{V} : (v,u) \in \mathcal{E}\}$ for $s = (v, H)$. The state transition probabilities are denoted by $P_{ss'}(u) = \mathrm{P}\left(s_{k+1} = s' | s_k = s, a_k = u\right)$ where $s, s' \in S$ and $u \in A_s$. Then, for $s = (v, H)$ and $u \in A_s$, if $v \in H$ (i.e., $v$ is revisited), we have

$$P_{ss'}(u) = \begin{cases} 1, & \text{if } s' = g_{\text{trans}}(u, H) \\ 0, & \text{else} \end{cases},$$

and if $v \notin H$, we have

$$P_{ss'}(u) = \begin{cases} 1 - p_v, & \text{if } s' = g_{\text{trans}}(u, H) \\ p_v, & \text{if } s' = s_t \\ 0, & \text{else} \end{cases},$$

where $g_{\text{trans}}(u, H) = \begin{cases} (u, H \cup \{v\}), & \text{if } u \in \mathcal{V}' \\ s_t, & \text{if } u \in T \end{cases}$. This implies that at node $v$, the robot will experience success with probability $p_v$ if $v$ has not been visited before, i.e., $v \notin H$. The terminal state $s_t$ is absorbing, i.e., $P_{s_t s_t}(u) = 1, \forall u \in A_{s_t}$. The cost $g_{\text{cost}}(s, u)$ incurred when action/control $u \in A_s$ is taken in state $s \in S$ is given by $g_{\text{cost}}\left(s = (v, H), u\right) = \begin{cases} (1 - p_v) l_{uv}, & \text{if } v \notin H \\ l_{uv}, & \text{if } v \in H \end{cases}$, representing the expected cost incurred when going from $v$ to $u$ conditioned on the set of already visited nodes $H$.

The optimal (minimum expected) cost incurred from any state $s_1$ is then given by

$$\overline{J}^*_{s_1} = \min_{\mu} \underset{\{s_k\}}{\mathbb{E}} \left[ \sum_{k=1}^{\infty} g_{\text{cost}}(s_k, \mu_{s_k}) \right],$$

where $\mu$ is a policy that prescribes what action to take/neighbor to choose at a given state, i.e., $\mu_s$ is the action to take at state $s$. The policy $\mu$, specifies which node to move to next, i.e., if at state $s$, then $\mu_s$ denotes which node to go to next. The objective is to find the optimal policy $\mu^*$ that would minimize the expected cost from any given state of the SSP formulation. Given the optimal policy $\mu^*$, we can then extract the optimal solution path of (5.2). Let $(s_1, \cdots, s_m = s_t)$ be the sequence of states such that $s_1 = (v_s, H_1 = \{\})$ and $s_{k+1} = (v^*_{k+1}, H_{k+1})$, $k = 1, \cdots, m-2$, where $v^*_{k+1} = \mu^*(s_k)$ and $H_{k+1} = H_k \cup \{v^*_k\}$. This sequence must end at $s_m = s_t$ for some finite $m$, since the expected cost is not well defined otherwise. The optimal path starting from node $v_s$ is then extracted from this solution as $\mathcal{P}^* = (v_s, v^*_2, \cdots, v^*_m)$.

In the following Lemma, we show that the optimal solution can be characterized by the Bellman equation.

**Lemma 5.1** *The optimal cost function $\overline{J}^*$ is the unique solution of the Bellman equation:*

$$\overline{J}^*_s = \min_{u \in A_s} \left[ g_{cost}(s,u) + \sum_{s' \in S \setminus \{s_t\}} P_{ss'}(u) \overline{J}^*_{s'} \right],$$

*and the optimal policy $\mu^*$ is given by*

$$\mu^*_s = \arg\min_{u \in A_s} \left[ g_{cost}(s,u) + \sum_{s' \in S \setminus \{s_t\}} P_{ss'}(u) \overline{J}^*_{s'} \right],$$

*for all $s \in S \setminus \{s_t\}$.*

*Proof:* Let $\overline{J}^\mu_s$ denote the cost of state $s$ for a policy $\mu$. We first review the definition of a *proper policy*. A policy $\mu$ is said to be proper if, when using this policy, there is a positive probability that the terminal state will be reached after at most $|S|$ stages, regardless of the initial state [102]. We next show that the MDP formulation satisfies the following properties: 1) there exists at least one proper policy, and 2) for every improper policy $\mu$, there exists at least one state with cost $\overline{J}_s^{\mu} = \infty$. We know that there exists at least one proper policy since the policy corresponding to taking the shortest path to the nearest terminal node irrespective of the history of nodes visited, is a proper policy. Moreover, since $g_{\text{cost}}(s, u) > 0$ for all $s \neq s_t$, every cycle in the state space not including the destination has strictly positive cost. This implies property 2 is true. The proof is then provided in [102]. ■

The optimal solution can then be found by the value iteration method. Given an initialization $\overline{J}_s(0)$, for all $s \in S \setminus \{s_t\}$, value iteration produces the sequence:

$$\overline{J}_s(k+1) = \min_{u \in A_s} \left[ g_{\text{cost}}(s, u) + \sum_{s' \in S \setminus \{s_t\}} P_{ss'}(u) \overline{J}_{s'}(k) \right],$$

for all $s \in S \setminus \{s_t\}$. This sequence converges to the optimal cost $\overline{J}_s^*$, for each $s \in S \setminus \{s_t\}$.

**Lemma 5.2** *When starting from $\overline{J}_s(0) = \infty$ for all $s \in S \setminus \{s_t\}$, the value iteration method yields the optimal solution after at most $|S| = |\mathcal{V}'| \times 2^{|\mathcal{V}'|} + 1$ iterations.*

*Proof:* Let $\mu^*$ be the optimal policy. Consider a directed graph with the states of the MDP as nodes which has an edge $(s, s')$ if $P_{ss'}(\mu_s^*) > 0$. We will first show that this graph is acyclic. Note that a state $s = (v, H)$, where $v \notin H$, can never be revisited regardless of the policy used, since a transition from $s$ will occur either to $s_t$ or a state with $H = H \cup \{v\}$. Then, any cycle in the directed graph corresponding to $\mu^*$ would only have states of the form $s = (v, H)$ with $v \in H$. Moreover, any state $s = (v, H)$ in the cycle cannot have a transition to state $s_t$ since $v \in H$. Thus, if there is a cycle, the cost of any state $s$ in the cycle will be $\overline{J}_s^{\mu^*} = \infty$, which results in a contradiction. The value iteration method converges in $|S|$ iterations when the graph corresponding to the optimal policy $\mu^*$ is acyclic [102]. ■

**Remark 5.1** *Each stage of the value iteration process has a computational cost of $O(|\mathcal{E}| 2^{|\mathcal{V}'|})$*

*since for each state $s = (v, H)$ there is an associated a computational cost of $O(|A_v|)$. Then, from Lemma 5.2, we can see that the overall computational cost of value iteration is $O(|\mathcal{V}'||\mathcal{E}|2^{2|\mathcal{V}'|})$, which is exponential in the number of nodes in the graph. Note, however, that the brute force approach of enumerating all paths has a much larger computational cost of $O(|\mathcal{V}'|!)$.*

The exponential space complexity prevents the stochastic shortest path formulation from providing a scalable solution for solving the problem for larger graphs. A general question then arises as to whether this high computational complexity result is a result of the Markov Decision Process formulation. In other words, can we optimally solve the Min-Exp-Cost-Path problem with a low computational complexity using an alternate method? We next show that the Min-Exp-Cost-Path problem is inherently computationally complex (NP-hard).

## 5.2 Computational Complexity

In this section, we prove that Min-Exp-Cost-Path is NP-hard. In order to do so, we first consider the extension of the Min-Exp-Cost-Path problem to the setting where there is no terminal node, which we refer to as the Min-Exp-Cost-Path-NT problem (Min-Exp-Cost-Path No Terminal node). We prove that Min-Exp-Cost-Path-NT is NP-hard, a result we then utilize to prove that Min-Exp-Cost-Path is NP-hard.

Motivated by the negative space complexity result of our MDP formulation, we then discuss a setting where we restrict ourselves to the class of *simple paths*, i.e., cycle free paths, and we refer to the minimum expected cost until success problem in this setting as the Min-Exp-Cost-Simple-Path problem. This serves as the setting for our path planning approaches of Section 5.3 and 5.4. Furthermore, we show that we can obtain a solution to the Min-Exp-Cost-Path problem from a solution of the Min-Exp-Cost-Simple-Path problem in an appropriately defined complete graph.

### 5.2.1 Min-Exp-Cost-Path-NT Problem

Consider the graph-theoretic setup of the Min-Exp-Cost-Path problem of Section 5.1.1. In this subsection, we assume that there is no terminal node, i.e., the set $T = \{v \in \mathcal{V} : p_v = 1\}$ is empty. There is thus a finite probability of failure for any path in the graph and as a result the expected cost until success is not well defined. The expected cost of a path then includes the event of failure after traversing the entire path and its associated cost. The objective in *Min-Exp-Cost-Path-NT* is to obtain a path that visits all the vertices with a non-zero probability of success, i.e., $\{v \in \mathcal{V} : p_v > 0\}$, such that the expected cost is minimized. This objective finds the minimum expected cost path among all paths that have a minimum probability of failure. More formally, the objective for Min-Exp-Cost-Path-NT is given as

$$\begin{aligned}
\underset{\mathcal{P}}{\text{minimize}} \quad & \sum_{e \in \mathcal{E}(\mathcal{P})} \left[ \prod_{v \in \mathcal{V}(\mathcal{P}_e)} (1 - p_v) \right] l_e \\
\text{subject to} \quad & \mathcal{P} \text{ is a path of } \mathcal{G} \\
& \mathcal{P}[1] = v_s \\
& \mathcal{V}(\mathcal{P}) = \{v \in \mathcal{V} : p_v > 0\},
\end{aligned} \tag{5.3}$$

where $\mathcal{V}(\mathcal{P})$ is the set of all vertices in path $\mathcal{P}$.

**Remark 5.2** *The Min-Exp-Cost-Path-NT problem is an important problem on its own (to address cases where no prior knowledge is available on nodes with $p_v = 1$), even though we have primarily introduced it here to help prove that the Min-Exp-Cost-Path problem is NP-hard.*

### 5.2.2 NP-hardness

In order to establish that Min-Exp-Cost-Path is NP-hard, we first introduce the decision versions of Min-Exp-Cost-Path (MECPD) and Min-Exp-Cost-Path-NT (MECP-NTD).

**Definition 5.1 (Min-Exp-Cost-Path Decision Problem)** *Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with starting node $v_s \in \mathcal{V}$, edge weights $l_e$, $\forall e \in \mathcal{E}$, probability of success $p_v \in [0,1]$, $\forall v \in \mathcal{V}$, such that $T \neq \emptyset$, and budget $B_{MECP}$, does there exist a path $\mathcal{P}$ from $v_s$ such that the expected cost of the path $\mathcal{C}(\mathcal{P}, 1) \leq B_{MECP}$?*

**Definition 5.2 (Min-Exp-Cost-Path-NT Decision Problem)** *Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with starting node $v_s \in \mathcal{V}$, edge weights $l_e, \forall e \in \mathcal{E}$, probability of success $p_v \in [0,1), \forall v \in \mathcal{V}$ and budget $B_{MECPNT}$, does there exist a path $\mathcal{P}$ from $v_s$ that visits all nodes in $\{v \in \mathcal{V} : p_v > 0\}$ such that $\sum_{e \in \mathcal{E}(\mathcal{P})} \left[\prod_{v \in \mathcal{V}(\mathcal{P}_e)} (1 - p_v)\right] l_e \leq B_{MECPNT}$?*

In the following Lemma, we first show that we can reduce MECPNTD to MECPD. This implies that if we have a solver for MECPD, we can use it to solve MECPNTD as well.

**Lemma 5.3** *Min-Exp-Cost-Path-NT Decision problem reduces to Min-Exp-Cost-Path Decision problem.*

*Proof:* Consider a general instance of MECPNTD with graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, starting node $v_s \in \mathcal{V}$, edge weights $l_e, \forall e \in \mathcal{E}$, probability of success $p_v \in [0,1), \forall v \in \mathcal{V}$, and budget $B_{\mathrm{MECPNT}}$. We create an instance of MECPD by introducing a new node $v_t$ into the graph with $p_{v_t} = 1$. We add edges of cost $l$ between $v_t$ and all the existing nodes of the graph. We next show that if we choose a large enough value for $l$, then the Min-Exp-Cost-Path solution would visit all nodes in $\bar{\mathcal{V}} = \{v \in \mathcal{V} : p_v > 0\}$ before moving to the terminal node $v_t$. Let $l = 1.5D/\min_{v\in\bar{\mathcal{V}}} p_v$, where $D$ is the diameter of the graph. Then, the Min-Exp-Cost-Path solution, which we denote by $\mathcal{P}^*$ must visit all nodes in $\bar{\mathcal{V}}$ before moving to node $v_t$. We show this by contradiction. Assume that this is not the case. Since $\mathcal{P}^*$ has not visited all nodes in $\bar{\mathcal{V}}$, there exists a node $w \in \bar{\mathcal{V}}$ that does not belong to $\mathcal{P}^*$. Let $\mathcal{Q}^*$ be the subpath of $\mathcal{P}^*$ that lies in the original graph $\mathcal{G}$ and let $u$ be the last node in $\mathcal{Q}^*$. Consider the path $\mathcal{P}$ created by stitching together the path $\mathcal{Q}^*$, followed by the shortest path from $u$ to $w$ and then finally the terminal node $v_t$. Let $p_f = \prod_{v\in\mathcal{V}(\mathcal{Q}^*)}(1-p_v)$ be the probability of failure after traversing path $\mathcal{Q}^*$. The expected cost of path $\mathcal{P}$ then satisfies

$$\begin{aligned}\mathcal{C}(\mathcal{P},1) &\leq \sum_{e\in\mathcal{E}(\mathcal{Q}^*)}\left[\prod_{v\in\mathcal{V}(\mathcal{Q}^*_e)}(1-p_v)\right]l_e + p_f\left(l^{\min}_{uw} + (1-p_w)l\right)\\ &< \sum_{e\in\mathcal{E}(\mathcal{Q}^*)}\left[\prod_{v\in\mathcal{V}(\mathcal{Q}^*_e)}(1-p_v)\right]l_e + p_f l = \mathcal{C}(\mathcal{P}^*,1),\end{aligned}$$

where $l^{\min}_{uw}$ is the cost of the shortest path between $u$ and $w$. We thus have a contradiction.

Thus, $\mathcal{Q}^*$ visits all the nodes in $\bar{\mathcal{V}}$. Moreover, since $\mathcal{P}^*$ is a solution of Min-Exp-Cost-Path, we can see that $\mathcal{Q}^*$ must also be a solution of Min-Exp-Cost-Path-NT. Thus, setting a budget of $B_{\text{MECP}} = B_{\text{MECPNT}} + p_f l$, where $p_f = \prod_{v\in\mathcal{V}(\mathcal{Q}^*)}(1-p_v) = \prod_{v\in\bar{\mathcal{V}}}(1-p_v)$, implies that the general instance of MECPNTD is satisfied if and only if our instance of MECPD is satisfied. ■

**Remark 5.3** *Even though we utilize the above Lemma primarily to analyze the computational complexity of the problems, we will also utilize the construction provided for path planners for Min-Exp-Cost-Path-NT in Section 5.5.*

We next show that MECPNTD is NP-complete (NP-hard and in NP), which together with Lemma 5.3, implies that MECPD is NP-hard.

**Theorem 5.1** *Min-Exp-Cost-Path-NT Decision problem is NP-complete.*

*Proof:* Clearly MECPNTD is in NP, since given a path we can compute its associated expected cost in polynomial time. We next show that MECPNTD is NP-hard using a reduction from a rooted version of the NP-hard Hamiltonian path problem [103]. Consider an instance of the Hamiltonian path problem $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the objective is to determine if there exists a path originating from $v_s$ that visits each vertex only once. We create an instance of MECPNTD by setting the probability of success to a non-zero constant for all nodes, i.e., $p_v = p > 0$, $\forall v \in \mathcal{V}$. We create a complete graph and set edge weights as $l_e = \begin{cases} 1, & \text{if } e \in \mathcal{E} \\ 2, & \text{else} \end{cases}$.

A Hamiltonian path $\mathcal{P}$ on $\mathcal{G}$, if it exists, would have an expected distance cost of

$$\sum_{e \in \mathcal{E}(\mathcal{P})} \left[ \prod_{v \in \mathcal{V}(\mathcal{P}_e)} (1 - p_v) \right] l_e = \frac{1-p}{p} \left(1 - (1-p)^{|\mathcal{V}|-1}\right).$$

Any path on the complete graph that is not Hamiltonian on $\mathcal{G}$, would involve either more edges or an edge with a larger cost than 1 and would thus have a cost strictly greater than that of $\mathcal{P}$. Thus, by setting $B_{\text{MECPNT}} = \frac{1-p}{p} \left(1 - (1-p)^{|\mathcal{V}|-1}\right)$, there exists a Hamiltonian path if and only if the specific MECPNTD instance created is satisfied. Thus, the general MECPNTD problem is at least as hard as the Hamiltonian path problem. Since the Hamiltonian path problem is NP-hard, this implies that MECPNTD is NP-hard. ■

**Corollary 5.1** *Min-Exp-Cost-Path Decision problem is NP-complete.*

*Proof:* We can see that MECPD is in NP. The proof of NP-hardness follows directly from Lemma 5.3. MECPD is thus NP-complete. ■

### 5.2.3 Min-Exp-Cost-Simple-Path

We now propose ways to tackle the prohibitive computational complexity (space complexity) of our MDP formulation of Section 5.1.2, which possesses a state space of size exponential in the number of nodes in the graph. If we can restrict ourselves to paths that do not revisit nodes, known as *simple paths* (i.e., cycle free paths), then the expected cost from a node could be expressed in terms of the expected cost from the neighboring node that the path visits next.[1] We refer to this problem of minimizing the expected cost, while restricted to the space of simple paths, as the *Min-Exp-Cost-Simple-Path* problem. The Min-Exp-Cost-Simple-Path problem is also computationally hard as shown in the following Lemma.

**Lemma 5.4** *The decision version of Min-Exp-Cost-Simple-Path is NP-hard.*

*Proof:* This follows from Theorem 5.1 and Lemma 5.3, since the optimal path considered in the construction of Theorem 5.1 was a simple path that visited all nodes. ■

Note that the optimal path of Min-Exp-Cost-Path could involve revisiting nodes, implying that the optimal solution to Min-Exp-Cost-Simple-Path on $\mathcal{G}$ could be suboptimal. For instance, consider the toy problem of Fig 5.2. The optimal path starting from node 2, in this case, is $\mathcal{P}^* = (2, 1, 2, 3, 4)$.

Consider Min-Exp-Cost-Simple-Path on the following complete graph. This complete graph $\mathcal{G}_{\text{comp}}$ is formed from the original graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ by adding an edge between all pairs of vertices of the graph, excluding self-loops. The cost of the edge $(u, v)$ is the cost of the shortest path between $u$ and $v$ on $\mathcal{G}$ which we denote by $l_{uv}^{\min}$. This can be computed by

[1] Note that depending on how we impose a simple path, we may need to keep track of the visited nodes. However, as we shall see, this keeping track of the history will not result in an exponential memory requirement, as was the case for the original MDP formulation. We further note that it is also possible to impose simple paths without a need to keep track of the history of the visited nodes, as we shall see in Section 5.4.2.

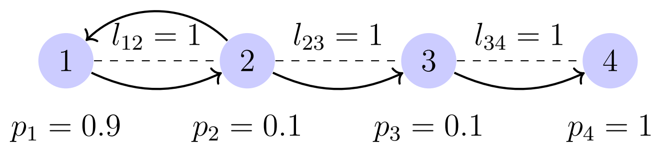


Figure 5.2: A toy example with the optimal path from node 2. The optimal path involves revisiting node 2.

the all-pairs shortest path Floyd-Warshall algorithm in $O(|\mathcal{V}|^3)$ computations. We next show in the following Lemma that the optimal solution of Min-Exp-Cost-Simple-Path on this complete graph can provide us with the optimal solution to Min-Exp-Cost-Path on the original graph.

**Lemma 5.5** *The solution to Min-Exp-Cost-Simple-Path on $\mathcal{G}_{comp}$ can be used to obtain the solution to Min-Exp-Cost-Path on $\mathcal{G}$.*

*Proof:* See Appendix C.1 for the proof. ■

Lemma 5.5 is a powerful result that allows us to asymptotically solve the Min-Exp-Cost-Path problem, with $\epsilon$ sub-optimality, as we shall see in the next Section.

## 5.3 Asymptotically $\epsilon$-suboptimal Path Planner

In this section, we propose a path planner, based on a game theoretic framework, that asymptotically gets arbitrarily close to the optimum solution of the Min-Exp-Cost-Path problem, i.e., it is an *asymptotically $\epsilon$-suboptimal solver*. This is important as it allows us to solve the NP-hard Min-Exp-Cost-Path problem, with near optimality, given enough time. More specifically, we utilize log-linear learning to asymptotically obtain the global potential minimizer of an appropriately defined potential game.

We start with the space of simple paths, i.e., we are interested in the Min-Exp-Cost-Simple-Path problem on a given graph $\mathcal{G}$. A node $v$ will then route to a single other node. Moreover, the expected cost from a node can then be expressed in terms of the expected cost from the neighbor it routes through. The state of the system can then be considered to be just the current node $v$, and the actions available at state $v$, $A_v = \{u \in \mathcal{V} : (v,u) \in \mathcal{E}\}$, is the set of neighbors of $v$. The policy $\mu$ specifies which node to move to next, i.e., if the current node is $v$, then $\mu_v$ is the next node to go to.

We next discuss our game-theoretic setting. So far, we viewed a node $v$ as a state and $A_v$ as the action space for state $v$. In contrast, in this game-theoretic setting, we interpret node $v$ as a player and $A_v$ as the action set of player $v$. Similarly, $\mu$ was viewed as a policy with $\mu_v$ specifying the action to take at state $v$. Here, we reinterpret $\mu$ as the joint action profile of the players with $\mu_v$ being the action of player $v$.

We consider a game $\{\mathcal{V}', \{A_v\}, \{\mathcal{J}_v\}\}$, where the set of non-terminal nodes $\mathcal{V}'$ are the players of the game and $A_v$ is the action set of node/player $v$. Moreover, $\mathcal{J}_v : A \to \mathbb{R}$ is the local cost function of player $v$, where $A = \prod_{v \in \mathcal{V}'} A_v$ is the space of joint actions. Finally, $\mathcal{J}_v(\mu)$ is the cost of the action profile $\mu$ as experienced by player $v$.

We first describe the expected cost from a node $v$ in terms of the action profile $\mu$. An action profile $\mu$ induces a directed graph on $\mathcal{G}$, which has the same set of nodes as $\mathcal{G}$ and directed edges from $v$ to $\mu_v$ for all $v \in \mathcal{V}'$. We call this the successor graph, using terminology from [104], and denote it by $\mathcal{SG}(\mu)$. As we shall show, our proposed strategy produces an action profile $\mu$ which induces a directed acyclic graph. This is referred to as an *acyclic successor graph* (ASG) [104].

Node $v$ is said to be *downstream* of $u$ in $\mathcal{SG}(\mu)$ if $v$ lies on the directed path from $u$ to the corresponding sink. Moreover, node $u$ is said to be *upstream* of $v$ in this case, and we denote the set of upstream nodes of $v$ by $U_v(\mu_{-v})$, where $\mu_{-v}$ denotes the action profile of all players except $v$. Let $v \in U_v(\mu_{-v})$ by convention. Note that $U_v(\mu_{-v})$ is only a function of $\mu_{-v}$ as it does not depend on the action of player $v$.

Let $\mathcal{P}(\mu, v)$ be the path from agent $v$ on this successor graph. We use the shorthand $C_v(\mu) = C(\mathcal{P}(\mu, v), 1)$, to denote the expected cost from node $v$ when following the path

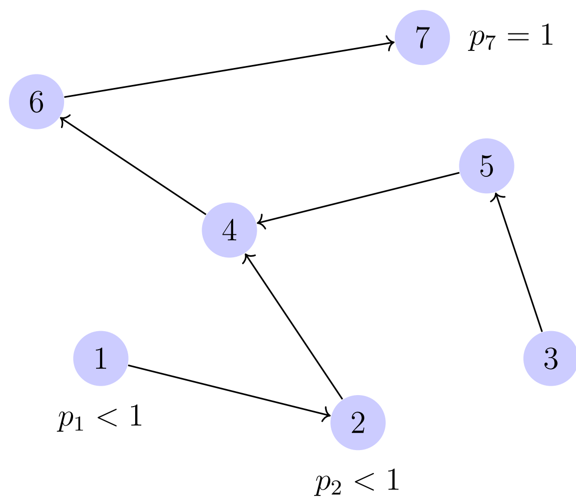


Figure 5.3: An example ASG induced by an action profile $\mu$.

$\mathcal{P}(\mu, v)$. Since $\mathcal{P}(\mu, v)$ is a path along $\mathcal{SG}(\mu)$, it can either end at some node or it can end in a cycle. If it ends in a cycle or at a node that is not a terminal node, we define the expected cost $C_v(\mu)$ to be infinity. If it does end at a terminal node, we obtain the following recursive relation from (5.1):

$$C_v(\mu) = (1 - p_v)\left(l_{v\mu_v} + C_{\mu_v}(\mu)\right), \tag{5.4}$$

where $C_{v_t}(\mu) = 0$ for all $v_t \in T$.

Let $A_{\text{ASG}}$ denote the set of action profiles such that the expected cost $C_v(\mu) < \infty$ for all $v \in \mathcal{V}$. This will only happen if the path $\mathcal{P}(\mu, v)$ ends at a terminal node for all $v$. This corresponds to $\mathcal{SG}(\mu)$ being an ASG with terminal nodes as sinks. Specifically, $\mathcal{SG}(\mu)$ would be a forest with the root or sink of each tree being a terminal node. An ASG is shown in Fig. 5.3 for the toy example from Fig. 5.1.

$\mu \in A_{\text{ASG}}$ implies that the action of player $v$ satisfies $\mu_v \in A_v^c(\mu_{-v})$, where $A_v^c(\mu_{-v}) = \{u \in \mathcal{V} : (v, u) \in \mathcal{E}, u \notin U_v(\mu_{-v}), C_u(\mu) < \infty\}$ is the set of actions that result in a finite expected cost from $v$. Note that $A_v^c(\mu_{-v})$ is a function of only $\mu_{-v}$. This is because

$u \notin U_v(\mu_{-v})$ implies $v \notin \mathcal{P}(\mu, u)$ which in turn implies that $C_u(\mu)$ is a function of only $\mu_{-v}$.

We next define the local cost function of player $v$ to be

$$\mathcal{J}_v(\mu) = \sum_{u \in U_v(\mu)} \varsigma_u C_u(\mu), \tag{5.5}$$

where $U_v(\mu)$ is the set of upstream nodes of $v$, and $\varsigma_u > 0$ are constants such that $\varsigma_{v_s} = 1$ and $\varsigma_v = \epsilon^{'}$, for all $v \neq v_s$, where $\epsilon^{'} > 0$ is a small constant.

We next show that these local cost functions induce a potential game over the action space $A_{\mathrm{ASG}}$. In order to do so, we first define a potential game over $A_{\mathrm{ASG}}$.[2]

**Definition 5.3 (Potential Game [105])** $\{\mathcal{V}', \{A_v\}, \{\mathcal{J}_v\}\}$ *is an exact potential game over $A_{ASG}$ if there exists a function $\Phi : A_{ASG} \rightarrow \mathbb{R}$ such that*

$$\mathcal{J}_v(\mu_v^{'}, \mu_{-v}) - \mathcal{J}_v(\mu_v, \mu_{-v}) = \Phi(\mu_v^{'}, \mu_{-v}) - \Phi(\mu_v, \mu_{-v}),$$

*for all $\mu_v^{'} \in A_v^c(\mu_{-v}), \mu = (\mu_v, \mu_{-v}) \in A_{ASG}$, and $v \in \mathcal{V}'$, where $\mu_{-v}$ denotes the action profile of all players except $v$.*

The function $\Phi$ is called the potential function. In the following Lemma, we show that using local cost functions as described in (5.5), results in an exact potential game.

**Lemma 5.6** *The game $\{\mathcal{V}', \{A_v\}, \{\mathcal{J}_v\}\}$, with local cost functions as defined in (5.5), is an exact potential game over $A_{ASG}$ with potential function*

$$\Phi(\mu) = \sum_{v \in \mathcal{V}'} \varsigma_v C_v(\mu) = C_{v_s}(\mu) + \epsilon^{'} \sum_{v \neq v_s} C_v(\mu). \tag{5.6}$$

[2]This differs from the usual definition of a potential game in that the joint action profiles are restricted to lie in $A_{\mathrm{ASG}}$.

*Proof:* Consider a node $v$ and $\mu = (\mu_v, \mu_{-v})$ and $\mu_v^{'}$ such that $C_v(\mu_v^{'}, \mu_{-v}) < C_v(\mu_v, \mu_{-v})$. From (5.4), we have that $C_u(\mu_v^{'}, \mu_{-v}) < C_u(\mu_v, \mu_{-v}), \quad \forall u \in U_v(\mu)$, where $U_v(\mu)$ is the set of upstream nodes from $v$. Furthermore, $C_u(\mu_v^{'}, \mu_{-v}) = C_u(\mu_v, \mu_{-v}), \quad \forall u \notin U_v(\mu)$. Thus, we have

$$\begin{aligned}
\Phi(\mu_v^{'}, \mu_{-v}) - \Phi(\mu) &= \sum_{u \in \mathcal{V}'} \varsigma_u \left[C_u(\mu_v^{'}, \mu_{-v}) - C_u(\mu)\right] \\
&= \sum_{u \in U_v(\mu)} \varsigma_u \left[C_u(\mu_v^{'}, \mu_{-v}) - C_u(\mu)\right] \\
&= \mathcal{J}_v(\mu_v^{'}, \mu_{-v}) - \mathcal{J}_v(\mu),
\end{aligned}$$

for all $\mu_v^{'} \in A_v^c(\mu_{-v})$, $\mu \in A_{\text{ASG}}$, and $v \in \mathcal{V}'$. ■

Minimizing $\Phi(\mu)$ gives us a solution that can be arbitrarily close to that of Min-Exp-Cost-Simple-Path since we can select the value of $\epsilon^{'}$ appropriately. Let $\mu^* = \arg\min_\mu \Phi(\mu)$ and $\mu^{\text{OPT}} = \arg\min_\mu C_{v_s}(\mu)$. Then,

$$C_{v_s}(\mu^*) + \epsilon^{'} \sum_{u \neq v_s} C_u(\mu^*) \leq C_{v_s}(\mu^{\text{OPT}}) + \epsilon^{'} \sum_{u \neq v_s} C_u(\mu^{\text{OPT}}).$$

Rearranging gives us

$$\begin{aligned}
C_{v_s}(\mu^*) &\leq C_{v_s}(\mu^{\text{OPT}}) + \epsilon^{'} \left[\sum_{u \neq v_s} C_u(\mu^{\text{OPT}}) - \sum_{u \neq v_s} C_u(\mu^*)\right] \\
&\leq C_{v_s}(\mu^{\text{OPT}}) + \epsilon^{'} |\mathcal{V}'| D,
\end{aligned}$$

where $D$ is the diameter of the graph. Thus minimizing $\Phi(\mu)$ gives us an $\epsilon$-suboptimal solution to the Min-Exp-Cost-Simple-Path problem, where $\epsilon = \epsilon^{'} |\mathcal{V}'| D$.

We next show how to asymptotically obtain the global minimizer of $\Phi(\mu)$ by utilizing a learning process known as log-linear learning [106].

**Log-linear Learning**

Let $\mu_v = a_\emptyset$ correspond to node $v$ not pointing to any successor node. We refer to this as a null action. Then, the log-linear process utilized in our setting is as follows:

1. The action profile $\mu(0)$ is initialized with a null action, i.e., $\mu_v(0) = a_\emptyset$ for all $v$. The local cost function is thus $\mathcal{J}_v(\mu(0)) = \infty$, for all $v \in \mathcal{V}'$.

2. At every iteration $k+1$, a node $v$ is randomly selected from $\mathcal{V}'$ uniformly. If $A_v^c(\mu_{-v}(k))$ is empty, we set $\mu_v(k+1) = a_\emptyset$. Else, node $v$ selects action $\mu_v(k+1) = \mu_v \in A_v^c(\mu_{-v}(k))$ with the following probability:

$$\Pr(\mu_v) = \frac{e^{-\frac{1}{\tau}(\mathcal{J}_v(\mu_v,\mu_{-v}(k)))}}{\sum_{\mu'_v \in A_v^c(\mu_{-v}(k))} e^{-\frac{1}{\tau}\left(\mathcal{J}_v(\mu'_v,\mu_{-v}(k))\right)}},$$

   where $\tau$ is a tunable parameter known as the temperature. The remaining nodes repeat their action, i.e., $\mu_u(k+1) = \mu_u(k)$ for $u \neq v$.

We next show that log-linear learning asymptotically obtains an $\epsilon$-suboptimal solution to the Min-Exp-Cost-Path problem. We first show, in the following Lemma, that it asymptotically provides an $\epsilon$-suboptimal solution to the Min-Exp-Cost-Simple-Path problem.

**Theorem 5.2** *As $\tau \to 0$, log-linear learning on a potential game with a local cost function defined in (5.5), asymptotically provides an $\epsilon$-suboptimal solution to the Min-Exp-Cost-Simple-Path problem.*

*Proof:* See Appendix C.2 for the proof. ■

**Lemma 5.7** *As $\tau \to 0$, log-linear learning on a potential game with a local cost function defined in (5.5) on the complete graph $\mathcal{G}_{comp}$, asymptotically provides an $\epsilon$-suboptimal solution to the Min-Exp-Cost-Path problem.*

*Proof:* From Theorem 5.2, we know that log-linear learning asymptotically provides an $\epsilon$-suboptimal solution to the Min-Exp-Cost-Simple-Path problem on the complete graph $\mathcal{G}_{\text{comp}}$. Using Lemma 5.5, we then utilize this solution to obtain an $\epsilon$-suboptimal solution to the Min-Exp-Cost-Path problem on $\mathcal{G}$. ■

**Remark 5.4** *We implement the log-linear learning algorithm by keeping track of the expected cost $C_v(\mu(k))$ in memory, for all nodes $v \in \mathcal{V}'$. In each iteration, we compute the set of upstream nodes of the selected node $v$ in order to compute the set $A_v^c(\mu(-k))$. From (5.4), we can see that the expected cost of each node upstream of $v$ can be expressed as a linear function of $C_v(\mu)$. Then we can compute an expression for $\mathcal{J}_v(\mu) = \sum_{u \in \mathcal{U}_v(\mu_{-v})} C_u(\mu)$ as a linear function of the expected cost $C_v(\mu)$ with a computational cost of $O(|\mathcal{V}'|)$. We can then compute $\mathcal{J}_v(\mu_v, \mu_{-v})$ for all $\mu_v \in A_v^c(\mu(-k))$ using this precomputed expression for $\mathcal{J}_v()$. Finally, once $\mu_v(k+1)$ is selected, we update the expected cost of $v$ as well as all its upstream nodes using (5.4). Thus, the overall computation cost of each iteration is $O(|\mathcal{V}'|)$.*

## 5.4 Fast Non-myopic Path Planners

In the previous section, we proposed an approach that finds an $\epsilon$-suboptimal solution to the Min-Exp-Cost-Path problem asymptotically. However, for certain applications, finding a suboptimal but fast solution may be more important. This motivates us to propose two suboptimal path planners that are non-myopic and very fast. We use the term non-myopic here to contrast with the myopic approaches of choosing your next step based on your immediate or short-term reward (e.g., local greedy search). We shall see an example of such a myopic heuristic in Section 5.5.

In this part, we first propose a non-myopic path planner based on a game theoretic framework that finds a directionally local minimum of the potential function $\Phi$ of (5.6).

We next propose a path planner based on an SSP formulation that provides us with the optimal path among the set of paths satisfying a mild assumption.

We assume simple paths in this Section. Lemma 5.5 can then be used to find a optimum non-simple path with minimal computation. Alternatively, the simple path solution can also be directly utilized.

### 5.4.1 Best Reply Process

Consider the potential game $\{\mathcal{V}', \{A_v\}, \{\mathcal{J}_v\}\}$ of Section 5.3 with local cost functions $\{\mathcal{J}_v\}$ as given in (5.5). We next show how to obtain a directionally local minimum of the potential function $\Phi(\mu) = C_{v_s}(\mu) + \epsilon' \sum_{v \neq v_s} C_v(\mu)$. In order to do so, we first review the definition of a Nash equilibrium.

**Definition 5.4 (Nash Equilibrium [107])** *An action profile $\mu^{NE}$ is said to be a pure Nash equilibrium if*

$$\mathcal{J}_v(\mu^{NE}) \leq \mathcal{J}_v(\mu_v, \mu_{-v}^{NE}), \;\; \forall \mu_v \in A_v, \forall v \in \mathcal{V}'$$

*where $\mu_{-v}$ denotes the action profile of all players except $v$.*

It can be seen that an action $\mu^{\mathrm{NE}}$ is a Nash equilibrium of a potential game if and only if it is a directionally local minimum of $\Phi$, i.e., $\Phi(\mu_v', \mu_{-v}^{\mathrm{NE}}) \geq \Phi(\mu^{\mathrm{NE}}), \;\; \forall \mu_v' \in A_v, \forall v \in \mathcal{V}'$. Since we have a potential game, a Nash equilibrium of the game is a directionally local minimum of $\Phi(\mu)$. We can find a Nash equilibrium of the game using a learning mechanism such as the best reply process [106], which we next discuss.

Let $\mu_v = a_\emptyset$ correspond to node $v$ not pointing to any successor node. We refer to this as a null action. The best reply process utilized in our setting is as follows:

1. The action profile $\mu(0)$ is initialized with a null action, i.e., $\mu_v(0) = a_\emptyset$ for all $v$.

The local cost function is thus $\mathcal{J}_v(\mu(0)) = \infty$, for all $v \in \mathcal{V}'$.

2. At iteration $k+1$, a node $v$ is randomly selected from $\mathcal{V}'$ uniformly. If $A_v^c(\mu_{-v}(k))$ is empty, we set $\mu_v(k+1) = a_\emptyset$. Else, the action of node $v$ is updated as

$$\begin{aligned}\mu_v(k+1) &= \underset{\mu_v \in A_v^c(\mu_{-v}(k))}{\arg\min} \; \mathcal{J}_v(\mu_v, \mu_{-v}(k)) \\ &= \underset{\mu_v \in A_v^c(\mu_{-v}(k))}{\arg\min} \; C_v(\mu_v, \mu_{-v}(k)) \\ &= \underset{u \in A_v^c(\mu_{-v}(k))}{\arg\min} \; \left\{(1-p_v)\left[l_{vu} + C_u(\mu(k))\right]\right\},\end{aligned}$$

where the second and third equality follow from (5.4). The actions of the remaining nodes stay the same, i.e., $\mu_u(k+1) = \mu_u(k)$, $\forall u \neq v$.

The best reply process in a potential game converges to a pure Nash equilibrium [106], which is also a directionally local minimum of $\Phi(\mu) = C_{v_s}(\mu) + \epsilon' \sum_{v \neq v_s} C_v(\mu)$.

Since a node is selected at random at each iteration in the best reply process, analyzing its convergence rate becomes challenging. Instead, in the following Theorem, we analyze the convergence rate of the best reply process when the nodes for update are selected deterministically in a cyclic manner. We show that it converges quickly to a directionally local minimum, and is thus an efficient path planner.

**Theorem 5.3 (Computational complexity)** *Consider the best reply process where we select the next node for update in a round robin fashion. Then, this process converges after at most $|\mathcal{V}'|^2$ iterations.*

*Proof:* See Appendix C.3 for the proof. ∎

**Remark 5.5** *We implement the best reply process by keeping track of the expected cost $C_v(\mu(k))$ in memory, for all nodes $v \in \mathcal{V}'$. In each iteration of the best reply process,* *we compute the set of upstream nodes of the selected node $v$ in order to compute the set $A_v^c(\mu(-k))$. Moreover, we compute $l_{v\mu_v} + C_{\mu_v}(\mu(k))$ for all $\mu_v \in A_v^c(\mu(-k))$ to find the action $\mu_v$ that minimizes the expected cost from $v$. Finally, once $\mu_v(k+1)$ is selected, we update the expected cost of $v$ as well as all the nodes upstream of it using (5.4). Then, the computation cost of each iteration is $O(|\mathcal{V}'|)$. Thus, from Theorem 5.3, the best reply process in a round robin setting has a computational complexity of $O(|\mathcal{V}'|^3)$.*

### 5.4.2 Imposing a Directed Acyclic Graph

We next propose an SSP-based path planner. We enforce that a node cannot be revisited by imposing a directed acyclic graph (DAG), $\mathcal{G}_{\text{DAG}}$, on the original graph. The state of the SSP formulation of Section 5.1.2 is then just the current node $v \in \mathcal{V}'$. The transition probability from state $v$ to state $u$ is then simply given as

$$P_{vu}(a_v) = \begin{cases} 1 - p_v, & \text{if } u = g_{\text{trans}}(a_v) \\ p_v, & \text{if } u = s_t \\ 0, & \text{else} \end{cases},$$

where $g_{\text{trans}}(a_v) = \begin{cases} a_v, & \text{if } a_v \in \mathcal{V}' \\ s_t, & \text{if } a_v \in T \end{cases}$, and the stage cost of action $u$ at state $v$ is given as $g_{\text{cost}}(v, u) = (1 - p_v) l_{vu}$. We refer to running value iteration on this SSP as the IDAG (imposing a DAG) path planner.

Imposing a DAG, $\mathcal{G}_{\text{DAG}} = (\mathcal{V}, \mathcal{E}_{\text{DAG}})$, corresponds to modifying the action space of each state $v$ such that only a subset of the neighbors are available actions, i.e., $A_v = \{u : (v, u) \in \mathcal{E}_{\text{DAG}}\}$. For instance, given a relative ordering of the nodes, a directed edge would be allowed from node $u$ to $v$, only if $v \geq u$ with respect to some ordering. As a concrete example, consider the case where a directed edge from node $u$ to $v$ exists only if

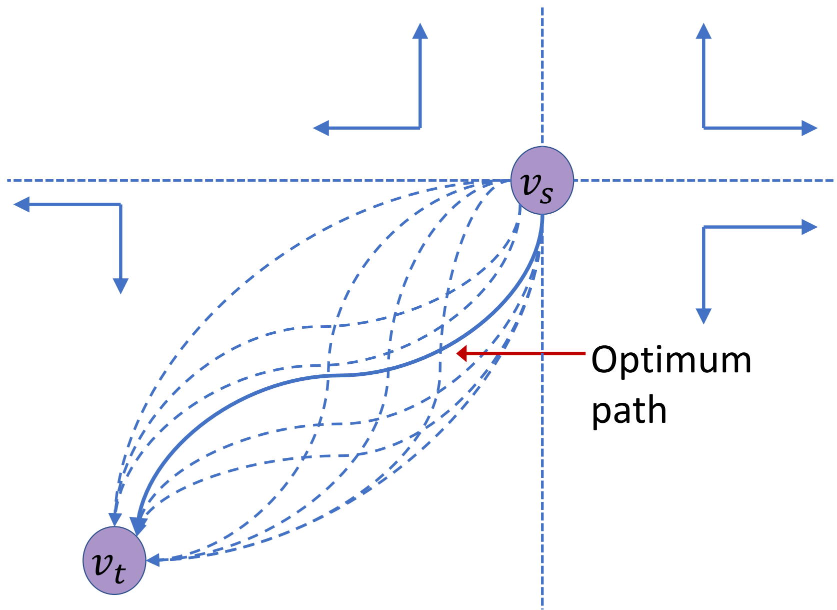


Figure 5.4: A DAG is imposed which allows only "outward" motion. The solution produced by SSP would be the best among all such paths from the start node $v_s$ to the terminal node $v_t$.

$v$ is farther away from the starting node $v_s$ on the graph than node $u$ is, i.e., $l^{\min}_{v_s v} > l^{\min}_{v_s u}$, where $l^{\min}_{v_s v}$ is the cost of the shortest path from $v_s$ to $v$ on the original graph $\mathcal{G}$. More specifically, the imposed DAG has the same set of nodes $\mathcal{V}$ as the original graph, and the set of edges is given by $\mathcal{E}_{\text{DAG}} = \{(u, v) \in \mathcal{E} : l^{\min}_{v_s v} > l^{\min}_{v_s u}\}$, where $(u, v)$ represents a directed edge from $u$ to $v$. For example, consider an $n \times n$ grid graph, where neighboring nodes are limited to {left, right, top, down} nodes. In the resulting DAG, only outward flowing edges from the start node are allowed, i.e., edges that take you further away from the start node. For instance, consider the start node $v_s$ as the center and for each quadrant, form outward moving edges, as shown in Fig. 5.4. In the first quadrant only right and top edges are allowed, in the second quadrant only left and top edges and so on. Fig. 5.4 shows an illustration of this, where several feasible paths from $v_s$ to a terminal node are shown.

Imposing this DAG is equivalent to placing the following requirement that a feasible

| Size of grid ($n$) | 5 | 10 | 15 | 20 | 25 | 30 | 40 | 50 |
|---|---|---|---|---|---|---|---|---|
| RTDP (MDP formulation) | 3.510 | - | - | - | - | - | - | - |
| Simulated Annealing | 3.510 | 7.515 | 84.138 | 159.88 | 150.59 | 25.997 | 426.71 | 406.71 |
| Best Reply | 3.510 | 6.722 | 8.588 | 10.437 | 10.569 | 11.032 | 12.930 | 12.530 |
| Log-linear | 3.510 | 6.722 | 8.588 | 10.437 | 10.569 | 11.032 | 13.450 | 12.186 |

Table 5.1: Expected traveled distance for the path produced by the various approaches for different grid sizes ($n$) with a single connected point ($n_t = 1$). We can see that RTDP is unable to produce a viable path for $n \geq 10$ and that simulated annealing produces paths with poor performance for increasing $n$.

path must satisfy: *Each successive node on the path must be further away from the starting node* $v_s$, i.e., for a path $\mathcal{P} = (v_1 = v_s, v_2, \cdots, v_m)$, the condition $l^{\min}_{v_s v_i} > l^{\min}_{v_s v_{i-1}}$ should be satisfied. In the case of a grid graph with a single terminal node, this implies that a path must always move towards the terminal node, which is a reasonable requirement to impose. We next show that we can obtain the optimal solution among all paths satisfying this requirement using value iteration.

The optimal solution with minimum expected cost on the imposed DAG $\mathcal{G}_{\text{DAG}}$ can be found by running value iteration:

$$\overline{J}_v(k+1) = \min_{u \in A_v} \left\{ (1-p_v) l_{vu} + (1-p_v) \overline{J}_{g_{\text{trans}}(u)}(k) \right\},$$

with the policy at iteration $k+1$ given by

$$\mu_v(k+1) = \arg\min_{u \in A_v} \left\{ (1-p_v) l_{vu} + (1-p_v) \overline{J}_{g_{\text{trans}}(u)}(k) \right\},$$

for all $v \in \mathcal{V}'$, where $\overline{J}_{s_t}(k) = 0$, for all $k$.

The following lemma shows that we can find this optimal solution efficiently.

**Lemma 5.8 (Computational complexity)** *When starting from* $\overline{J}_v(0) = \infty$*, for all* $v \in \mathcal{V}'$*, the value iteration method will yield the optimal solution after at most* $|\mathcal{V}'|$ *iterations.*

*Proof:* This follows from the convergence analysis of value iteration on an SSP with a DAG structure [102]. ■

**Remark 5.6** *Each stage of the value iteration process has a computation cost of* $O(|\mathcal{E}_{DAG}|)$ *since for each node we have as many computations as there are outgoing edges. Thus, from Lemma 5.8, we can see that the computational cost of value iteration is* $O(|\mathcal{V}'||\mathcal{E}_{DAG}|)$.

**Remark 5.7** *Log-linear learning, best reply, and IDAG, each have their own pros and cons. For instance, log-linear learning has strong asymptotic optimality guarantees. In contrast, best reply converges quickly to a directionally-local minimum but does not possess similar optimality guarantees. Numerically, for the applications considered in Section 5.5, the best reply solver performs better than the IDAG solver. However, the IDAG approach is considerably fast and provides a natural understanding of the solution it produces, being particularly suitable for spatial path planning problems. For instance, as shown in Fig. 5.4, the solution of IDAG for the imposed DAG is the best solution among all paths that move outward from the start node. More generally, it is the optimal solution among all the paths allowed by the imposed DAG.*

## 5.5 Numerical Results

In this section, we show the performance of our approaches for Min-Exp-Cost-Path, via numerical analysis of two applications. In our first application, a rover is exploring mars, to which we refer as the SamplingRover problem. In our second application, we then consider a realistic scenario of a robot planning a path in order to find a connected spot to a remote station. We see that in both scenarios our solvers perform well and outperform the naive and greedy heuristic approaches.

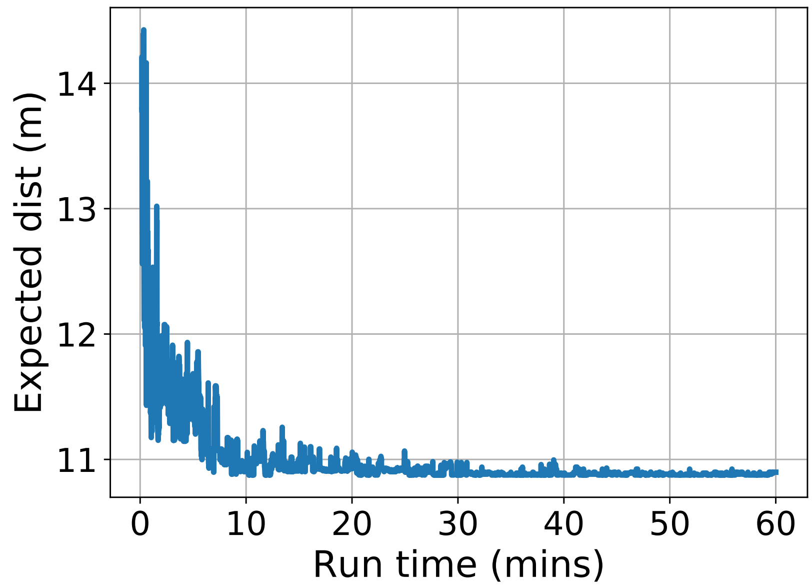


Figure 5.5: Evolution of the expected traveled distance with time for the log-linear learning approach with $n = 25$ and $n_t = 1$.

### 5.5.1 Sampling Rover

The scenario considered here is loosely inspired by the RockSample problem introduced in [108]. A rover on a science exploration mission is exploring an area looking for an object of interest for scientific studies. For instance, consider a rover exploring Mars with the objective of obtaining a sample of water. Based on prior information, which could for instance be from orbital flyovers over the area of interest or from the estimation by experts, the rover has an a priori probability of finding the object at any location.

An instance of the SamplingRover[$n$,$n_t$] consists of an $n \times n$ grid with $n_t$ locations of guaranteed success, i.e., $n_t$ nodes such that $p_v = 1$. The probability of success at each node is generated independently and uniformly from $[0, 0.1]$. At any node, the actions allowed by the rover are {left, right, top, down}. The starting position of the rover is taken to be at the center of the grid, $v_s = \left(\lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{2} \rfloor\right)$. When the number of points of guaranteed success ($n_t$) is 1, we take the location of the node with $p_v = 1$ at $(0, 0)$.

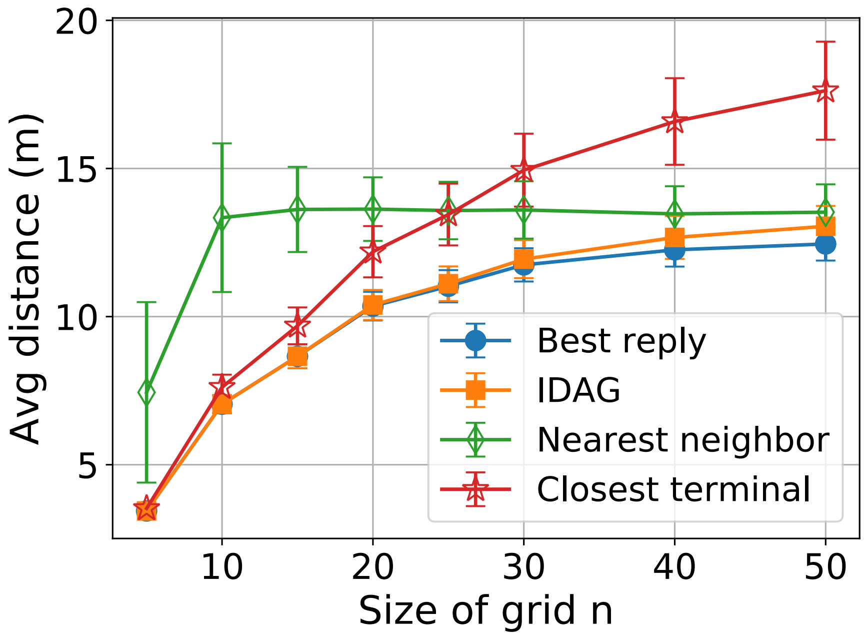


Figure 5.6: The expected traveled distance by the various approaches for different grid sizes ($n$) with a single connected point ($n_t = 1$). The results are averaged over 1000 different probability of success maps. The corresponding standard deviation is also shown in the form of error bars. We can see that the best reply and IDAG approaches outperform the greedy and closest terminal heuristics.

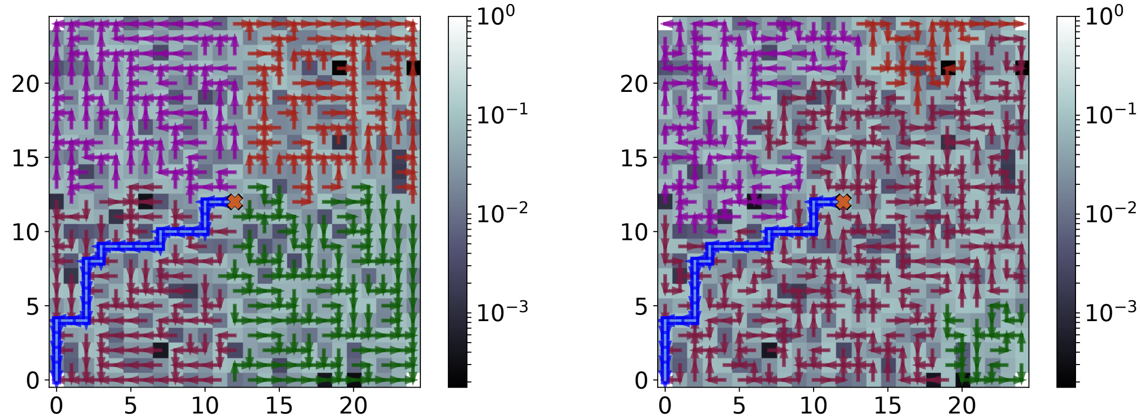


Figure 5.7: Acyclic successor graph (ASG) of (left) best reply process and (right) log-linear learning process for $n = 25$, when there are four nodes with $p_v = 1$. The trees corresponding to each of the four terminal nodes are marked in purple, red, green and brown. The path traveled from the starting node is also plotted (in blue). The starting position at $(12, 12)$ is marked by the orange "x". The background color plot specifies the probability of success of each node.

We found that log-linear learning on a complete graph produces similar results as log-linear learning on the original grid graph, but over longer run-times. Thus, unless explicitly mentioned otherwise, when we refer to the best reply or the log-linear learning approach, it is with respect to finding a simple path on the original grid graph. We set weight $\epsilon^{'} = 10^{-6}$ in $\Phi(\mu) = C_{v_s}(\mu) + \epsilon^{'} \sum_{v \in \mathcal{V}', v \neq v_s} C_v(\mu)$. We use a decaying temperature for log-linear learning. Through experimentation, we found that a decaying temperature of $\tau \propto k^{-0.75}$ (where $k$ is the iteration number) performs well.

We first compare our approach with alternate approaches for solving the Min-Exp-Cost-Path problem. We consider one instance of a probability of success map. We then implement Real Time Dynamic Programming (RTDP) [109], which is a heuristic search method that tries to obtain a good solution quickly for the MDP formulation of Section 5.1.2. Furthermore, we also implemented Simulated Annealing as implemented in [110] for the traveling salesman problem, where we modify the cost of a state to be the expected cost from the starting node. Moreover, the starting position of the rover is fixed as the start of the simulated annealing path. Table 5.1 shows the performance of RTDP, simulated annealing and our (asymptotically $\epsilon$-suboptimal) log-linear learning and (non-myopic fast) best reply approaches for various grid sizes ($n$) when $n_t = 1$, where for each approach we impose a computational time limit of an hour. We see that RTDP is unable to produce viable solutions for $n \geq 10$ due to the state explosion problem of the MDP formulation, as discussed in Section 5.1.2. Moreover, the performance of simulated annealing worsens significantly with increasing values of $n$. On the other hand, the best reply and log-linear learning approach produce solutions with good performance that outperform simulated annealing considerably (e.g., simulated annealing has 15 times more expected traveled distance than the best reply approach for $n = 20$).

We next show the asymptotically $\epsilon$-suboptimal behavior of the log-linear learning approach of Section 5.3. Fig. 5.5 shows the evolution of the expected distance with time for the solution produced by log-linear learning for an instance of a probability of success map with $n = 25$ and $n_t = 1$. In comparison, the best reply and IDAG approaches converged in 1.75 s and 0.25 s respectively.

**Remark 5.8** *We note that based on several numerical results, we have observed that the best reply and IDAG approaches produce results very close to those produced by log-linear learning. They thus act as fast efficient solvers. On the other hand, the log-linear learning approach provides a guarantee of optimality (within $\epsilon$) asymptotically. Thus all* 3 *approaches are useful depending on the application requirements.*

We next compare our proposed approaches with two heuristics. The first is a heuristic of moving straight towards the closest node with $p_v = 1$, which we refer to as the *closest terminal* heuristic. The second is a myopic greedy heuristic, where the rover at any time moves towards the node with the highest $p_v$ among its unvisited neighbors. We refer to this as the *nearest neighbor* heuristic. These are similar to strategies utilized in the optimal search theory literature [62, 65], where myopic strategies with limited lookahead are typically utilized. Fig. 5.6 shows the performance of the best reply, IDAG, nearest neighbor and closest terminal heuristic for various grid sizes ($n$) when $n_t = 1$. We generated a 1000 different probability of success maps, and averaged the expected traveled distance over them to obtain the plotted performance for each $n$. Also, the error bars in the plot represent the standard deviation of each approach. In Fig. 5.6, we can see that the best reply and IDAG approach outperform the greedy nearest neighbor heuristic as well as the closest terminal heuristic significantly. Moreover, the best reply approach outperforms the IDAG approach for larger $n$.

In order to gain more insight into the nature of the solution produced by our proposed approaches, we next consider a scenario where $n_t = 4$, where we place the four nodes of guaranteed success at the four corners of the workspace, i.e., at $(0, 0)$, $(0, n-1)$ $(n-1, 0)$ and $(n-1, n-1)$. Fig. 5.7 shows the ASG of the best reply process and log-linear learning for a sample such scenario, where we impose a computational time limit of 1 hour on the log-linear learning approach. We see that in both cases, the resulting ASG is a forest with 4 trees, each denoted with a different color in Fig. 5.7, where the roots of the 4 trees correspond to the 4 nodes of guaranteed success. As discussed in Section 5.4, the solution ASG of the best reply process is an equilibrium where no node can improve its expected traveled distance by switching the neighbor it routes to. The route followed by the rover is also plotted on the ASG, which can be seen to visit nodes of higher probability of success. Fig. 5.8 shows a plot of the routes traveled by the IDAG and the nearest neighbor approach. In this instance, the paths produced by the best reply and log-linear learning approach were the same as that of the IDAG approach.

We next consider the case of $n_t = 0$, which corresponds to no terminal node being present. This is an instance of the Min-Exp-Cost-Path-NT problem. In this setting,

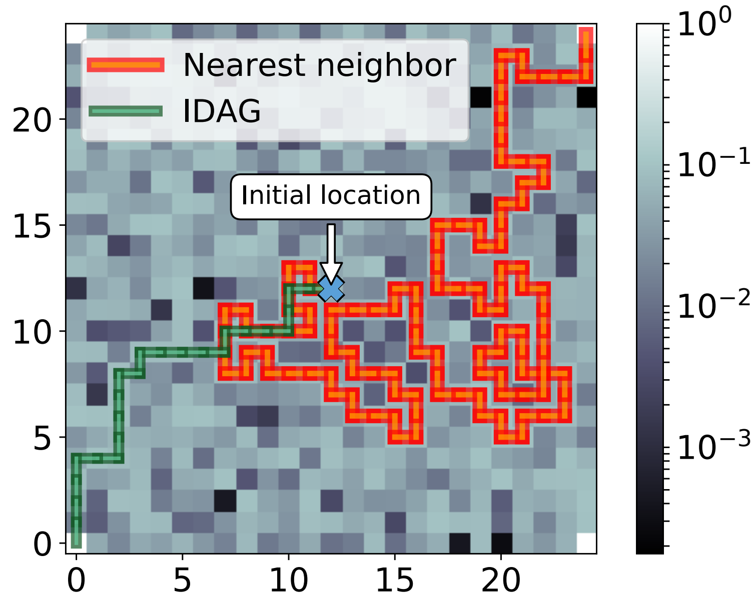


Figure 5.8: Path traveled by IDAG and nearest neighbor approach for $n = 25$, when there are four nodes with $p_v = 1$. The solution path produced by the best reply and log-linear learning approaches are the same as that of the IDAG approach in this instance. The background color plot specifies the probability of success of each node.

the solution we are looking for is a tour of all nodes $\{v \in \mathcal{V} : p_v > 0\}$ that minimizes $\sum_{e\in\mathcal{E}(\mathcal{P})} \prod_{v\in\mathcal{V}(\mathcal{P}_e)}(1 - p_v)l_e$. In order to facilitate the use of our approaches on the Min-Exp-Cost-Path-NT problem, we introduce a terminal node in the grid graph as discussed in the construction in the proof of Lemma 5.3. We include an edge weight $l = 1.5 \times \frac{D}{\min_v p_v}$ between the artificial terminal node and all other nodes, where $D = 2n$ is the diameter of the graph. Note that these solution paths may not visit all the nodes in the grid graph, due to the limited computation time. The best reply process was run 100 times and the best solution was selected among the solutions produced. Moreover, we impose a computational time limit of 1 hour on the log-linear learning approach. Fig. 5.9 shows the ASG for the best reply and log-linear learning process as well as the path traveled from the starting node for both cases. We can see that the paths produced by both approaches traverse through nodes of high probability of success. Since success is not guaranteed

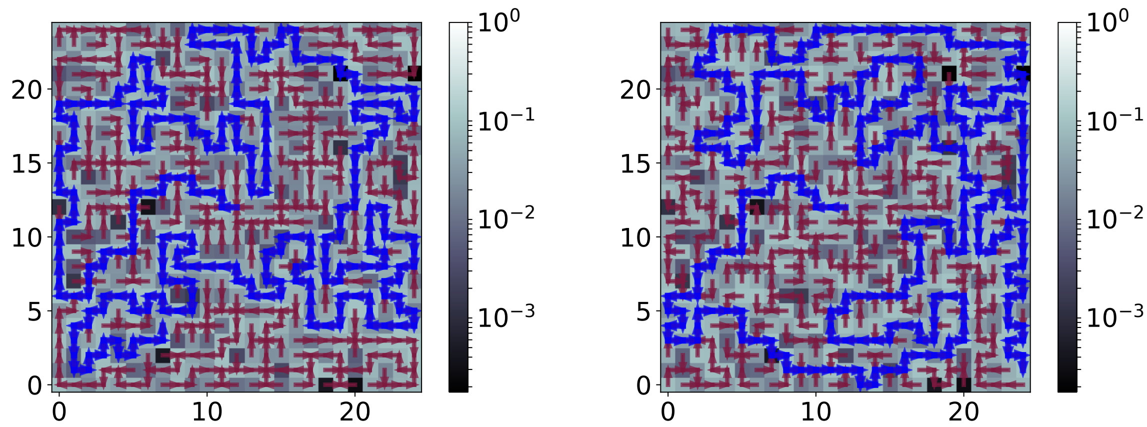


Figure 5.9: Acyclic successor graph (ASG) of (left) best reply process and (right) log–linear learning process for $n = 25$ when there is no terminal node. The path traveled from the starting node is also plotted (in blue). The starting position at $(12, 12)$ is marked by the orange "x". The background color plot specifies the probability of success of each node.

| | Exp distance (m) | Prob of fail along path |
|---|---|---|
| Best reply | 12.40 | $3.5e-7$ |
| Log-linear learning | 12.51 | $2.5e-7$ |

Table 5.2: The expected traveled distance and the probability of failure along path for the best reply and log-linear learning approaches.

when traversing along a solution path of an approach, expected distance until success is no longer well defined. In other words, we no longer have a single metric by which to judge the quality of a solution. Instead, we now have two metrics, the probability of failure along a path and the expected distance of traversing the path. Table. 5.2 shows the performance of the best reply and log-linear approaches on these metrics for the sample scenario shown in Fig. 5.9. We see that both best reply and log-linear approaches produce a solution with good performance.

### 5.5.2 Connectivity seeking robot

In this section, we consider the scenario of a robot seeking to get connected to a remote station. We say that the robot is connected if it is able to reliably transfer information to the remote station. This would imply satisfying a Quality of Service (QoS) requirement such as a target bit error rate (BER), which would in turn imply a minimum required received channel power given a fixed transmit power. Thus, in order for the robot to get connected, it needs to find a location where the channel power, when transmitting from that location, would be greater than the minimum required channel power. However, the robot's prior knowledge of the channel is stochastic. Thus, for a robot seeking to do this in an energy efficient manner, its goal would be to plan a path such that it gets connected with a minimum expected traveled distance.

For the robot to plan such a path, it would require an assessment of the channel quality at any unvisited location. As discussed in Section 2.1, the spatial variations of the channel have three main components: path loss, shadowing and multipath fading. The shadowing component is the result of attenuation through large stationary objects in the environment, such as buildings or hills. We thus assume that the path loss and shadowing component stay constant with time. Multipath fading, on the other hand, is the result of the additions of multiple paths. Thus, small changes in the positions of reflectors and scatterers in the environment can cause large changes in the multipath fading signal. Thus, depending on environmental factors such as speed of motion of the scatterers, the multipath component could be time-invariant or time-varying. In this section, we assume that the multipath component is time-varying. This then implies that the multipath value at any location during the operation phase of the robot, is independent of its corresponding value when the prior measurements were collected [73,111,112]. Note that the probabilistic channel prediction framework of Section 2.2 focuses on the case where the multipath component is time-invariant. It can be easily extended to account for the case where the multipath component of the operation phase is independent of its corresponding prior measurement value. See [49] for more details on this.

Consider a scenario where the robot is located in the center of a 50 m $\times$ 50 m workspace as shown in Fig. 5.10, with the remote station located at the origin. The channel is generated using the realistic probabilistic channel model of Section 2.1, with the following parameters that were obtained from real channel measurements in downtown San Francisco [73] : path loss exponent $n_{\text{PL}} = 4.2$, shadowing power $\sigma_{\text{SH}} = 2.9$ and shadowing decorrelation distance $\beta_{\text{SH}} = 12.92$ m. Moreover, the multipath fading is taken to be uncorrelated Rician fading with the parameter $K_{\text{ric}} = 1.59$. In order for the robot to be connected, we require a minimum required received power of $P_{R,\text{th,dBm}} = -80$ dBmW. We take the maximum transmission power of a node to be $P_{0,\text{dBm}} = 27$ dBmW [80]. The robot is assumed to have 5 % a priori measurements in the workspace.

We discretize the workspace of the robot into cells of size 1 m by 1 m. A cell is connected if there exists a location in the cell that is connected. For instance, consider a cell that consists of locations $r = [r_1, \cdots, r_k]^T$. The probability of failure of connectivity of the cell is then given by $\Pr(\Gamma(r_i) < \gamma_{\text{th}}, \forall i \leq k)$, where $\Gamma(r) = [\Gamma(r_1) \cdots \Gamma(r_k)]^{\text{T}} \sim \mathcal{N}\left(\overline{\Gamma}(r), \Sigma(r)\right)$ is a Gaussian random vector as obtained from our channel prediction framework, and $\gamma_{\text{th}}$ is the minimum required channel power for connectivity. See [49] for more details on this estimation. We next construct a grid graph with each cell serving as a node on our graph. This gives us a grid graph of dimension 50x50 with a probability of connectivity assigned to each node. We also add a new terminal node to the graph with probability of connectivity 1, which represents the remote station at the origin. We attach the node in the workspace closest to the remote station to this terminal node with an edge cost equal to the expected distance until connectivity when moving straight towards the remote station from the node. This can be calculated based on the work in

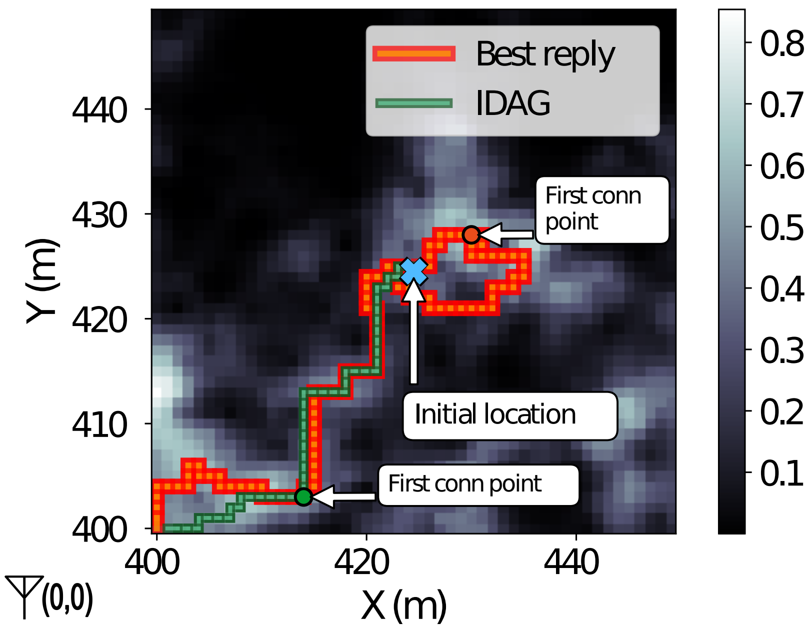


Figure 5.10: Solution paths produced by the best reply and IDAG approaches for a channel realization. Also shown is the first connected node on the respective paths for the true channel realization. The background plot denotes the predicted probability of connectivity, which is used by the robot for path planning.

Chapter 4.

| | Best reply | IDAG | Nearest neighbor | Closest terminal |
|---|---|---|---|---|
| Avg distance (m) | 28.40±25.93 | 32.90±26.12 | 44.17±56.22 | 50.24±30.38 |

Table 5.3: The average traveled distance along with the corresponding standard deviation, for our proposed approaches and for the greedy nearest neighbor and closest terminal heuristic approaches. The average is obtained by averaging over 500 channel realizations. We can see that our approaches results in a significant reduction in the traveled distance.

We next compare our proposed approaches with the greedy nearest neighbor heuristic as well as the closest terminal heuristic of moving straight towards the remote station. We calculate the performance of the approaches based on the true probability of connectivity of a node calculated based on the true value of the channel. Fig. 5.10 shows the solution path produced by the best reply and IDAG heuristic for a sample channel realization. The background plot denotes the predicted probability of connectivity. We see that

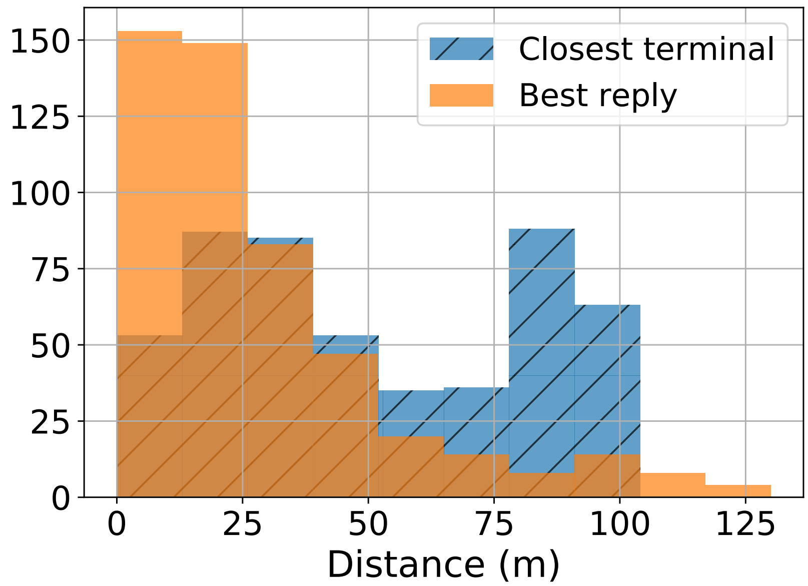


Figure 5.11: Histogram of the expected cost of the best reply and closest terminal heuristic over 500 channel realizations.

the paths produced take detours on the path to the connected point to visit areas of good probability of connectivity. Table. 5.3 shows the expected distance along with the corresponding standard deviation, for the best reply, IDAG, nearest neighbor and closest terminal approaches averaged over 500 channel realizations. We do not include the performance of log-linear as it takes longer to arrive at a good solution and is thus impractical to average over 500 channel realizations. However, in our simulations, we did observe that the performance of best reply was generally similar to the performance of log-linear learning. We see that the best reply and IDAG approach outperformed the nearest neighbor and closest terminal heuristics significantly. For instance, the best reply approach provided an overall 35% and 44% reduction in the expected traveled distance when compared to the nearest neighbor and closest terminal heuristics respectively. Fig. 5.11 shows the histogram of the expected cost of the best reply and closest terminal heuristic over the 500 channel realizations. We can see that the expected cost associated with the best reply heuristic is typically better than that associated with the closest terminal heuristic.

**Remark 5.9** *Note that our framework can be extended to the case where the robot updates the probabilities of success as it operates in the environment.*

# Chapter 6

# Conclusions and Future Work

In this dissertation, we showed how we can utilize the mobility of unmanned vehicles to enable and optimize communication. We proposed energy efficient path planning and communication strategies that exploit the mobility of unmanned vehicles in order to enable connectivity and new modes of communication.

We next summarize the results of each chapter and suggest some possible extensions for each topic.

## 6.1 Energy Optimal Distributed Beamforming using Unmanned Vehicles

In Chapter 3, we considered a motion and communication co-optimization problem where a team of unmanned vehicles were tasked with cooperatively beamforming common information to a remote station in a realistic communication environment, while minimizing the total energy consumption (both motion and communication energy). For the case where the channel is known, we found an $\epsilon$-suboptimal solution by proving that the original optimization problem can be posed as a series of multiple-choice knapsack problems. This solution provided the robots with the locations for distributed beamforming as well as the optimum transmission powers. We then extended our analysis and methodology to the case where the robots probabilistically predict the channel at unvisited locations. Finally, our simulation results showed the performance of the motion energy-aware and total energy-aware approaches for both perfect channel knowledge cases (MEMP and TEMP) as well as the stochastic cases (MESS and TESS). Overall, our results highlighted the underlying trends of the optimum strategy and indicated a considerable energy saving.

In Chapter 3, we have optimized for the final location of the robot and have not considered the channel quality along the path of the robot. An interesting future direction is to plan the path of the robots accounting for the statistics of the channel quality along the traveled path, perhaps by utilizing the results of Chapter 4 and Chapter 5.

## 6.2 Statistics of the Distance Traveled until Connectivity for Unmanned Vehicles

In Chapter 4, we considered the scenario of a robot that seeks to get connected to another robot or a remote operator, as it moves along a path. We started by mathematically characterizing the PDF of the distance traveled until connectivity along straight paths, using a stochastic differential equation approach when multipath can be ignored, and a recursive characterization method for the case of multipath. We then developed a theoretical characterization of a more general space of paths, based on properties of the path such as its curvature, for which we can theoretically characterize the PDF of the FPD. Our characterizations not only enable new theoretical analysis but also allow for an efficient low-complexity implementation. Finally, we confirmed our theoretical results with simulations with real channel parameters from downtown San Francisco, and highlighted interesting trends of the FPD.

Developing a low-computational complexity method for characterizing the statistics of the distance traveled until connectivity for paths that are not approximately-Markovian is an interesting future work direction.

## 6.3 Path Planning for Minimizing the Expected Cost until Success/Connectivity for Unmanned Vehicles

In Chapter 5, we considered the problem of path planning on a graph for minimizing the expected cost until success. We showed that this problem is NP-complete and that it can be posed in a Markov Decision Process framework as a stochastic shortest path problem. We then proposed a path planner based on a game-theoretic framework that yields an $\epsilon$-suboptimal solution to this problem asymptotically. In addition, we also proposed two non-myopic suboptimal strategies that find a good solution efficiently. Finally, through numerical results we showed that the proposed path planners outperform greedy heuristics significantly. For example, in numerical simulations, we considered the scenarios of a rover on mars searching for an object for scientific study. Our results then indicated a significant reduction in the expected cost incurred when using our proposed approaches.

We then applied our results to the scenario of planning an energy efficient path for a connectivity seeking robot, which is of particular relevance to this dissertation. Our numerical results, using real channel parameters from downtown San Francisco, indicated a significant reduction in the expected traveled distance (e.g., 35% reduction), when using our proposed approaches.

Some open questions and interesting directions to pursue in this area are:

- Developing algorithms with provable performance guarantees that run in polynomial time ($\alpha$-approximation algorithms) for the Min-Exp-Cost-Path problem.
- Development of fast solvers specifically tailored for the case where there are no terminal nodes, i.e., the Min-Exp-Cost-Path-NT problem.
- Applying the results in Chapter 5 to areas such as satisficing search and theorem solving [67].
- For the scenario of a connectivity seeking robot, we can use the analysis of Chapter 4 to obtain a more realistic characterization of the cost of a path. Then planning the path of a robot for establishing connectivity using this more realistic cost is an interesting direction for future work.

# Appendices

# Appendix A

## A.1 Extended Fenton-Wilkinson Method

Consider a lognormal random vector $\alpha = [\alpha_1 \cdots \alpha_N]^{\mathrm{T}}$ with distribution $[20\log_{10}\alpha_1 \cdots 20\log_{10}\alpha_N]^{\mathrm{T}} \sim \mathcal{N}(\overline{\Gamma}, \Sigma)$. The distribution can alternatively be expressed as $[\ln\alpha_1 \cdots \ln\alpha_N]^{\mathrm{T}} \sim \mathcal{N}(\xi\overline{\Gamma}, \xi^2\Sigma)$, where $\xi = 0.05\ln 10$. Let $\alpha_{\mathrm{sum}}$ with distribution $20\log_{10}\alpha_{\mathrm{sum}} \sim \mathcal{N}(\overline{\Gamma}_{\mathrm{sum}}, \sigma^2_{\mathrm{sum}})$ denote the lognormal random variable approximating $\sum_i \alpha_i$. The first and second moments of $\sum_i \alpha_i$ are given by

$$\mathbb{E}\left[\sum_i \alpha_i\right] = \sum_i e^{\xi\overline{\Gamma}_i + \xi^2\Sigma_{ii}/2}$$

and

$$\mathbb{E}\left[\left(\sum_i \alpha_i\right)^2\right] = \sum_i e^{2\xi\overline{\Gamma}_i + 2\xi^2\Sigma_{ii}} + 2\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} e^{\xi(\overline{\Gamma}_i + \overline{\Gamma}_j)} e^{\frac{\xi^2}{2}(\Sigma_{ii} + \Sigma_{jj} + 2\Sigma_{ij}))},$$

where $\overline{\Gamma}_i$ is the $i^{\mathrm{th}}$ entry of $\mu$ and $\Sigma_{ij}$ is the $ij^{\mathrm{th}}$ entry of $\Sigma$. In the extended Fenton-Wilkinson method [77], the first and second moments of $\alpha_{\mathrm{sum}}$ and $\sum_i \alpha_i$ are equated to obtain

$$\overline{\Gamma}_{\text{sum}} = \frac{1}{\xi}\left(2\ln\left(\mathbb{E}\left[\sum_i \alpha_i\right]\right) - \frac{1}{2}\ln\left(\mathbb{E}\left[\left(\sum_i \alpha_i\right)^2\right]\right)\right)$$

$$\sigma^2_{\text{sum}} = \frac{1}{\xi^2}\left(\ln\left(\mathbb{E}\left[\left(\sum_i \alpha_i\right)^2\right]\right) - 2\ln\left(\mathbb{E}\left[\sum_i \alpha_i\right]\right)\right).$$

## A.2 Proof of Lemma 3.4

We first prove the following lemma, which we shall use in proving Lemma 3.4.

**Lemma A.1** *Let* $\psi : \mathbb{R}^n_+ \to \mathbb{R}_+$ *with* $\psi(\rho) = \frac{\sum_{i=1}^n \rho_i^2}{\ln\left(1+\xi\left(\sum_{i=1}^n \alpha_i\rho_i\right)^2\right)}$ *where* $\xi, \alpha_i > 0$. *Given* $\rho$, *let* $\mathcal{I} = \{i : \frac{\rho_i}{\alpha_i} \geq \frac{\rho_j}{\alpha_j}\forall j\}$ *and let* $\nu \in \mathbb{R}^n$ *be such that its* $i^{th}$ *element is* $\nu(i) = \begin{cases} -\alpha_i, & \forall i \in \mathcal{I} \\ 0, & else \end{cases}$. *Then* $\psi(\rho)$ *is strictly decreasing in direction* $\nu$, *i.e.* $(\nabla\psi)^T\nu < 0$.

*Proof:*

$$\frac{\partial}{\partial\rho_k}\psi(\rho) = \frac{2\rho_k}{\ln\left(1+\xi\left(\sum_{i=1}^n \alpha_i\rho_i\right)^2\right)} - \frac{\left(\sum_{i=1}^n \rho_i^2\right)2\xi\left(\sum_{i=1}^n \alpha_i\rho_i\right)\alpha_k}{\left(1+\xi\left(\sum_{i=1}^n \alpha_i\rho_i\right)^2\right)\left[\ln\left(1+\xi\left(\sum_{i=1}^n \alpha_i\rho_i\right)^2\right)\right]^2}. \tag{A.1}$$

Let $y = \xi\left(\sum_{i=1}^n \alpha_i\rho_i\right)^2$. Also, $\alpha_k\left(\sum_{i=1}^n \rho_i^2\right) = \rho_k\left(\sum_{i=1}^n \alpha_i\rho_i\right) + \sum_{i=1}^n\left(\alpha_k\rho_i - \alpha_i\rho_k\right)\rho_i$. Inserting this in (A.1) and rearranging result in

$$\nabla\psi(\rho))^T\nu = \frac{2}{\ln(1+y)}\left\{\left(\sum_{k\in\mathcal{I}} -\alpha_k\rho_k\right)\left[1 - \frac{y}{(1+y)\ln(1+y)}\right]\right.$$
$$\left. + \frac{\xi\left(\sum_{i=1}^n \alpha_i\rho_i\right)\left(\sum_{i=1}^n\left[\left(\sum_{k\in\mathcal{I}}\alpha_k^2\right)\rho_i - \alpha_i\left(\sum_{k\in\mathcal{I}}\alpha_k\rho_k\right)\right]\rho_i\right)}{(1+y)\ln(1+y)}\right\}.$$

We have $\frac{d}{dy}\left((1+y)\ln(1+y)-y\right)=\ln(1+y)>0$ for $y>0$. Also, $(1+y)\ln(1+y)-y|_{y=0}=0$. Thus $(1+y)\ln(1+y)-y>0$, which results in $\frac{y}{(1+y)\ln(1+y)}<1$ for $y>0$. Since $\frac{\rho_k}{\alpha_k}=\frac{\rho_j}{\alpha_j}, \forall j,k\in\mathcal{I}$, we have

$$\begin{aligned}\frac{\sum_{k\in\mathcal{I}}\alpha_k\rho_k}{\sum_{k\in\mathcal{I}}\alpha_k^2}&=\frac{\alpha_j\rho_j+\sum_{k\in\mathcal{I},k\neq j}\alpha_k^2\frac{\rho_k}{\alpha_k}}{\sum_{k\in\mathcal{I}}\alpha_k^2}\\&=\frac{\rho_j}{\alpha_j}\geq\frac{\rho_i}{\alpha_i},\end{aligned}$$

for all $i$, and for $j\in\mathcal{I}$. Thus, we have $\left(\sum_{k\in\mathcal{I}}\alpha_k^2\right)\rho_i-\alpha_i\left(\sum_{k\in\mathcal{I}}\alpha_k\rho_k\right)\leq 0,\ \forall i$, resulting in $(\nabla\psi(\rho))^T\nu<0$. ■

*Proof:* [Proof of Lemma 3.4] We prove this by contradiction. Assume that $\sum_{i=1}^N\alpha(\overline{r}_{j_i^{\mathrm{OPT}}})\rho_i^{\mathrm{OPT}}>\alpha_{R,\mathrm{th}}$. Let $\mathcal{I}=\{i:\frac{\rho_i}{\alpha(\overline{r}_{j_i^{\mathrm{OPT}}})}\geq\frac{\rho_k}{\alpha(\overline{r}_{j_k^{\mathrm{OPT}}})}\forall k\}$ and let $\nu\in\mathbb{R}^N$ be such that $\nu(i)=\begin{cases}-\alpha(\overline{r}_{j_i^{\mathrm{OPT}}}), & \forall i\in\mathcal{I}\\ 0, & \text{else}\end{cases}$. We decrease $\rho$ in the direction of $\nu$ until either $\frac{\rho_i}{\alpha(\overline{r}_{j_i^{\mathrm{OPT}}})}=\frac{\rho_k}{\alpha(\overline{r}_{j_k^{\mathrm{OPT}}})}$ for some $i\notin\mathcal{I}$ $(k\in\mathcal{I})$, or $\sum_i\alpha(\overline{r}_{j_i^{\mathrm{OPT}}})\rho_i=\alpha_{R,\mathrm{th}}$. If $\frac{\rho_i}{\alpha(\overline{r}_{j_i^{\mathrm{OPT}}})}=\frac{\rho_k}{\alpha(\overline{r}_{j_k^{\mathrm{OPT}}})}$ for some $i\notin\mathcal{I}$ $(k\in\mathcal{I})$, we add $i$ to $\mathcal{I}$, update our $\nu$, and continue decreasing $\rho$. If $\sum_i\alpha(\overline{r}_{j_i^{\mathrm{OPT}}})\rho_i=\alpha_{R,\mathrm{th}}$, we terminate our update.

Let $\nabla_\rho J_{\mathrm{TEMP}}=[\frac{\partial}{\partial\rho_1}J_{\mathrm{TEMP}}\cdots\frac{\partial}{\partial\rho_N}J_{\mathrm{TEMP}}]^{\mathrm{T}}$. From Lemma A.1, we have $(\nabla_\rho J_{\mathrm{TEMP}})^T\nu<0$ and hence $J_{\mathrm{TEMP}}(\{z_{ij}^{\mathrm{OPT}}\},\rho)<J_{\mathrm{TEMP}}(\{z_{ij}^{\mathrm{OPT}}\},\rho^{\mathrm{OPT}})$. Also, $\rho$ is a feasible solution since $\sum_i\alpha(\overline{r}_{j_i^{\mathrm{OPT}}})\rho_i=\alpha_{R,\mathrm{th}}$. We thus have a contradiction. ■

## A.3 Proof of Lemma 3.5

*Proof:* We introduce Lagrange multipliers $\xi,\iota\in\mathbb{R}^N$ for the inequality constraints $\rho_i\leq 1$ and $\rho_i\geq 0$ respectively, and $\lambda_0\in\mathbb{R}$ for the constraint $\sum_i\alpha_i\rho_i\geq\alpha_{R,\mathrm{th}}$. The Lagrangian is given as $L(\rho,\lambda_0,\xi,\iota)=\sum_i\rho_i^2+\sum_i\xi_i(\rho_i-1)+\sum_i\iota_i(-\rho_i)+\lambda_0\left(\alpha_{R,\mathrm{th}}-\sum_i\alpha_i\rho_i\right)$.

We have the following KKT conditions:

$$2\rho_i^* - \lambda_0^*\alpha_i + \xi_i^* - \iota_i^* = 0, \quad \xi_i^*(\rho_i^* - 1) = 0,$$
$$\iota_i^*\rho_i^* = 0, \quad \lambda_0^*\big(\alpha_{R,\text{th}} - \sum_i \alpha_i\rho_i^*\big) = 0,$$
$$\xi^* \succeq 0, \quad \iota^* \succeq 0, \quad \lambda_0^* > 0, \quad 0 \le \rho_i^* \le 1, \quad \sum_i \alpha_i\rho_i^* \ge \alpha_{R,\text{th}}.$$

Assume $\iota_i^* > 0$ for some $i$. Then $\rho_i^* = 0$ and thus $\xi_i^* = 0$. But $\rho_i^* = \frac{\lambda_0^*}{2}\alpha_i - \frac{\xi_i^*}{2} + \frac{\iota_i^*}{2} = \frac{\lambda_0^*}{2}\alpha_i + \frac{\iota_i^*}{2} > 0$, resulting in a contradiction. Therefore, $\iota_i^* = 0, \forall i$. Assume $\lambda_0^* = 0$. Then $\rho_i^* = -\frac{\xi_i^*}{2} = 0$, and hence $\sum_i \alpha_i\rho_i^* = 0 < \alpha_{R,\text{th}}$, resulting in a contradiction. Thus $\sum_i \alpha_i\rho_i^* = \alpha_{R,\text{th}}$. If $\frac{\lambda_0^*}{2}\alpha_i > 1$ then $\xi_i^* > 0$ which in turn implies that $\rho_i^* = 1$. If $\frac{\lambda_0^*}{2}\alpha_i < 1$, then $\xi_i^* = 0$ and hence $\rho_i^* = \frac{\lambda_0^*}{2}\alpha_i$. Thus, $\rho_i^* = \min\{\lambda\alpha_i, 1\}$, where $\lambda = \frac{\lambda_0^*}{2} > 0$ is such that $\sum_i \min\{\lambda\alpha_i, 1\}\alpha_i = \alpha_{R,\text{th}}$. ■

# Appendix B

## B.1 Proof of Lemma 4.2

*Proof:* Let $r(b) = (x(b), y(b))$ be the equation of the path parameterized by arc length. Since the path is parameterized by arc length, we have

$$\|r'(b)\|^2 = |x'(b)|^2 + |y'(b)|^2 = 1. \tag{B.1}$$

Moreover, we have the curvature constraint

$$\|r''(b)\|^2 = |x''(b)|^2 + |y''(b)|^2 \leq \kappa^2. \tag{B.2}$$

Let $b_0$ denote the current point, i.e., the center of the ball. Without loss of generality, let $(x(b_0), y(b_0)) = (0, 0)$ and let the tangent at $b_0$ be parallel to the x-axis, i.e., $x'(b_0) = -1$, $y'(b_0) = 0$, as shown in Fig. 4.4.

We first prove that no point of $r_{\text{ball}}$ can lie outside the shaded region of Fig. 4.4. Note that the shaded region has a boundary on the left corresponding to $x = -d_{\text{th}}$, and the two other boundaries correspond to circular arcs with curvature $\kappa$. Let us consider traveling backward along the path. For a given distance $d_x$ traveled along the negative x-axis (i.e., $x(b) = -d_x$), the path which maximizes the distance traveled along the y-axis $|y(b)|$, is the one that minimizes the x-axis velocity $|x'(b)|$ and maximizes the y-axis velocity $|y'(b)|$ the most. This corresponds to the circular path $(R_c \cos(b/R_c), R_c \sin(b/R_c))$ with constant curvature $\kappa$. Thus, for any path satisfying (B.1) and (B.2), the y-axis coordinate is bounded above and below by the circular arc. This implies that the segment $r_{\text{ball}}$ lies within the shaded region.

We next show that if $\kappa < 1/d_{\text{th}}$, then $r_{\text{ball}}$ cannot loop within the ball. Note that, by definition, $r_{\text{ball}}$ loops within the ball if $x'(b) > 0$ for some point on the path within the shaded region. The circular path with curvature $\kappa$ is the path that maximizes $x'(b)$. From Fig. 4.4, we can see that if $\kappa = 1/d_{\text{th}}$, then $x'(b) = 0$ at $x(b) = -d_{\text{th}}$ for the circular path. Thus, if $\kappa < 1/d_{\text{th}}$, we have $x'(b) > 0$ for any point of the path within the shaded region.

Finally, we determine the bound on the length of $r_{\text{ball}}$. If we travel a distance of $d_{\text{th}}$ along the negative x-axis, then we are guaranteed to have exit the ball. The path that maximizes its length before covering $d_{\text{th}}$ along the negative x-axis, would be the one that reduces the x-axis velocity $|x'(b)|$ the most. This maximal length path corresponds to the circular path with constant curvature $\kappa$. Any other path satisfying (B.1) and (B.2) would exit the shaded region before this circular path, i.e., the length of the segment of any path would be less that the length of this circular arc. The length of this circular arc can be found from the geometry of the figure. The chord length can be seen to be $2R_c \sin(\phi/2)$ where $R_c = 1/\kappa$. Moreover, we have $\cos(\phi/2) = \frac{d_{\text{th}}}{2R_c \sin(\phi/2)}$ which implies that $\phi = \sin^{-1}\left(\frac{d_{\text{th}}}{R_c}\right)$. This gives us the arc length as $2\pi R_c \times \frac{\phi}{2\pi} = R_c \sin^{-1}\left(\frac{d_{\text{th}}}{R_c}\right) = \frac{1}{\kappa}\sin^{-1}(\kappa d_{\text{th}})$. ■

## B.2 Proof of Lemma 4.5

*Proof:* Using (4.8), we can show that $m = c_1\Gamma_{\mathrm{SH},-1} + c_r\Gamma_{\mathrm{SH},r}$ where

$$c_1 = \frac{e^{-d_1/\beta_{\mathrm{SH}}} - e^{-(d_{1r}+d_r)/\beta_{\mathrm{SH}}}}{1 - e^{-2d_{1r}/\beta_{\mathrm{SH}}}},$$

$$c_r = \frac{e^{-d_r/\beta_{\mathrm{SH}}} - e^{-(d_1+d_{1r})/\beta_{\mathrm{SH}}}}{1 - e^{-2d_{1r}/\beta_{\mathrm{SH}}}}.$$

Then, the difference in mean $\Delta m = m - \hat{m}$ is distributed as $\mathcal{N}(0, \sigma^2_{\Delta m})$, where using (4.10) we have

$$\sigma^2_{\Delta m} = \sigma^2_{\mathrm{SH}} \frac{\left(e^{-d_r/\beta_{\mathrm{SH}}} - e^{-(d_1+d_{1r})/\beta_{\mathrm{SH}}}\right)^2}{1 - e^{-2d_{1r}/\beta_{\mathrm{SH}}}}.$$

Moreover, using (4.9) we can calculate

$$\frac{\sigma^2}{\sigma^2_{\mathrm{SH}}} = 1 - \frac{e^{-2d_1/\beta_{\mathrm{SH}}} + e^{-2d_r/\beta_{\mathrm{SH}}} - 2e^{-(d_1+d_r+d_{1r})/\beta_{\mathrm{SH}}}}{1 - e^{-2d_{1r}/\beta_{\mathrm{SH}}}}.$$

The difference in variance $\Delta\sigma^2 = \sigma^2 - \hat{\sigma}^2$ can be calculated as

$$\begin{aligned}\Delta\sigma^2 &= -\sigma^2_{\mathrm{SH}} \frac{\left(e^{-d_r/\beta_{\mathrm{SH}}} - e^{-(d_1+d_{1r})/\beta_{\mathrm{SH}}}\right)^2}{1 - e^{-2d_{1r}/\beta_{\mathrm{SH}}}} \\ &= -\sigma^2_{\Delta m}.\end{aligned}$$

From (4.11), we then have

$$KL = \frac{\sigma^2_{\Delta m}}{2\hat{\sigma}^2}\chi^2_1 + \frac{1}{2}\left(-\frac{|\Delta\sigma^2|}{\hat{\sigma}^2} - \log_e\left(1 - \frac{|\Delta\sigma^2|}{\hat{\sigma}^2}\right)\right).$$

Since $\mathbb{E}[\chi_1^2] = 1$ and $\mathrm{Var}[\chi_1^2] = 2$, we can calculate the mean $m_{KL}$ and the standard deviation $\sigma_{KL}$ to be as stated in the lemma. ■

## B.3 Proof of Lemma 4.6

*Proof:* Consider all possible locations of the general point (see Fig. 4.5 (top)) at a fixed distance $d_r$. From the geometry of Fig. 4.5 (top), we can see that $d_{1r} = \sqrt{d_1^2 + d_r^2 - 2d_1 d_r \cos\theta}$. Varying $\theta$, results in varying $d_{1r}$ which can take values in $[d_r - d_1, d_r + d_1]$. From Lemma 4.5, we can see that the $\theta$ that has a maximum impact on the KL divergence is the one that would minimize $m_{KL}$ and $\sigma_{KL}$. This would occur when we maximize $\sigma_{\Delta m}^2 = \sigma_{\mathrm{SH}}^2 e^{-d_r/\beta_{\mathrm{SH}}} \frac{(1-e^{-(z-z_l)})^2}{1-e^{-2z}}$ where $z = d_{1r}/\beta_{\mathrm{SH}}$ and $z_l = (d_r - d_1)/\beta_{\mathrm{SH}}$. We wish to maximize $\psi(z) = \frac{(1-e^{-(z-z_l)})^2}{1-e^{-2z}}$. Taking it's derivative gives us

$$\frac{\mathrm{d}}{\mathrm{d}z}\psi(z) = \frac{2(1-e^{-(z-z_l)})}{(1-e^{-2z})^2}(e^{-(z-z_l)} - e^{-2z}).$$

Then $\frac{\mathrm{d}}{\mathrm{d}z}\psi(z) > 0$ if $z > -z_l$, which is true as long as $d_r > d_1$.

Thus, maximizing $\sigma_{\Delta m}^2$ occurs at $\theta = \pi$ where $d_{1r}$ takes its maximum value of $d_1 + d_r$. Setting $\theta = \pi$ gives us

$$\sigma_{\Delta m}^2 = \sigma_{\mathrm{SH}}^2 \frac{\left(e^{-d_r/\beta_{\mathrm{SH}}} - e^{-(2d_1+d_r)/\beta_{\mathrm{SH}}}\right)^2}{1 - e^{-2(d_1+d_r)/\beta_{\mathrm{SH}}}}.$$

From Lemma 4.5, we can see that satisfying the KL divergence parameters implies that $\frac{\sigma_{\Delta m}^2}{\hat{\sigma}^2} \leq 1 - e^{-2\epsilon_m}$, and $\frac{\sigma_{\Delta m}^2}{\hat{\sigma}^2} \leq \sqrt{2}\epsilon_\sigma$. Let $\epsilon_d = \min\left\{1 - e^{-2\epsilon_m}, \sqrt{2}\epsilon_\sigma\right\}$. Thus, we obtain the constraint

$$\frac{e^{-2d_r/\beta_{\mathrm{SH}}}(1-\rho^2)^2}{(1-\rho^2 e^{-2d_r/\beta_{\mathrm{SH}}})(1-\rho^2)} \leq \epsilon_d,$$

which in turn gives us the constraint

$$d_r \geq \frac{\beta_{\mathrm{SH}}}{2} \log_e \left( \rho^2 + \frac{1-\rho^2}{\epsilon_d} \right).$$

■

## B.4 Proof of Lemma 4.7

*Proof:* Consider the scenario of Fig. 4.5 (bottom left) where $d_1 = \Delta d$. We will choose the location of the general point $(\Gamma_{\mathrm{SH},r})$, which lies within the shaded region, such that it maximizes the impact (in terms of the KL divergence) on the approximation. From Lemma 4.5, we can see that the point that has a maximum impact on the KL divergence is the one that would maximize $\sigma^2_{\Delta m}$. From the proof of Lemma 4.6, we know that for a fixed $d_r$ and varying $\theta$, the maximum value of $\sigma^2_{\Delta m}$ occurs at the maximum value of $d_{1r}$. This occurs at the boundary of the shaded region, i.e., at a point on the circular arc. Since this holds for all $d_1 < d_r \leq d_{\mathrm{th}}$, we know that the point that maximizes $\sigma^2_{\Delta m}$ lies on the circular path with constant curvature $\kappa$.

We thus consider the setting in Fig. 4.5 (bottom right) with a fixed curvature $\kappa$. From the geometry of the figure, we have the following relations: $d_1 = 2R_c \sin\left(\frac{\Delta\phi}{2}\right)$, $d_{1r} = 2R_c \sin\left(\frac{\phi}{2}\right)$ and $d_r = 2R_c \sin\left(\frac{\phi+\Delta\phi}{2}\right)$. Since $d_1 = \Delta d$, we have $\Delta\phi = 2\sin^{-1}(\kappa\Delta d/2)$. From Lemma 4.3, we have the constraint that $\kappa < 1/d_{\mathrm{th}}$. This guarantees that the path will leave the ball. Moreover, from the geometry of the figure, we can see that this will occur at the angle $\phi$ such that $d_r = 2R_c \sin\left(\frac{\phi+\Delta\phi}{2}\right) = d_{\mathrm{th}}$. This occurs at $\phi = \psi_{\mathrm{cons}}(\kappa) = 2\sin^{-1}(\frac{\kappa d_{\mathrm{th}}}{2}) - \Delta\phi$.

From Lemma 4.5, we can see that satisfying the KL divergence parameters implies that $\frac{\sigma^2_{\Delta m}}{\hat{\sigma}^2} \leq 1 - e^{-2\epsilon_m}$, and $\frac{\sigma^2_{\Delta m}}{\hat{\sigma}^2} \leq \sqrt{2}\epsilon_\sigma$. Let $\epsilon_d = \min\left\{1 - e^{-2\epsilon_m}, \sqrt{2}\epsilon_\sigma\right\}$. Thus, the point on the path that maximizes the KL divergence occurs at the angle

$$\arg \max_{0<\phi\leq\psi_{\text{cons}}(\kappa)} \psi_{\text{opt}}(\kappa,\phi),$$

where

$$\begin{aligned}\psi_{\text{opt}}(\kappa,\phi) &= \frac{\sigma_{\Delta m}^2}{\hat{\sigma}^2}\\ &= \frac{\left(e^{-\frac{2}{\kappa\beta_{\text{SH}}}\sin(\frac{\phi+\Delta\phi}{2})} - \rho e^{-\frac{2}{\kappa\beta_{\text{SH}}}\sin(\frac{\phi}{2})}\right)^2}{(1-e^{-\frac{4}{\kappa\beta_{\text{SH}}}\sin(\frac{\phi}{2})})(1-\rho^2)}.\end{aligned}$$

We wish to find the maximum curvature $\kappa$, such that this maximum impact still satisfies the KL divergence parameters, i.e.,

$$\max_{0<\phi\leq\psi_{\text{cons}}(\kappa)} \psi_{\text{opt}}(\kappa,\phi) \leq \epsilon_d.$$

This results in the optimization problem stated in the lemma. ■

# Appendix C

## C.1 Proof of Lemma 5.5

We first describe some properties of the solution of Min-Exp-Cost-Path and Min-Exp-Cost-Simple-Path.

**Definition C.1** *Consider a path $\mathcal{P} = (v_1, v_2, \cdots, v_m)$. A node $v_i$ is a* revisited *node in the $i^{th}$ location of $\mathcal{P}$ if $v_i = v_j$ for some $j < i$. A node $v_i$ is a* first-visit *node in the $i^{th}$ location of $\mathcal{P}$ if $v_i \neq v_j$ for all $j < i$.*

**Property C.1** *Let $\mathcal{P}^* = (v_1, v_2, \cdots, v_m)$ be a solution to Min-Exp-Cost-Path on $\mathcal{G}$. Consider any subpath $(v_i, v_{i+1}, \cdots, v_{j-1}, v_j)$ of $\mathcal{P}^*$ such that $v_i$ and $v_j$ are first-visit nodes, and $v_{i+1}, \cdots, v_{j-1}$ are revisited nodes. Then, $(v_i, v_{i+1}, \cdots, v_{j-1}, v_j)$ is the shortest path between $v_i$ and $v_j$.*

*Proof:* We show this by contradiction. Assume otherwise, i.e., $(v_i, v_{i+1}, \cdots, v_{j-1}, v_j)$ is not the shortest path between $v_i$ and $v_j$. Let $\mathcal{Q}$ be the path produced by replacing this subpath in $\mathcal{P}^*$ with the shortest path between $v_i$ and $v_j$. Let us denote this shortest path by $(v_i, u_{i+1}, \cdots, u_{\tilde{j}-1}, u_{\tilde{j}})$ where $u_{\tilde{j}} = v_j$. Then,

$$\begin{aligned} C(\mathcal{Q}, i) &= (1 - p_{v_i}) \Bigg[ l_{v_i u_{i+1}} + \Big[ \prod_{k \in \mathcal{L}_{i+1}} (1 - p_{u_k}) \Big] l_{u_{i+1} u_{i+2}} \\ &\quad + \cdots + \Big[ \prod_{k \in \mathcal{L}_{\tilde{j}-1}} (1 - p_{u_k}) \Big] \Big[ l_{u_{\tilde{j}-1} u_{\tilde{j}}} + C(\mathcal{Q}, \tilde{j}) \Big] \Bigg] \\ &\leq (1 - p_{v_i}) \Bigg[ l^{\min}_{v_i v_j} + \Big[ \prod_{k \in \mathcal{L}_{\tilde{j}-1}} (1 - p_{u_k}) \Big] C(\mathcal{Q}, \tilde{j}) \Bigg], \end{aligned}$$

where $\mathcal{L}_m = \{k \in \{i+1, \cdots, m\} : u_k$ is a first visit node of $\mathcal{Q}\}$. The nodes $(u_{i+1}, \cdots, u_{\tilde{j}})$ could be first visit nodes of $\mathcal{Q}$ or repeated nodes. We next show that in either scenario the expected cost of $\mathcal{Q}$ would be smaller than that of $\mathcal{P}$. If they are all revisited nodes or if they are first-visit nodes that are not revisited after node $u_{\tilde{j}}$, then $C(\mathcal{Q}, \tilde{j}) = C(\mathcal{P}^*, j)$. If some or all of $(u_{i+1}, \cdots, u_{\tilde{j}})$ are first-visit nodes of $\mathcal{Q}$ that are visited later on, then $[\prod_{k \in \mathcal{L}_{\tilde{j}-1}} (1 - p_{u_k})] C(\mathcal{Q}, \tilde{j}) \leq C(\mathcal{P}^*, j)$, since success at a first visit node $u_k$ can occur earlier in path $\mathcal{Q}$ in comparison to $\mathcal{P}^*$ (which discounts the cost of all following edges). Thus, in either case, we have the inequality

$$\begin{aligned} C(\mathcal{Q}, i) &\leq (1 - p_{v_i}) \left[ l^{\min}_{v_i v_j} + C(\mathcal{P}^*, j) \right] \\ &< (1 - p_{v_i}) \left[ l_{v_i v_{i+1}} + \cdots + l_{v_{j-1} v_j} + C(\mathcal{P}^*, j) \right] \\ &= C(\mathcal{P}^*, i). \end{aligned}$$

This implies that $C(\mathcal{Q}, 1) < C(\mathcal{P}^*, 1)$ resulting in a contradiction. ■

**Property C.2** *Let $\mathcal{P}^* = (v_1, v_2, \cdots, v_m)$ be a solution of Min-Exp-Cost-Simple-Path on complete graph $\mathcal{G}_{comp}$. Consider any two consecutive nodes $v_i$ and $v_{i+1}$. The shortest path between $v_i$ and $v_{i+1}$ in $\mathcal{G}$ would only consist of nodes that have been visited earlier in $\mathcal{P}^*$.*

*Proof:* Suppose this is not true for consecutive nodes $v_i$ and $v_{i+1}$. Then there exists at least a single node $u$ that lies on the shortest path between $v_i$ and $v_{i+1}$, and that has not been visited earlier in $\mathcal{P}^*$. Let $\mathcal{Q}$ be the path formed from $\mathcal{P}^*$ when $u$ is added between $v_i$ and $v_{i+1}$. The expected cost of $\mathcal{Q}$ from the $i^{\text{th}}$ node onwards is given by

$$\begin{aligned} C(\mathcal{Q}, i) &= (1 - p_{v_i}) \left[ l_{v_i u} + (1 - p_u) \left[ l_{u v_{i+1}} + C(\mathcal{Q}, i+2) \right] \right] \\ &< (1 - p_{v_i}) \left[ l_{v_i v_{i+1}} + C(\mathcal{P}^*, i+1) \right]. \end{aligned}$$

This implies that the expected cost of $Q$ would be less than that of $\mathcal{P}^*$, resulting in a contradiction. ■

*Proof:* [Proof of Lemma 5.5] Let $\mathcal{P}$ be the solution to Min-Exp-Cost-Path on $\mathcal{G}$ and let $\mathcal{Q}$ be the solution of the Min-Exp-Cost-Simple-Path on $\mathcal{G}_{\text{comp}}$. From Property C.1, we know that the path produced by removing revisited nodes in $\mathcal{P}$, will produce a feasible solution to Min-Exp-Cost-Simple-Path on $\mathcal{G}_{\text{comp}}$ with the same cost as $\mathcal{P}$. Thus, the cost of $\mathcal{P}$ is greater than or equal that of $\mathcal{Q}$. Similarly, from Property C.2, we know that the path produced by expanding the shortest path between any adjacent nodes in $\mathcal{Q}$, will be a feasible solution to Min-Exp-Cost-Path on $\mathcal{G}$ with the same cost as $\mathcal{Q}$. Thus, this path produced from $\mathcal{Q}$ will be an optimal solution to Min-Exp-Cost-Path on $\mathcal{G}$. ■

## C.2 Proof of Theorem 5.2

Log-linear learning induces a Markov process on the action profile space $A_{\text{ASG}} \cup A_{\emptyset}$, where $A_{\emptyset} = \{\mu : \mu_v = a_{\emptyset} \text{ for some } v\}$. In the following lemma, we first show that $A_{\text{ASG}}$ is a closed communicating recurrent class.

**Lemma C.1** *$A_{ASG}$ is a closed communicating recurrent class.*

*Proof:* We first show that $A_{\text{ASG}}$ is a communicating class, i.e., there is a finite transition sequence from $\mu^s$ to $\mu^f$ with non-zero probability for all $\mu^s, \mu^f \in A_{\text{ASG}}$. Consider the set of states $\mathcal{R}_0, \mathcal{R}_1, \cdots$, defined by the recursion $\mathcal{R}_{k+1} = \{v : \mu_v^f \in \mathcal{R}_k\}$, where $\mathcal{R}_0 = T$, i.e., $\mathcal{R}_k$ is the set of all nodes that are $k$ hops away from the set of terminal nodes $T$ in the ASG $\mathcal{SG}(\mu^f)$. Let $\bar{k}$ be the last of the sets that is non-empty. Since $\mu^f \in A_{\text{ASG}}$, we have $\bar{k} \leq |\mathcal{V}|$ and $\cup_{m=0}^{\bar{k}} \mathcal{R}_m = \mathcal{V}$. We transition from $\mu^s$ to $\mu^f$ by sequentially switching from $\mu_v = \mu_v^s$ to $\mu_v = \mu_v^f$, for all $v \in \mathcal{R}_k$, starting at $k = 1$ and incrementing $k$ until $k = \bar{k}$, i.e., we first change the action of nodes in $\mathcal{R}_1$, and then $\mathcal{R}_2$ and so on until $\mathcal{R}_{\bar{k}}$. We next show that this transition sequence has a non-zero probability by showing that each component transition has a non-zero probability. At stage $k+1$, consider the transition where we switch the action of a node $v \in \mathcal{R}_k$, and let $\mu$ be the current action. At this stage we have already changed the action of players in $\mathcal{R}_1, \cdots, \mathcal{R}_k$, and for the current graph $\mathcal{SG}(\mu)$, there is a path leading from $\mu_v^f \in \mathcal{R}_k$ to a terminal node in $\mathcal{R}_0$. Moreover, $\mu_v^f$ is not upstream of $v$ since the intermediate nodes of the path are in $\mathcal{R}_{k-1}, \cdots, \mathcal{R}_1$. Then, $\mu_v^f \in A_v^c(\mu_{-v})$, which implies that the transition $(\mu_v^s, \mu_{-v}) \to (\mu_v^f, \mu_{-v})$ has a non-zero probability. Thus, $A_{\text{ASG}}$ is a communicating class.

We next show that $A_{\text{ASG}}$ is closed. Consider a state $\mu \in A_{\text{ASG}}$, and a node $v \in \mathcal{V}'$. Then, $A_v^c(\mu_{-v})$ is not empty, since $\mu_v \in A_v^c(\mu_{-v})$. This implies that $\mu_v$ can not be set as the null action $a_\emptyset$. Thus, $A_{\text{ASG}}$ is closed. Since $A_{\text{ASG}}$ is a closed communicating class, every action profile $\mu \in A_{\text{ASG}}$ is a recurrent state. ■

We next show, in the following lemma, that all states in $A_\emptyset$ are transient states.

**Lemma C.2** *Any state $\mu \in A_\emptyset$ is a transient state.*

*Proof:* Consider a state $\mu^s \in A_\emptyset$ and a state $\mu^f \in A_{\text{ASG}}$. We can design a transition sequence of non-zero probability from $\mu^s$ to $\mu^f$ similar to how we did so in the proof of Lemma C.1, as the sequence designed did not depend on $\mu^s$. Moreover, from Lemma C.1, we know that $A_{\text{ASG}}$ is a closed class. Thus, there is a finite non-zero probability that the state $\mu^s \in A_\emptyset$ will never be revisited. ■

*Proof:* [Proof of Theorem 5.2] From Lemma C.1 and Lemma C.2, we know that there is exactly one closed communicating recurrent class. Thus, the stationary distribution of the Markov chain induced by log-linear learning is unique. The transition probability from state $\mu$ to $\mu^{'} = (\mu_v^{'}, \mu_{-v})$ for $\mu, \mu^{'} \in A_{\text{ASG}}$ is given as

$$P_{\mu\mu'} = \frac{1}{|\mathcal{V}'|} \frac{e^{-\frac{1}{\tau}\left(\mathcal{J}_v(\mu_v^{'}, \mu_{-v})\right)}}{\sum_{\mu_v^{''} \in A_v^c(\mu_{-v})} e^{-\frac{1}{\tau}\left(\mathcal{J}_v(\mu_v^{''}, \mu_{-v})\right)}},$$

denote . We can reformulate this as

$$P_{\mu\mu'} = \frac{1}{|\mathcal{V}'|} \frac{e^{-\frac{1}{\tau}\left(\phi(\mu_v^{'}, \mu_{-v})\right)}}{\sum_{\mu_v^{''} \in A_v^c(\mu_{-v})} e^{-\frac{1}{\tau}\left(\phi(\mu_v^{''}, \mu_{-v}(k))\right)}},$$

using $\mathcal{J}_v(\mu_v^{'}, \mu_{-v}) - \mathcal{J}_v(\mu_v, \mu_{-v}) = \phi(\mu_v^{'}, \mu_{-v}) - \phi(\mu_v, \mu_{-v})$ from Lemma 5.6. Then, we can see that the probability distribution $\Pi \in \Delta(A_{\text{ASG}})$ given by

$$\Pi(\mu) = \frac{e^{-\frac{1}{\tau}\phi(\mu)}}{\sum_{\mu'' \in A_{\text{ASG}}} e^{-\frac{1}{\tau}\phi(\mu'')}},$$

satisfies the detailed balance equation $\Pi_\mu P_{\mu\mu'} = \Pi_{\mu'} P_{\mu'\mu}$. Thus, $\Pi$ is the unique stationary distribution. As temperature $\tau \to 0$, the weight of the stationary distribution will be on the global minimizers of the potential function [106]. In other words, $\lim_{\tau\to 0} \sum_{\mu \in \arg\min_{\mu' \in A_{\text{ASG}}} \phi(\mu')} \Pi(\mu) = 1$. Thus, asymptotically, log-linear learning provides us with the global minimizer of $\phi(\mu) = C_{v_s}(\mu) + \epsilon^{'} \sum_{v \neq v_s} C_v(\mu)$, an $\epsilon$-suboptimal solution to the Min-Exp-Cost-Simple-Path problem. ∎

## C.3 Proof of Theorem 5.3

*Proof:* We first show that there exists an $k_l$ such that $\mu(k) \in A_{\text{ASG}}$ for all $k \geq k_l$. Let $A_\emptyset = \{\mu : \mu_v = a_\emptyset \text{ for some } v\}$ denote the set of action profiles with at least one player playing a null action. Consider a action profile $\mu \in A_\emptyset$. Then there must exist a node $v \in \{u : \mu_u = a_\emptyset\}$ which has a neighbor in $T \cup \{u : \mu_u \neq a_\emptyset\}$, since otherwise $\{u : \mu_u = a_\emptyset\}$ and $T \cup \{u : \mu_u \neq a_\emptyset\}$ are not connected, contradicting the assumption that the graph is connected. Then, $A_v^c(\mu_{-v})$ is non-empty, and when node $v$ is selected in the round robin iteration it will play a non-null action. Moreover, $\mu_v \neq a_\emptyset$ for all subsequent iterations, since its current action at any iteration $k$ will always belong to $A_v^c(\mu_{-v}(k))$. We can apply this reasoning repeatedly to show that eventually at some iteration $k_l$ the set $\{u : \mu_u(k_l) = a_\emptyset\}$ will be empty, i.e., $\mu(k_l) \in A_{\text{ASG}}$. Furthermore, $\mu(k) \in A_{\text{ASG}}$ for all $k \geq k_l$.

We next prove that $C_v(\mu(k+1)) \leq C_v(\mu(k))$ for all $v \in \mathcal{V}'$ and for all $k$. Let $v$ be the node selected at stage $k+1$. Clearly, if $\mu_v(k) = a_\emptyset$ this is true. Else,

$$\begin{aligned} C_v(\mu(k+1)) &= \min_{u \in A_v^c(\mu_{-v}(k))} \{(1-p_v)\left[l_{vu} + C_u(\mu(k))\right]\} \\ &\leq (1-p_v)\left[l_{v\mu_v(k)} + C_{\mu_v(k)}(\mu(k))\right] \qquad \text{(C.1)} \\ &= C_v(\mu(k)), \end{aligned}$$

where (C.1) follows since $\mu_v(k) \in A_v^c(\mu_{-v}(k))$. From (5.4), we have that $C_u(\mu(k+1)) \leq C_u(\mu(k)),\ \ \forall u \in U_v(\mu)$, where $U_v(\mu)$ is the set of upstream nodes from $v$. Furthermore, $C_u(\mu(k+1)) = C_u(\mu(k)),\ \ \forall u \notin U_v(\mu)$. Thus, $C_v(\mu(k+1)) \leq C_v(\mu(k))$ for all $v \in \mathcal{V}'$.

Since $\{C_v(\mu(k))\}_k$ is a monotonically non-increasing sequence, bounded by below from 0, we know that the limit exists. Moreover, since $\mu$ belongs to a finite space, we know that convergence must occur in a finite number of iterations. It should be noted however, that the limit can be different based on the order of the nodes in the round robin. Let $\mu^* \in A_{\text{ASG}}$ denote the solution at convergence for the particular order of nodes. We assume that, when selecting $\mu_v$, ties are broken using a consistent set of rules, since otherwise we may cycle repeatedly through action profiles having the same expected costs $\{C_v(\mu)\}_v$.

We next show that we converge to this limit in $|\mathcal{V}'|^2$ iterations. Let $n = |\mathcal{V}'|$. Consider the set of states $\mathcal{R}_0, \mathcal{R}_1, \cdots$, defined by the recursion $\mathcal{R}_{k+1} = \{v : \mu_v^* \in \mathcal{R}_k\}$, where $\mathcal{R}_0 = T$, i.e., $\mathcal{R}_k$ is the set of all nodes that are $k$ hops away from the set of terminal nodes $T$ in the ASG $\mathcal{SG}(\mu^*)$. Let $\bar{k}$ be the last of the sets that is non-empty. Since $\mu^* \in A_{\text{ASG}}$, we have $\bar{k} \leq n$ and $\cup_{m=0}^{\bar{k}} \mathcal{R}_m = \mathcal{V}$. We next show by induction that $\mu_v(nk) = \mu_v^*, \ \forall v \in \cup_{m=0}^{k} \mathcal{R}_m$, for $k = 0, 1, \cdots, \bar{k}$. This is true for $k = 0$. Assume that it holds true at stage $k$, i.e., $\mu_v(nk) = \mu_v^*$ for all $v \in \cup_{m=1}^{k} \mathcal{R}_0$. Since $\{C_v(\mu(k))\}_k$ is monotonically non-increasing, we have $C_v(\mu^*) \leq C_v(\mu(k+1))$. Moreover, since any node $v \in \cup_{m=0}^{k+1} \mathcal{R}_m$ would be selected once in round $k+1$ of the round robin process, we have

$$\begin{aligned} C_v(\mu(n(k+1))) &= \min_{u \in A_v^c(\mu_{-v}(n(k+1)-1))} \Big\{ (1-p_v) \times \big[ l_{vu} + C_u(\mu(n(k+1)-1)) \big] \Big\} \\ &\leq (1-p_v) \big[ l_{v\mu_v^*} + C_{\mu_v}(\mu^*) \big] \qquad \text{(C.2)} \\ &= C_v(\mu^*). \end{aligned}$$

where (C.2) follows based on the induction hypothesis, since $\mu_v^*$ leads to a direct path to a terminal node, and is not an upstream node of $v$. Thus, $\mu_v(n(k+1)) = \mu_v^*$ for all $v \in \cup_{m=0}^{k+1} \mathcal{R}_m$. This implies that the best reply process, when we cycle through the nodes in a round robin, converges within at most $n^2$ iterations. ■

## C.4 Relation to the Discounted-Reward Traveling Salesman Problem

In this section, we show the relationship between the Min-Exp-Cost-Path-NT problem of Section 5.2.1 and the Discounted-Reward-TSP, a path planning problem studied in the theoretical computer science community [68]. Note that this section is merely pointing out the relationship between the objectives/constraints of the two problems, and is not claiming that one is reducible to the other. In Discounted-Reward-TSP, each node $v$ has a prize $\pi_v$ associated with it and each edge $(u, v)$ has a cost $l_{uv}$ associated with it. The goal is to find a path $\mathcal{P}$ that visits all nodes and that maximizes the discounted reward collected $\sum_v \xi^{l_v^{\mathcal{P}}} \pi_v$, where $\xi < 1$ is the discount factor, and $l_v^{\mathcal{P}} = \sum_{e \in \mathcal{E}(\mathcal{P}_v)} l_e$ is the cost incurred along path $\mathcal{P}$ until node $v$.

In the setting of our Min-Exp-Cost-Path-NT problem, the prize of a node $v$ is taken as $\pi_v = \log_\xi(1 - p_v)$ for a value of $\xi < 1$. Our Min-Exp-Cost-Path-NT objective can then be reformulated as $\sum_{e \in \mathcal{E}(\mathcal{P})} \xi^{\pi_e^{\mathcal{P}}} l_e$, where $\pi_e^{\mathcal{P}} = \sum_{v \in \mathcal{V}(\mathcal{P}_e)} \pi_v$ is the reward collected along path $\mathcal{P}$ until edge $e$ is encountered. We can refer to this problem as the Discounted-Cost-TSP problem, drawing a parallel to the Discounted-Reward-TSP problem described above. However, note that our problem is not the same as the Discounted-Reward-TSP problem. Rather, we simply illustrated a relationship between the two problems, which can lead to further explorations in this area.

## C.5 Formulation as Stochastic Shortest Path Problem with Recourse

In this section, we show that we can formulate the Min-Exp-Cost-Path problem as a special case of the stochastic shortest path problem with recourse [69]. The terminology of stochastic shortest path here is different from its usage in 5.1.2. The stochastic shortest path problem with recourse consists of a graph where the edge weights are random variables taking values from a finite range. As the graph is traversed, the realizations of the cost of an edge is learned when one of its end nodes are visited. The goal is to find a policy that minimizes the expected cost from a source node $v_s$ to a destination node $v_t$. The best policy would determine where to go next based on the currently available information.

Consider the Min-Exp-Cost-Path problem on a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with probability of success $p_v \in [0, 1]$, for all $v \in \mathcal{V}$. We can formulate this as a special case of this stochastic shortest path problem with recourse, by adding a node $v_t$ which acts as the destination node. Each node in $\mathcal{G}$ is connected to $v_t$ with a edge of random weight. The edge from $v$ to $v_t$ has weight $l_{vv_t} = \begin{cases} 0, & \text{w.p. } p_v \\ \infty, & \text{w.p. } 1 - p_v \end{cases}$. The remaining set of edges $\mathcal{E}$ are deterministic. The solution to the shortest path problem from $v_s$ to $v_t$ with recourse, would provide a policy that would give us the solution to the Min-Exp-Cost-Path problem. The policy in this special case would produce a path from $v_s$ to a node in the set of terminal nodes $T$. However, the general stochastic shortest path with recourse is a much harder problem to solve than the Min-Exp-Cost-Path problem and the heuristics utilized for stochastic shortest path with recourse are not particularly suited to our specific problem. For instance, in the open loop feedback certainty equivalent heuristic [113], at each iteration, the uncertain edge costs are replaced with their expectation and the next node is chosen according to the deterministic shortest path to the destination. In our setting this would correspond to the heuristic of moving along the deterministic shortest path to the closest terminal node. Such a heuristic would ignore the probability of success $p_v$ of the nodes.

# Bibliography


[1] A. Goldsmith, *Wireless communications.* Cambridge university press, 2005.

[2] A. Ghaffarkhah and Y. Mostofi, *Channel learning and communication-aware motion planning in mobile networks*, in *American Control Conference (ACC),*, pp. 5413–5420, IEEE, 2010.

[3] M. Malmirchegini and Y. Mostofi, *On the spatial predictability of communication channels*, *IEEE Transactions on Wireless Communications,* **11** (2012), no. 3 964–978.

[4] M. Malmirchegini and Y. Mostofi, *An integrated sparsity and model-based probabilistic framework for estimating the spatial variations of communication channels*, *Physical Communication* **5** (2012), no. 2 102–118.

[5] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, *Coverage control for mobile sensing networks*, in *IEEE International Conference on Robotics and Automation (ICRA)*, vol. 2, pp. 1327–1332, 2002.

[6] P. Tokekar, J. Vander Hook, D. Mulla, and V. Isler, *Sensor planning for a symbiotic uav and ugv system for precision agriculture*, *IEEE Transactions on Robotics* **32** (2016), no. 6 1498–1511.

[7] D. W. Casbeer, D. B. Kingston, R. W. Beard, and T. W. McLain, *Cooperative forest fire surveillance using a team of small unmanned air vehicles*, *International Journal of Systems Science* **37** (2006), no. 6 351–360.

[8] B. Grocholsky, J. Keller, R. V. Kumar, and G. J. Pappas, *Cooperative air and ground surveillance*, .

[9] K. Zhou, S. I. Roumeliotis, *et. al.*, *Multirobot active target tracking with combinations of relative observations*, *IEEE Transactions on Robotics* **27** (2011), no. 4 678–695.

[10] N. E. Leonard, D. A. Paley, F. Lekien, R. Sepulchre, D. M. Fratantoni, and R. E. Davis, *Collective motion, sensor networks, and ocean sampling*, *Proceedings of the IEEE* **95** (2007), no. 1 48–74.

[11] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, *Wireless communication using unmanned aerial vehicles (uavs): Optimal transport theory for hover time optimization*, *IEEE Transactions on Wireless Communications* **16** (2017), no. 12 8052–8066.

[12] M. Alzenad, A. El-Keyi, and H. Yanikomeroglu, *3d placement of an unmanned aerial vehicle base station for maximum coverage of users with different qos requirements*, *IEEE Wireless Communications Letters* (2017).

[13] J. Lyu, Y. Zeng, R. Zhang, and T. J. Lim, *Placement optimization of uav-mounted mobile base stations*, *IEEE Communications Letters* **21** (2017), no. 3 604–607.

[14] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, *Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage.*, *IEEE Communications Letters* **20** (2016), no. 8 1647–1650.

[15] Y. Kantaros and M. M. Zavlanos, *Distributed communication-aware coverage control by mobile sensor networks*, *Automatica* **63** (2016) 209–220.

[16] A. Ghaffarkhah and Y. Mostofi, *Dynamic networked coverage of time-varying environments in the presence of fading communication channels*, *ACM Transactions on Sensor Networks (TOSN)* **10** (2014), no. 3 45.

[17] K. Daniel, S. Rohde, N. Goddemeier, and C. Wietfeld, *A communication aware steering strategy avoiding self-separation of flying robot swarms*, in *Intelligent Systems (IS), IEEE International Conference*, pp. 254–259, 2010.

[18] G. A. Hollinger and S. Singh, *Multirobot coordination with periodic connectivity: Theory and experiments*, *IEEE Transactions on Robotics* **28** (2012), no. 4 967–973.

[19] A. Ghaffarkhah and Y. Mostofi, *Path planning for networked robotic surveillance*, *IEEE Transactions on Signal Processing* **60** (July, 2012) 3560–3575.

[20] M. Stachura and E. W. Frew, *Cooperative target localization with a communication-aware unmanned aircraft system*, *AIAA Journal of Guidance, Control, and Dynamics* **34** (2011), no. 5 1352–1362.

[21] M. M. Zavlanos, M. B. Egerstedt, and G. J. Pappas, *Graph-theoretic connectivity control of mobile robot networks*, *Proceedings of the IEEE* **99** (2011), no. 9 1525–1540.

[22] M. M. Zavlanos and G. J. Pappas, *Distributed connectivity control of mobile networks*, *IEEE Transactions on Robotics* **24** (2008), no. 6 1416–1428.

[23] M. Stachura and E. Frew, *Communication-aware information-gathering experiments with an unmanned aircraft system*, *Journal of Field Robotics* **34** (2017), no. 4 736–756.

[24] S. Kemna, D. A. Caron, and G. S. Sukhatme, *Adaptive informative sampling with autonomous underwater vehicles: Acoustic versus surface communications*, in *MTS/IEEE OCEANS*, pp. 1–8, 2016.

[25] A. Ghaffarkhah and Y. Mostofi, *Communication-aware motion planning in mobile networks*, *IEEE Transactions on Automatic Control* **56** (2011), no. 10 2478–2485.

[26] S. Ponda, J. Redding, H.-L. Choi, J. P. How, M. Vavrina, and J. Vian, *Decentralized planning for complex missions with dynamic communication constraints*, in *American Control Conference (ACC), 2010*, pp. 3998–4003, IEEE, 2010.

[27] S. S. Ponda, L. B. Johnson, A. N. Kopeikin, H.-L. Choi, and J. P. How, *Distributed planning strategies to ensure network connectivity for dynamic heterogeneous teams*, *IEEE Journal on Selected Areas in Communications* **30** (2012), no. 5 861–869.

[28] Y. Yan and Y. Mostofi, *Co-optimization of communication and motion planning of a robotic operation under resource constraints and in fading environments*, *IEEE Transactions on Wireless Communications,* **12** (2013), no. 4 1562–1572.

[29] U. Ali, H. Cai, Y. Mostofi, and Y. Wardi, *Motion-communication co-optimization with cooperative load transfer in mobile robotics: an optimal control perspective*, *IEEE Transactions on Control of Network Systems* (2018).

[30] M. Lindhé and K. Johansson, *Using robot mobility to exploit multipath fading*, *IEEE Wireless Communications,* **16** (2009), no. 1 30–37.

[31] Y. Zeng and R. Zhang, *Energy-efficient uav communication with trajectory optimization*, *IEEE Transactions on Wireless Communications* **16** (2017), no. 6 3747–3760.

[32] F. Jiang and A. L. Swindlehurst, *Optimization of uav heading for the ground-to-air uplink*, *IEEE Journal on Selected Areas in Communications* **30** (2012), no. 5 993–1005.

[33] C. Dixon and E. Frew, *Optimizing cascaded chains of unmanned aircraft acting as communication relays*, *IEEE Journal on Selected Areas in Communications* **30** (2012), no. 5 883–898.

[34] Y. Yan and Y. Mostofi, *Robotic router formation in realistic communication environments*, *IEEE Transactions on Robotics,* **28** (2012), no. 4 810–827.

[35] N. Chatzipanagiotis and M. M. Zavlanos, *Distributed scheduling of network connectivity using mobile access point robots*, *IEEE Transactions on Robotics* **32** (2016), no. 6 1333–1346.

[36] Y. Zeng, R. Zhang, and T. J. Lim, *Throughput maximization for uav-enabled mobile relaying systems*, *IEEE Transactions on Communications* **64** (2016), no. 12 4983–4996.

[37] J. Fink, A. Ribeiro, and V. Kumar, *Robust control of mobility and communications in autonomous robot teams*, *IEEE Access* **1** (2013) 290–309.

[38] N. Chatzipanagiotis, Y. Liu, A. Petropulu, and M. M. Zavlanos, *Distributed cooperative beamforming in multi-source multi-destination clustered systems*, *IEEE Transactions on Signal Processing* **62** (2014), no. 23 6105–6117.

[39] N. Chatzipanagiotis, Y. Liu, A. Petropulu, and M. Zavlanos, *Controlling groups of mobile beamformers*, in *IEEE Conference on Decision and Control (CDC)*, pp. 1984–1989, 2012.

[40] D. S. Kalogerias and A. P. Petropulu, *Mobile beamforming & spatially controlled relay communications*, in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6405–6409, 2016.

[41] Y. Yan and Y. Mostofi, *Efficient clustering and path planning strategies for robotic data collection using space-filling curves*, *IEEE Transactions on Control of Network Systems* (2016).

[42] O. Tekdas, V. Isler, J. H. Lim, and A. Terzis, *Using mobile robots to harvest data from sensor fields*, *IEEE Wireless Communications* **16** (2009), no. 1 22–28.

[43] G. Hollinger, S. Choudhary, P. Qarabaqi, C. Murphy, U. Mitra, G. Sukhatme, M. Stojanovic, H. Singh, and F. Hover, *Underwater data collection using robotic sensor networks*, *IEEE Journal on Selected Areas in Communications* **30** (2012), no. 5 899–911.

[44] Y. Yan and Y. Mostofi, *To go or not to go: On energy-aware and communication-aware robotic operation*, *IEEE Transactions on Control of Network Systems,* **1** (2014), no. 3 218–231.

[45] A. J. Carfang, E. W. Frew, and D. B. Kingston, *Cascaded optimization of aircraft trajectories for persistent data ferrying*, *AIAA Journal of Aerospace Information Systems* **11** (2014), no. 12 807–820.

[46] R. Mudumbai, D. Brown III, U. Madhow, and H. Poor, *Distributed transmit beamforming: challenges and recent progress*, *IEEE Communications Magazine,* **47** (2009), no. 2 102–110.

[47] P. Ochiai, H.and Mitran, H. Poor, and V. Tarokh, *Collaborative beamforming for distributed wireless ad hoc sensor networks*, *IEEE Transactions on Signal Processing,* **53** (2005), no. 11 4110–4124.

[48] J. Hou, Z. Lin, W. Xu, and G. Yan, *Distributed transmit beamforming with autonomous and self-organizing mobile antennas*, in *IEEE Global Telecommunications Conference*, pp. 1–5, 2010.

[49] A. Muralidharan and Y. Mostofi, *Path planning for a connectivity seeking robot*, in *IEEE Globecom Workshop on Wireless Networking for Unmanned Autonomous Vehicles*, pp. 1–6, 2017.

[50] S. Caccamo, R. Parasuraman, L. Freda, M. Gianni, and P. Ogren, *Rcamp: Resilient communication-aware motion planner and autonomous repair of wireless connectivity in mobile robots*, in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2153–0866, 2017.

[51] K. Molaverdikhani and M. Tabeshian, *Mapping the probability of microlensing detection of extra-solar planets*, *arXiv preprint arXiv:0911.4424* (2009).

[52] S. Rosenthal, M. Veloso, and A. Dey, *Is someone in this office available to help me?*, *Journal of Intelligent & Robotic Systems* **66** (2012), no. 1 205–221.

[53] R. Dai, J. Maximoff, and M. Mesbahi, *Optimal trajectory generation for establishing connectivity in proximity networks*, *IEEE Transactions on Aerospace and Electronic Systems* **49** (2013), no. 3 1968–1981.

[54] Y. Kantaros and M. M. Zavlanos, *Distributed intermittent connectivity control of mobile robot networks*, *IEEE Transactions on Automatic Control* **62** (2017), no. 7 3109–3121.

[55] C. Ooi and C. Schindelhauer, *Minimal energy path planning for wireless robots*, *Mobile Networks and Applications* **14** (2009), no. 3 309–321.

[56] U. Ali, H. Cai, Y. Mostofi, and Y. Wardi, *Motion and communication co-optimization with path planning and online channel prediction*, in *American Control Conference (ACC)*, pp. 7079–7084, 2016.

[57] A. Muralidharan and Y. Mostofi, *Distributed beamforming using mobile robots*, in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6385–6389, 2016.

[58] S. Karaman and E. Frazzoli, *Incremental sampling-based algorithms for optimal motion planning*, *Robotics Science and Systems VI* **104** (2010).

[59] M. Likhachev, D. Ferguson, G. Gordon, A. Stentz, and S. Thrun, *Anytime search in dynamic graphs*, *Artificial Intelligence* **172** (2008), no. 14 1613–1643.

[60] P. Jaillet, *Probabilistic traveling salesman problems.* PhD thesis, Massachusetts Institute of Technology, 1985.

[61] D. J. Bertsimas, *A vehicle routing problem with stochastic demand*, *Operations Research* **40** (1992), no. 3 574–585.

[62] T. Chung and J. Burdick, *Analysis of search decision making using probabilistic search strategies*, *IEEE Transactions on Robotics* **28** (2012), no. 1 132–144.

[63] G. Hollinger, S. Singh, J. Djugash, and A. Kehagias, *Efficient multi-robot search for a moving target*, *The International Journal of Robotics Research* **28** (2009), no. 2 201–219.

[64] T. Chung, G. Hollinger, and V. Isler, *Search and pursuit-evasion in mobile robotics*, *Autonomous robots* **31** (2011), no. 4 299.

[65] F. Bourgault, T. Furukawa, and H. Durrant-Whyte, *Optimal search for a lost target in a bayesian world*, in *Field and service robotics*, pp. 209–222, Springer, 2003.

[66] A. Blum, P. Chalasani, D. Coppersmith, B. Pulleyblank, P. Raghavan, and M. Sudan, *The minimum latency problem*, in *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pp. 163–171, ACM, 1994.

[67] H. A. Simon and J. Kadane, *Optimal problem-solving search: All-or-none solutions*, *Artificial Intelligence* **6** (1975), no. 3 235–247.

[68] A. Blum, S. Chawla, D. Karger, T. Lane, A. Meyerson, and M. Minkoff, *Approximation algorithms for orienteering and discounted-reward tsp*, *SIAM Journal on Computing* **37** (2007), no. 2 653–670.

[69] G. Polychronopoulos and J. Tsitsiklis, *Stochastic shortest path problems with recourse*, .

[70] T. S. Rappaport, *Wireless communications: principles and practice*, vol. 2. prentice hall PTR New Jersey, 1996.

[71] H. Hashemi, *A study of temporal and spatial variations of the indoor radio propagation channel*, in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 127–134, IEEE, 1994.

[72] A. Muralidharan and Y. Mostofi, *Energy optimal distributed beamforming using unmanned vehicles*, *IEEE Transactions on Control of Network Systems* (2017).

[73] W. Smith and D. Cox, *Urban propagation modeling for wireless systems*, tech. rep., DTIC Document, 2004.

[74] D. Goldenberg, J. Lin, A. Morse, B. Rosen, and Y. Yang, *Towards mobility as a network control primitive*, in *ACM international symposium on Mobile ad hoc networking and computing*, pp. 163–174, 2004.

[75] C. Hsu, *Design and analysis of capacity-achieving codes and optimal receivers with low complexity.* PhD thesis, University of Michigan, 2006.

[76] D. Pisinger, *A minimal algorithm for the multiple-choice knapsack problem*, *European Journal of Operational Research* **83** (1995), no. 2 394–410.

[77] A. Abu-Dayya and N. C. Beaulieu, *Outage probabilities in the presence of correlated lognormal interferers*, *IEEE Transactions on Vehicular Technology,* **43** (1994), no. 1 164–173.

[78] J. Esary, F. Proschan, and D. Walkup, *Association of random variables, with applications*, *The Annals of Mathematical Statistics* **38** (1967), no. 5 1466–1474.

[79] K. Du and M. Swamy, *Wireless communication systems: from RF subsystems to 4G enabling technologies.* Cambridge University Press, 2010.

[80] S. Lönn, U. Forssen, P. Vecchia, A. Ahlbom, and M. Feychting, *Output power levels from mobile phones in different geographical areas; implications for exposure assessment*, *Occupational and Environmental Medicine* **61** (2004), no. 9 769–772.

[81] A. Muralidharan and Y. Mostofi, *First passage distance to connectivity for mobile robots*, in *American Control Conference (ACC)*, pp. 1517–1523, 2017.

[82] A. Muralidharan and Y. Mostofi, *Statistics of the distance traveled until connectivity for unmanned vehicles*, *Autonomous Robots, special issue on Robot Communication Challenges (submitted)*.

[83] A. J. Siegert, *On the first passage time probability problem*, *Physical Review* **81** (1951), no. 4 617.

[84] C. Gardiner, *Stochastic methods.* Springer Berlin, 2009.

[85] L. M. Ricciardi and S. Sato, *First-passage-time density and moments of the ornstein-uhlenbeck process*, *Journal of Applied Probability* (1988) 43–57.

[86] L. M. Ricciardi and L. Sacerdote, *The ornstein-uhlenbeck process as a model for neuronal activity*, *Biological cybernetics* **35** (1979), no. 1 1–9.

[87] B. Leblanc and O. Scaillet, *Path dependent options on yields in the affine term structure model*, *Finance and Stochastics* **2** (1998), no. 4 349–367.

[88] E. Di Nardo, A. Nobile, E. Pirozzi, and L. Ricciardi, *A computational approach to first-passage-time problems for gauss–markov processes*, *Advances in Applied Probability* **33** (2001), no. 2 453–482.

[89] R. M. Dudley, *Real analysis and probability*, vol. 74. Cambridge University Press, 2002.

[90] A. Papoulis and S. U. Pillai, *Probability, random variables, and stochastic processes*. Tata McGraw-Hill Education, 2002.

[91] C. Mehr and J. McFadden, *Certain properties of gaussian processes and their first-passage times*, *Journal of the Royal Statistical Society. Series B (Methodological)* (1965) 505–522.

[92] J. L. Doob, *Heuristic approach to the kolmogorov-smirnov theorems*, *The Annals of Mathematical Statistics* **20** (1949), no. 3 393–403.

[93] S. M. Kay, *Fundamentals of statistical signal processing, volume i: Estimation theory (v. 1)*, *PTR Prentice-Hall, Englewood Cliffs* (1993).

[94] M. Kline, *Calculus: an intuitive and physical approach*. Courier Corporation, 1998.

[95] D. Eberly, *Moving along a curve with specified speed*, *Preprint, see http://www. geometrictools. com* (2008) 2.

[96] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 2012.

[97] C. P. Robert, *Intrinsic losses*, *Theory and decision* **40** (1996), no. 2 191–214.

[98] H. O. Lancaster and E. Seneta, *Chi-square distribution*, *Encyclopedia of biostatistics* **2** (2005).

[99] P. Craig, *A new reconstruction of multivariate normal orthant probabilities*, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **70** (2008), no. 1 227–243.

[100] D. A. Darling and A. Siegert, *The first passage problem for a continuous markov process*, *The Annals of Mathematical Statistics* (1953) 624–639.

[101] A. Muralidharan and Y. Mostofi, *Path planning for minimizing the expected cost until success*, *IEEE Transactions on Robotics (under review)*.

[102] D. Bertsekas, *Dynamic programming and optimal control*, vol. 1. Athena Scientific Belmont, MA, 1995.

[103] M. Garey and D. Johnson, *Computers and intractability*, vol. 29. W. H. Freeman, New York, 2002.

[104] J. Garcia-Lunes-Aceves, *Loop-free routing using diffusing computations*, *IEEE/ACM Transactions on Networking (TON)* **1** (1993), no. 1 130–141.

[105] D. Monderer and L. Shapley, *Potential games*, *Games and economic behavior* **14** (1996), no. 1 124–143.

[106] J. R. Marden and J. S. Shamma, *Revisiting log-linear learning: Asynchrony, completeness and payoff-based implementation*, *Games and Economic Behavior* **75** (2012), no. 2 788–808.

[107] D. Fudenberg and J. Tirole, *Game theory, 1991*, *Cambridge, Massachusetts* **393** (1991) 12.

[108] T. Smith and R. Simmons, *Heuristic search value iteration for pomdps*, in *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pp. 520–527, AUAI Press, 2004.

[109] A. Barto, S. Bradtke, and S. Singh, *Learning to act using real-time dynamic programming*, *Artificial intelligence* **72** (1995), no. 1-2 81–138.

[110] S. Kirkpatrick, C. Gelatt, and M. Vecchi, *Optimization by simulated annealing*, *science* **220** (1983), no. 4598 671–680.

[111] R. Narasimhan and D. Cox, *A handoff algorithm for wireless systems using pattern recognition*, in *Personal, Indoor and Mobile Radio Communications, 1998. The Ninth IEEE International Symposium on*, vol. 1, pp. 335–339, IEEE, 1998.

[112] R. Narasimhan and D. Cox, *Estimation of mobile speed and average received power with application to corner detection and handoff*, in *in 33rd Asilomar Conf. on Signals, Systems & Computers*, Citeseer, 1999.

[113] S. Gao and I. Chabini, *Optimal routing policy problems in stochastic time-dependent networks*, *Transportation Research Part B: Methodological* **40** (2006), no. 2 93–122.