University of California
Santa Barbara

Path Planning Algorithms for Robotic Agents

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

Doctor of Philosophy
in
Mechanical Engineering

by

Pushkarini Agharkar

Committee in charge:

Professor Francesco Bullo, Chair
Professor Bassam Bamieh
Professor Brad Paden
Professor Yasamin Mostofi
Professor João Hespanha

March 2016

The Dissertation of Pushkarini Agharkar is approved.

Professor Bassam Bamieh

Professor Brad Paden

Professor Yasamin Mostofi

Professor João Hespanha

Professor Francesco Bullo, Committee Chair

June 2015

Path Planning Algorithms

for Robotic Agents

by

Pushkarini Agharkar

To my father,

Dr. Hemchandra Agharkar (M.D, D.P.M)

Acknowledgements

First and foremost, I would like to thank my advisor Professor Francesco Bullo for advising me in my research at UCSB. I thank him for being a great role model, for his immense patience and for teaching me the best research practices. I would also like to thank him for encouraging the wonderful environment of the motion lab. I would also like to thank Dr. Shaunak Bopardikar for mentoring me through the thought process which resulted in what is the second chapter of this thesis. I would also like to thank Rushabh Patel for a collaboration which led to results in the third chapter of this thesis, as well as my other collaborators Dr. Sara Susca and Prof. Sonia Martinez. I thank my dissertation committee for providing their time and advice.

I would like to thank Prof. Niket Kaisare and Prof. R. I. Sujith from IIT Madras. Its due to their encouragement and guidance that I deciced to pursue these four years of research. I would like to thank friends from the motion lab especially Anahita, Fabio, Vaibhav, Jeff, Wenjun and Sepehr for their gestures big and small. I would like to thank Prof. Scott Marcus for his inspiring attitude, his sitar lessons and also our sitar ensemble at UCSB.

I am grateful to have supportive roommates like Stephanie, Christine and Amber throughout these years. I thank them for introducing me to this country. I am thankful to Mani, Bonnie, Raga and Vikram for wonderful vacations and time-outs. I am terribly lucky to have friends like Santosh, Dinesh and Sumit in Santa Barbara. The time spent with them and our bond has kept me afloat and given me so much to look forward to. I would also like to thank Sushant for helping me transition out of graduate school.

I thank Ramani for growing up when I wasn't looking and for advising and encouraging me in work and other matters. I thank my mother for passing on the joie de vivre.

Curriculum Vitæ

Pushkarini Agharkar

Education

2011	B.Tech. in Aerospace Engineering, Indian Institute of Technology Madras.
2011	M.Tech. in Aerospace Engineering, Indian Institute of Technology Madras.

Work Experience

Jul-Sept 2014	Summer Intern, Bosch LLC, Palo Alto, California
May-Jul 2010	Research Intern, École Polytechnique, Paris, France
May-Jul 2009	Summer Intern, Larsen & Toubro, Mumbai, India

Selected Publications

(J1) Vehicle Routing Algorithms for Radially Escaping Targets. P. Agharkar, S. D. Bopardikar, F. Bullo, *SIAM Journal on Control and Optimization, Note: submitted.*

(J2) Quickest Detection over Robotic Roadmaps. P. Agharkar, F. Bullo, *IEEE Transactions on Robotics, Note: submitted.*

(J3) Robotic Surveillance and Markov Chains with Minimal First Passage Time. R. Patel, P. Agharkar, F. Bullo, *IEEE Transactions on Automatic Control, Note: accepted.*

(J4) Synchronization of Beads on a Ring by Feedback Control. S. Susca, P. Agharkar, S. Martinez, F. Bullo, *SIAM Journal on Control and Optimization, 52(2): 914-938, 2014.*

(C1) Robotic Surveillance and Markov Chains with Minimal First Passage Time. P. Agharkar, P. Patel, F. Bullo, *IEEE Conf. on Decision and Control, Los Angeles, CA, USA, pages 6603-6608, Dec. 2014.*

(C2) Vehicle Routing Algorithms to Intercept Escaping Targets. P. Agharkar, F. Bullo, *American Control Conference, Portland, OR, USA, pages 952-957, June 2014*

Abstract

Path Planning Algorithms

for Robotic Agents

by

Pushkarini Agharkar

The focus of this work is path planning algorithms for autonomous agents. Specifically, we study problems in three areas where path planning to direct the motion of autonomous agents is critical for their performance. The first problem is a vehicle routing problem in which mobile demands appear in an environment and the task of the autonomous agent is to stop the demands from escaping the environment boundary. We first propose two fundamental performance bounds for the proposed problem. We then propose routing algorithms for this problem with performance guarantees. We examine the gap between these guarantees and the fundamental performance bounds. The second problem is a surveillance problem in a networked environment. The tasks of the autonomous surveillance agent in this problem are to (1) detect unknown intruder locations and (2) detect anomalies based on noisy measurements. We propose Markov chain based routing algorithms for the surveillance agent to achieve these goals. We parameterize these routing algorithms using a property of Markov chains called the mean first passage time. We also frame optimization problems to obtain optimal algorithms for the two surveillance tasks. The third problem studied in this work is a boundary guarding problem in which the task of a set of patrolling agents constrained to move on a ring is to achieve synchronization using only local communication. We propose a coordination algorithm to solve this problem and identify initial agent configurations under which synchronization is guaranteed.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Autonomous robotic agents have numerous applications, for instance in hospital and office delivery system [89],[56], as museum tour guides [23], for topological mapping [99], in warehouse management [106] and building maintenance and surveillance [68]. Apart from the onboard sensors and the dexterity of the robots, the performance of these robots depends on the *path planning algorithms*, that is, the algorithms which determine their motion in the environment under service. In order to optimize the performance of the robots, these algorithms have to deal with various service allocation problems as well as facilitate coordination amongst robots in a multi-robot system.

We focus our attention of the motion planning of autonomous robots in three specific areas- (1) vehicle routing problems, (2) surveillance for intruder and anomaly detection and (3) coordinated boundary guarding. We now review the current literature in these areas. We will also state the motivation for the problem setups considered in the next chapters.

1.1 Vehicle Routing Problems

The Vehicle Routing Problem (VRP) was first introduced by [31] and has received wide attention for a long time [62],[43]. Due to a recent surge of activity in the area of motion planning for autonomous robots, a lot of variants of the VRP have been addressed over the last decade. An extensive list of such problems can be found in [22].

The most well-known VRP is arguably the classical Traveling Salesman Problem (TSP). In the TSP, a traveling salesman has to conduct the shortest tour of a given number of cities, visiting each city exactly once. The TSP like most other VRPs is NP-hard. The TSP and its extensions to other VRP problems have been explored extensively [11, 100]. Of the numerous extensions of the VRP, these are the extensions particularly relevant to the specific problem that we study:

(1) *Dynamic Vehicle Routing (DVR)* [83] problems, in which the arrival process of the demands to be serviced is stochastic.

(2) *VRP with time-windows* [33],[100], in which demands have to be serviced within a time-window.

(3) *VRP with moving demands* [48] where the vehicle has to intercept demands moving with arbitrary velocities.

The problem introduced in Chapter 2 draws from all these extensions. We now review some problems and the algorithms proposed for them in these extension areas, with particular attention to the VRP with moving targets, and also justify our problem setup.
(Dynamic Vehicle Routing): In DVR problems, demands arrive according to a stochastic rather than a deterministic process. This setup is motivated by stochastic arrival of demands in different applications, e.g. demands for services [98, 10] or goods [42, 7]. In DVR problems at least a part of the input is unknown to the vehicle and the

vehicle has to modify its path based on real-time information of new demand locations. In contrast to static vehicle routing, these problems hence require routing policies instead of pre-planned routes. The dynamically changing routes which are a result of these routing policies can be computed and executed due to technological advances like the introduction of the GPS and GIS and the widespread use of mobile and smartphones.

A review of Dynamic Vehicle Routing Problems was conducted in [83]. An example DVR problem which is also arguably the most general model for vehicle routing problems that have both a dynamic and a stochastic component is the $m-$vehicle Dynamic Traveling Repairman Problem (m-DTRP), which was introduced by [84] and mainly studied by Bertsimas and van Ryzin in [11], [12] and [13]. In [80] authors introduce policies for the m-DTRP which are adaptive with respect to the environment parameters and also provably optimal in light and heavy demand load conditions.

(VRP with time-windows): In situations where the demands are active only for a limited time period, time-windows are introduced in the setup of the VRP. This modification can enhance customer satisfaction and is necessary in situations where demand generation degrades over time, e.g. in the case of sensors which are active for some time on receiving information before going into an energy-saving "sleep" mode. The VRP with time-windows (VRPTW) was reviewed in [63]. Significant progress has been made in this class of VRPs, see for example, [93, 100, 20, 34]. In [79], authors model the problem with time-windows in a dynamic environment. They also consider stochastic time windows within which targets are required to be serviced. Modified versions of VRP with time-windows have been studied in [15, 79].

(VRP with moving demands): Several researchers have worked on dynamic vehicle routing problems (VRPs) involving moving targets in the past. The approximation complexity of Moving-Target TSP was studied in [46], where it was shown that Moving-Target TSP with n targets cannot be approximated better than by a factor of $2^{O(\sqrt{n})}$

times optimal within polynomial time unless $P = NP$. The authors in the same work also showed that if targets have the same velocities, then there is a polynomial time approximation for the Moving-Target TSP. Authors in [48] give a $2 + \epsilon$ approximation algorithm for instances of the Moving-Target TSP in which $O(\frac{\log n}{\log \log n})$ of the n points are moving with arbitrary velocity. Authors in [16] study a variant of the Moving-Target VRP in which targets appear on a segment and move with the same velocity. They prove that a first come first serve policy minimizes the expected time to service a target when the target arrival rate is very high as well as when the target speed is close to the vehicle speed. Authors in [26] study a kinetic variant of the k-delivery TSP where all targets move with the same velocity and a robotic arm moving with a finite capacity must intercept them. They provide constant-factor approximation for the problem. Authors in [6] study a grasp and delivery problem motivated by robot navigation and propose a 2-factor approximation algorithm. In [15], the moving targets have to be serviced within a time-window and a policy based on repeated computation of longest paths through the available set of targets is proposed to this end.

Apart from the above broader areas, more recent results on the subject of routing problems involving the task of target interception consider more general models for target behavior [8, 66, 61]. In [8], the authors propose a partitioning strategy for a multiple vehicle multiple target problem in which the targets can apply an evading strategy in response to the actions of the service vehicle. We consider a vehicle routing problem in which demands appear according to a stochastic process. On appearing, they move in the radially outward direction so that all demands have different speeds, depending on where they originate in the environment. The time-windows within which they need to be serviced also depend on their point of origin. They do not, however, modify their direction to evade an approaching vehicle.

Motivation: The problem setup in Chapter 2 is significantly different from earlier

setups in the following ways: The moving targets have different velocities depending on their angular location, as opposed to having same velocities as assumed in many problem setups looked at in literature [16, 46, 90, 79]. They also have different deadlines depending on their radial location as opposed to having the same deadline or time window before which they should be serviced [15, 16]. They move along radial direction so as to escape the environment as quickly as they can. One application of this problem setup is in robotic patrolling where it is necessary to stop malicious agents from leaving a region so as to protect the surroundings.

1.2 Robotic Surveillance

The second area of application which we study is robotic surveillance. The surveillance problem has appeared in the literature in various manifestations. Theoretical analysis of the surveillance problem was conducted in [28] and a survey of various surveillance scenarios and the corresponding approaches was presented in [4]. Surveillance strategies that minimize the refresh time, i.e. time period between subsequent visits to regions have been proposed in [76],[91] and [92]. In [76], authors propose optimal algorithms which minimize the refresh time for chain and tree graphs and constant factor algorithm for cyclic graphs. Authors in [92] consider the problem of minimizing specific weighted sums of refresh times and design non-intersecting tours on graphs for this surveillance criterion. In [91], the authors design speed controllers on closed paths to minimize the refresh time for a given set of points of interest in the environment.

The surveillance policies proposed in [76],[91] and [92] are deterministic in nature. Stochastic surveillance strategies assume importance in scenarios where the intruders can move or hide to avoid detection and as a result, the movement of the surveillance vehicle is required to be non-deterministic. A main result of [94] also shows that deterministic

policies are ill-suited when designing strategies with arbitrary constraints on those visit frequencies. Several authors have used Markov chain based approaches to design stochastic strategies for various surveillance tasks. Authors in [94] use the Metropolis-Hastings algorithm to achieve specified frequency of visits to regions of the environment. In [44], authors design random walk strategies on hypergraphs and parametrically vary the local transition probabilities over time in order to achieve fast convergence to a desired visit frequency distribution. In [95], authors use the fastest mixing Markov chain for quickest detection of anomalies.

Motivated by practical applications, the surveillance problem has also been dealt with in other innovative ways. For example authors in [87] consider different intruder models and present routing strategies for surveillance in scenarios corresponding to them. In [67] wireless sensor networks are utilized for intruder detection in previously unknown environments. In [5], the authors explore strategies for surveillance using a multi-agent ground vehicle system which must maintain connectivity between agents. A non-cooperative game framework is utilized in [27] to determine an optimal strategy for intruder detection, and in [77] a similar framework is used to analyze intruder detection for ad-hoc mobile networks.

We consider two problems within the broad area of robotic surveillance, the first one is concerned with the detection of unknown intruder location and the second with the quickest detection of anomalies in networked environments. We propose Markov chain based routing strategies for the problems. Both the problems and the proposed robotic routing strategies are parameterized by the metric called the *mean first passage time* of Markov chains. We now present the literature on the mean first passage time.

For a random walk associated with a Markov chain, the mean first passage time, also known as the *Kemeny constant*, of the chain is the expected time taken by a random walker to travel from an arbitrary start node to a second randomly-selected node in a

network. The Kemeny constant of a Markov chain first appeared in [55] and has since been studied by several scientists, e.g., see [52, 58] and references therein. Bounds on the mean first passage time for an arbitrary Markov chain over various network topologies appear in [52, 64].

The mean first passage time is closely related to other well-known metrics for graphs and Markov chains. We discuss two such quantities in what follows. First, the *Kirchhoff index* [59], also known as the *effective graph resistance* [38], is a related metric quantifying the distance between pairs of vertices in an electric network. The relationship between electrical networks and random walks on graphs is explained elaborately in [37]. For an arbitrary graph, the Kirchoff index and the Kemeny constant can be calculated from the eigenvalues of the conductance matrix and the transition matrix, respectively. The relationship between these two quantities for regular graphs is established in [75]. Second, the *mixing rate* of an irreducible Markov chain is the rate at which an arbitrary distribution converges to the chain's stationary distribution [35]. It is well-know that the mixing rate is related to the second largest eigenvalue of the transition matrix of the Markov chain. The influential text [65] provides a detailed review of the mixing rate and of other notions of mixing. Recently, [58] refers to the Kemeny constant as the "expected time to mixing" and relates it to the mixing rate.

Motivation: There are several motivations for the problem setups considered in Chapter 3 and 4. First of all, the setups highlight the effectiveness of the notion of the mean first passage time which is relevant to surveillance tasks in which each region should be accessible from the other regions in the environment in minumum time. The setups also take into account travel times required by robotic agents to travel across the regions of a networked environment. Third, the mean first passage time, which we analyze in the process of proposing strategies for the two problems is of independent mathematical interest and has applications potentially outside the area of robotics, e.g. in determining

how quickly information propagates in an online network [9] or how quickly an epidemic spreads through a contact network [105]. Lastly, the problem setups are motivated by realistic surveillance scenarios, namely detection of an unknown intruder location in least amount of time (Chapter 3) and surveillance under extreme modelling uncertainty and measurement noise (Chapter 4).

1.3 Boundary Guarding and Coordination

The third area of application is the problem of guarding environment boundaries. We study a multi-agent boundary guarding problem in which the agents achieve synchronization in motion along the boundary of the environment. The synchronization problem can also be seen as a consensus problem in which the agents reach consensus on the environment partition in order to service the environment in a distributed manner.

Consensus algorithms have been extensively studied, beginning with the early work on averaging opinions and stochastic matrices in [32]. For the setting of non-degenerate stochastic matrices, [102] gives convergence conditions for consensus algorithms under mild connectivity assumptions. Recent references on average consensus, algebraic graph methods and symmetric stochastic matrices include [72, 54]. Recent surveys [40, 74, 85] discuss attractive properties of these algorithms such as convergence under delays and communication failures, and robustness to communication noise.

Synchronization in itself has been a widely studied problem and has been explored for multi-agent system coordination; e.g. see [104, 36, 73, 70, 82]. In [104] a generalized distributed network of nonlinear dynamic systems with access to global information is considered and synchronization in the network is shown to occur for strong enough coupling strengths. The authors in [36] and [73] present distributed algorithms using which synchronization is achieved in multi-agent systems using event triggered and self

triggered control respectively. The authors in [70] draw analogies between impulsive and diffusive synchronization in the weak coupling limit.

References on the problem of perimeter estimation and monitoring by mobile robots include [29, 107, 97, 101]. Patrolling problems have also been studied in [76, 2, 69]. More relevant to the problem setup in this chapter are the studies in [25, 57], which make use of the steady-state orbit for even number of synchronized agents described here and referred to as 'balanced' synchronization.

Motivation: A possible worldly motivation for the study of this class of algorithms is the surveillance of regions in a 2D space. Some examples of similar problems in literature include [25, 57]. In [25] pairs of agents have to be released at particular points, sequentially, and with the same speed. In contrast, in our algorithm the number of agents can be odd, the agents can be released at arbitrary positions and with arbitrary speeds. The distributed algorithm in [57] requires only that the agents move with a fixed speed. However, it can not be easily extended to a perimeter which is a closed curve unless the agents are assumed to have unique identifiers. Further, we stabilize a broader range of trajectories, namely 'unbalanced' synchronization.

Apart from the application in boundary-patrolling, the study of the n beads problem can find justification on more fundamental grounds. Namely, the investigation of under what conditions systems subject to impacts and controlled dynamics are robustly stable, and what techniques can be useful helpful in proving such stability properties. Both these aspects have motivated us to consider this synchronization problem.

1.4 Organization

The thesis is organized as follows. In Chapter 2, we introduce and analyze the Radially Escaping Targets problem and propose routing policies for autonomous agents for this

problem. In Chapter 3 and Chapter 4 we study two surveillance problems. We propose routing policies for surveillance agents in the two setups, employing the notion of the mean first passage times of Markov chains in both the cases. In Chapter 5 we study a boundary patrolling problem involving consensus between robotic agents using limited communication. Finally, in Chapter 6 we summarize the results of the previous chapters and also list some open problems.

Chapter 2

Radially Escaping Targets Problem

The various extensions of the vehicle routing problems were discussed in Section 1.1. In this chapter we propose a novel vehicle routing problem involving moving targets. In the setup of this problem, a single target maintains the same velocity throughout with the intention of escaping the environment as quickly as possible. One application of this problem setup is in robotic patrolling where it is necessary to stop malicious agents from leaving a region so as to protect the surroundings.

The problem setup in this chapter is significantly different from earlier setups in the following ways: The moving targets have different velocities depending on their angular location, as opposed to having same velocities as assumed in many problem setups looked at in literature [16, 46, 90, 79]. They also have different deadlines depending on their radial location as opposed to having the same deadline or time window before which they should be serviced [15, 16].

2.1 Contributions

The contributions of this chapter [1] can be summarized as follows. We introduce a novel dynamic vehicle routing problem termed the *Radially Escaping Targets (RET)* problem. The RET problem has three parameters: the target arrival rate λ, the target speed $v < 1$ and the environment radius D.

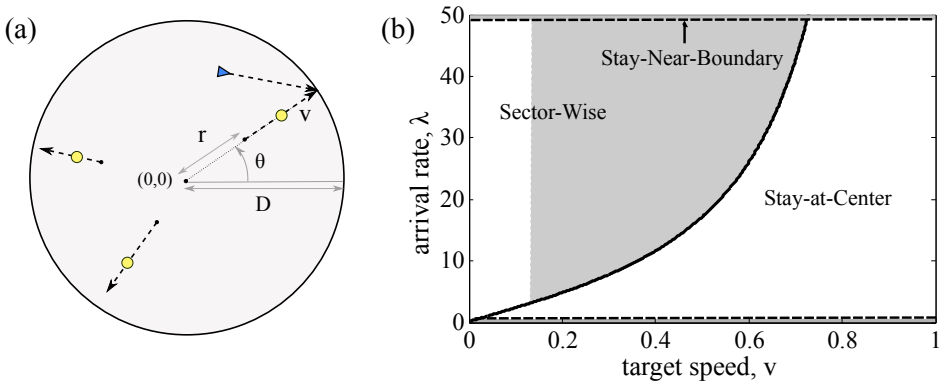

Figure 2.1: (a) Schematic of the Radially escaping targets (RET) problem. (b) The parameter regimes where the Stay-at-Center (SAC), Sector-Wise(SW) and Stay-Near-Boundary(SNB) policies are designed are shown for D=1. The gray shaded regions indicate the parameter regimes in which the policies are constant factor optimal.

We first determine two policy independent upper bounds on the fraction of targets that can be captured for the RET problem. In the process, we derive a novel method to establish upper and lower bounds on the path through radially escaping targets. Next, we formulate three policies: Stay-at-Center (SAC), Sector-wise (SW) and Stay-Near-Boundary (SNB) policy. The SAC policy is designed for low arrival rates while the SW policy is formulated for moderate arrival rates. The SNB policy is designed for high arrival rates. Lower bounds on the fraction of targets captured using the SAC, SW and SNB policies are obtained. In Table 2.1, we summarize these lower bounds and also present the factor of optimality (defined as the ratio of the fundamental upper bound for

[1] This work is a product of collaboration with Dr. Shaunak Bopardikar

the RET problem to the capture fraction of a policy). The symbol $\beta \approx 0.7120 \pm 0.0002$ and

$$\alpha(v) = \frac{\sqrt{v}}{\pi^2}\left(\left(\frac{v}{(1-v^2)^{3/2}}\right)^{1/2} + \frac{10}{3}\frac{(1-v^2)^{1/2}}{v}\right)^{-1/2}.$$

In Fig. 2.1(b), the design regimes for the SAC, SW and SNB policies are shown. The gray shaded regions indicate the regimes where the policies are constant factor optimal. The SAC and SNB policy are constant factor optimal in the asymptotic regimes of $\lambda \to 0^+$ and $\lambda \to +\infty$ respectively. The gray shaded regions separated by dashed lines are representative of these asymptotic regimes. For fixed target speed, the SW policy is within a constant factor of the optimal in the gray shaded region in the middle. We present numerical simulations which empirically verify our results.

Table 2.1: Performance of policies for the RET problem

Design Regime	Algorithm	Regime of constant factor optimality	Factor of optimality
Light load	Stay-At-Center	$\lambda \to 0^+$	1
Moderate load	Sector-wise	$\lambda > \frac{7\pi v}{(1-v^2)^{3/2}D}, v > \frac{1}{4\sqrt{2}}$	$\frac{1}{\alpha(v)}$
Fixed speed, heavy load	Stay-Near-Boundary	$\lambda \to +\infty$, $D > 1$	$\frac{7\beta}{2}$

The set-up of the RET problem can be viewed as a dynamical system where targets are generated via a stochastic process. The dynamical system needs to be controlled using a control law or policy in order to stop the targets from escaping the environment. The performance metric to evaluate the policy is the capture fraction of the targets which needs to be maximized. Fundamental upper bounds and achievable lower bounds on the capture fraction is the topic of the chapter. We study the gap between them as well.

2.2 Organization

This Chapter is organized as follows. In Section 2.1 we state the contributions of the work. In chapter 2.3 we describe the problem setup. The preliminary results are stated in Section 2.4 and the main results, i.e. the routing policies for the proposed problem are presented in 2.5. The chapter ends with numerical simulations presented in Section 2.6.

2.3 Problem Formulation

We start with introducing a DVR problem in which the environment is a disk of radius D given by:

$$\mathcal{E} = \{(r, \theta) : 0 \leq r \leq D \ \forall \theta \in [0, 2\pi)\}.$$

Targets appear independently and uniformly distributed in $\mathcal{E}$ with uniform spatial density. Their arrival times are modeled using a Poisson process with rate λ [86]. Uniform spatial distribution of the targets is realized through probability density functions $f(r) = 2r/D^2$ and $e(\theta) = 1/2\pi$ where r and θ are random variables describing the location of appearing targets in radial coordinates. Once the targets appear, they move radially outwards with a constant speed $v < 1$ and eventually reach the boundary of the environment. A vehicle with speed of 1 and confined to move in $\mathcal{E}$ intercepts the targets and captures them before they escape the environment. We refer to this problem as the Radially Escaping Targets (RET) problem for convenience and a schematic of the problem is shown in Fig. 2.1(a). The parameters of the RET problem are the target speed v, arrival rate λ and disk radius D.

Let $\mathcal{Q}(t) \subset \mathcal{E}$ denote the set of positions of all targets that have appeared but have not been serviced or have escaped before time t. Let $p(t) \in \mathcal{E}$ be the position of the vehicle at time t. A policy for the vehicle is a map $P : \mathcal{E} \times \text{FIN}(\mathcal{E}) \to \mathbb{R}^2$, where $\text{FIN}(\mathcal{E})$

is the set of finite subsets of $\mathcal{E}$, assigning a velocity to the service vehicle as a function of the current state of the system: $\dot{p}(t) = P(p(t), \mathcal{Q}(t))$. Let $m_{\text{cap}}(t)$ be the number of targets that have appeared and have been captured before time t and $m_{\text{miss}}(t)$ be the number of targets that have escaped and $m_{\text{tot}}(t) = m_{\text{miss}}(t) + m_{\text{cap}}(t)$, then the goal of this problem can be stated as follows:

Problem Statement Find policies P that maximize the fraction of targets that are serviced $\mathbb{F}_{\text{cap}}(P)$, termed as the *capture fraction*. Formally, for a policy P, we define the steady state average capture fraction as

$$\mathbb{F}_{\text{cap}}(P) := \limsup_{t \to +\infty} \mathbb{E}\left[\frac{m_{\text{cap}}(t)}{m_{\text{cap}}(t)+m_{\text{miss}}(t)}\right]$$

where the expectation is with respect to the stochastic process that generates the targets.

Each target has a deadline depending on when and where it appears in the environment. We propose policies for the service vehicle suitable for specific target speeds and arrival rates with provable guarantees on their performance. We first present some preliminary results which will be used to analyze policies for the RET problem.

2.4 Preliminary results

We start with reviewing some established results to intercept moving targets in shortest time as well as propose methods to obtain bounds on paths through a set of moving targets.

2.4.1 Time to capture a single target

The optimal strategy (i.e., taking minimum time) for a vehicle to capture a target moving at a speed less than that of the vehicle is to move in a straight line with maximum speed to intercept the target based on the constant bearing principle [53]. In the following definition, this result is stated in terms of radial coordinates.

Definition 1 (Constant bearing principle) *The time taken by the vehicle starting from* $p = (x, 0)$ *and moving with unit speed to capture a target located at* $q = (r, \theta)$ *and moving radially outward with constant speed* $v < 1$ *is*

$$T(p,q) = \frac{-v(x\cos\theta - r) + (v^2(x\cos\theta - r)^2 - (1-v^2)(2rx\cos\theta - x^2 - r^2))^{1/2}}{1-v^2}.$$

The next result gives a relation of the distance between the vehicle and target location to the time required to capture the moving target.

Lemma 2 (Time to capture) *The time* $T(p,q)$ *required by the vehicle starting from* $p = (x, 0)$ *and moving with unit speed to capture a target at* $q = (r, \theta)$ *moving radially outward with speed* v *satisfies the following inequality*

$$T(p,q) \leq \left(\frac{2v}{1-v^2} + \frac{1}{\sqrt{1-v^2}}\right) d(p,q),$$

where $d(p,q) = \sqrt{x^2 + r^2 - 2xr\cos\theta}$ *is the Euclidean distance between* p *and* q*. If* $r \leq x\cos\theta$*, then*

$$T(p,q) \leq \left(\frac{1}{\sqrt{1-v^2}}\right) d(p,q)$$

Proof: We start with providing an upper bound on the positive root y^+ of a quadratic equation. For the quadratic equation $ay^2 + by + c = 0$, if $a > 0$ and $c < 0$, then there are two possibilities: $b \geq 0$ or $b < 0$.

$$y^+ = \frac{-b+\sqrt{b^2-4ac}}{2a} = \begin{cases} \dfrac{-b+\sqrt{b^2+4a\,|c|}}{2a} \leq \dfrac{-b+b+2\sqrt{a\,|c|}}{2a} = \sqrt{\dfrac{|c|}{a}}, & b \geq 0, \\ \dfrac{-b+\sqrt{|b|^2+4a\,|c|}}{2a} \leq \dfrac{|b|}{a} + \sqrt{\dfrac{|c|}{a}}, & b < 0. \end{cases}$$

Since the time taken $T := T(p,q)$ to capture a target at q starting from p satisfies the following quadratic equation,

$$T^2(1-v^2) + 2vT(x\cos\theta - r) - (x^2 + r^2 - 2xr\cos\theta) = 0,$$

the result follows. ■

2.4.2 Optimal placement of vehicle

By optimal placement, we mean the location at which the vehicle should be placed in order for it to have the highest probability of capturing a target. To determine optimal placement, we start by defining the capturable set of a vehicle location.

Definition 3 (Capturable set) *A vehicle located at* $(x,0)$ *and moving with unit speed can only reach targets located in the capturable set*

$$C(x,v,D) := \{(r,\theta) \in \mathcal{E} : r < r_c \ \forall\theta \in [0,2\pi)\}$$

using the constant bearing principle, where

$$r_c(x,v,D,\theta) = \max\left(0, D - v\sqrt{D^2+x^2-2xD\cos\theta}\right).$$

These are the locations for which $r + vT \leq D$. The expression for r_c is obtained by

setting $r_c + vT = D$. The radial location r_c corresponds to the locations of targets that the vehicle can capture just before they escape the disk. The probability that a target is in the capturable set of a particular vehicle location $(x, 0)$ is given by

$$\rho(x, v, D) := \frac{\int_0^{2\pi}\int_0^D \mathbb{P}\left[(r,\theta) \in C(x, v, D)\right] f(r)e(\theta)drd\theta}{\int_0^{2\pi}\int_0^D \mathbb{P}\left[(r,\theta) \in \mathcal{E}\right] f(r)e(\theta)drd\theta} = \frac{\int_0^{2\pi}\int_0^{r_c} f(r)e(\theta)drd\theta}{\pi D^2}.$$

When the vehicle is at location $p^* = (x^*(v, D), 0)$ where

$$x^*(v, D) := \arg\max_{0\le x\le D} \rho(x, v, D), \tag{2.1}$$

the probability of it capturing a target is maximum. The vehicle location $x^*(v, D)$ is referred to as the *optimal location*. Let $\rho^*(v, D) := \rho(x^*, v, D)$. Closed form expressions for x^* and ρ^* do not appear to be possible for all $v \in (0, 1)$. However, from numerical calculations it is known that $x^* = 0$ for $v \in (0, 0.5]$ irrespective of the value of the parameter D. The numerically computed variation of $x^*(v, D)$ and $\rho^*(v, D)$ for $D = 1$ is shown in Fig. 2.2. For target speed $v \le 0.5$, $x^* = 0$ and the vehicle location $p^* = (0, 0)$ maximizes the probability of the vehicle being able to capture a target before it escapes. For higher speeds, this location is closer to the boundary. There is a qualitative difference between these two cases. For the former case, $p^* = (0, 0)$ is the unique vehicle location which maximizes ρ whereas for the later case, the set of corresponding optimal locations is all points with radial coordinate equal to x^*.

Theorem 4 (Capture fraction upper bound) *For every policy P for the* $\mathrm{RET}(v, \lambda)$ *problem,* $\mathbb{F}_{\mathrm{cap}}(P) \le \rho^*(v, D)$.

Proof: Let the vehicle start from x_1 and service target at p_1. The probability of the vehicle capturing this target is maximum when $x_1 = x^*$. The best case scenario is

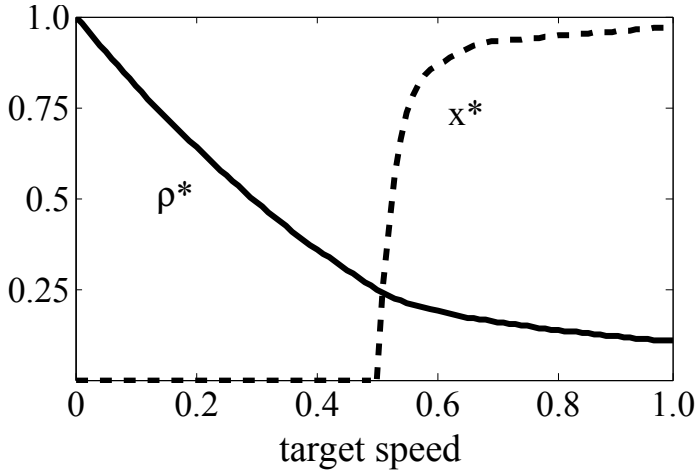

Figure 2.2: Optimal vehicle location x^* and the maximum probability ρ^* of capturing an escaping target starting from $(x^*, 0)$ as a function of target speed v for the RET problem with $D = 1$.

that no new target appears while the vehicle services it and repositions itself at x_2 so as to increase the probability of capturing a new target at p_2. This can be realized for a suitably low value of arrival rate λ. In order to maximize the probability of capturing the new target, $x_2 = x^*$ as well. Thus, to maximize the probability of capturing every new target, the vehicle returns to x^* and waits for a target to appear. With this strategy, the vehicle can still only capture targets which appear within $C(x^*, v, D)$. The fraction of targets which satisfy this criterion is $\rho^*(v, D)$. Thus, the vehicle can capture no more than $\rho^*(v, D)$ fraction of targets. ■

2.4.3 Quantification of targets inside the environment

In this subsection we quantify the number of targets in an unserviced region in the environment. We distinguish between targets *originating* and *accumulating* in a certain region. Targets are said to have accumulated in a region when after appearing, they spend time in the region, in the course of their trajectories.

Definition 5 (Annular section) *The annular section* $A(a, b, \theta_1, \theta_2) \subset \mathbb{R}^2$ *is the set*

$A(a, b, \theta_1, \theta_2) := \{(r, \theta) | a \leq r \leq b, \theta \in [\theta_1, \theta_2]\}$.

Lemma 6 (Accumulated targets in an annular section) *For $0 < a < b < D$, let n_A be the number of targets accumulated at steady state in an unserviced annular section $A(a, b, 0, 2\pi)$ and $f_a(x)$ be the distribution of the accumulating targets w.r.t the radial location $x \in [0, D]$. Then,*

(i) $\mathbb{E}[n_A] = (b^3 - a^3)\lambda/3vD^2$,

(ii) $\mathrm{Var}[n_A] = (b^3 - a^3)\lambda/3vD^2$ *and*

(iii) $f_a(x) = \lambda x^2/vD^2$.

Proof: Firstly, steady state is assumed, meaning that the initial transient has already passed, hence the time at which the snapshot is taken is $t \geq D/v$. Also, by unserviced, we mean that the vehicle has not serviced targets in the region under consideration for at least time D/v before the time instant under consideration. Let us examine the number of targets accumulating in the annulus $A_r := A(r, r + \Delta r, 0, 2\pi)$ due to targets appearing in the annulus $R_1 := A(p_1, p_1 + \Delta p_1, 0, 2\pi)$. Let us also assume that Δr and Δp are infinitesimal. The intensity of the Poisson arrival process on R_1 is directly proportional to its area and is equal to $2\pi p_1 \Delta p_1 \lambda / \pi D^2 = 2p_1 \Delta p_1 \lambda / D^2$.

$$
\begin{aligned}
&\mathbb{P}[A_r \text{ contains } n \text{ targets originating from } R_1] \\
&\qquad = \mathbb{P}\left[n \text{ targets originated from } R_1 \text{ in time interval } \left[t, t + \frac{\Delta r}{v}\right]\right] \\
&\qquad = \mathbb{P}\left[n \text{ targets originated from } R_1 \text{ in time interval of length} \frac{\Delta r}{v}\right] \\
&\qquad = \exp\left(-\frac{2p_1 \Delta p_1 \lambda \Delta r}{D^2 v}\right) \frac{\left(\frac{2p_1 \Delta p_1 \lambda \Delta r}{D^2 v}\right)^n}{n!},
\end{aligned}
$$

where $t = (r - p_1)/v$. Thus, the process of targets accumulating in A_r due to targets originating in R_1 is spatially Poisson with intensity $\text{area}(R_1)/(\pi D^2)\lambda/v = 2p_1\Delta p_1\lambda/D^2v$.

Next, let us examine the process of accumulation of targets in A_r due to two annuli $R_1 := A(p_1, p_1 + \Delta p_1, 0, 2\pi)$ and $R_2 := A(p_2, p_2 + \Delta p_2, 0, 2\pi)$.

$$\begin{aligned}\mathbb{P}[A_r \text{ contains } n \text{ targets from } R_1 \cup R_2] = &\sum_{i=0}^{n} \left[\exp\left(\frac{-2p_1\Delta p_1\lambda\Delta r}{D^2v}\right) \frac{\left(\frac{2p_1\Delta p_1\lambda\Delta r}{D^2v}\right)^i}{i!} \right. \\ &\left. \times \exp\left(\frac{-2p_2\Delta p_2\lambda\Delta r}{D^2v}\right) \frac{\left(\frac{2p_2\Delta p_2\lambda\Delta r}{D^2v}\right)^{(n-i)}}{(n-i)!} \right] \\ = &\exp\left(\frac{-2(p_1\Delta p_1 + p_2\Delta p_2)\lambda\Delta r}{D^2v}\right) \\ &\times \frac{\left(\frac{2(p_1\Delta p_1 + p_2\Delta p_2)\lambda\Delta r}{D^2v}\right)^n}{n!} \end{aligned} \tag{2.2}$$

Thus the process of targets accumulating in A_r due to targets originating in $R_1 \cup R_2$ is also spatially Poisson and the intensity of this process, given by $(2p_1\Delta p_1 + 2p_2\Delta p_2)\lambda/D^2v = (\text{area}(R_1) + \text{area}(R_2))/(\pi D^2)\lambda/v$, is the sum of the intensities due to R_1 and R_2. This can be extended to all the rings of radii $p \in [0, r]$. So arrival process of all targets accumulating in A_r is also spatially Poisson and has intensity $\text{area}(A(0, r, 0, 2\pi))/(\pi D^2)\lambda/v = (r^2\lambda/vD^2)$. Thus the expected number as well as the variance of targets accumulating in the unserviced annulus A_r is $r^2\lambda\Delta r/vD^2$.

Next, consider an annular section $A(a, b, 0, 2\pi)$. Since Poisson processes are additive, the arrival process of targets accumulating in $A(a, b, 0, 2\pi)$ is Poisson and is the sum of processes of targets accumulating in disjoint annuli like A_r with $r \in [a, b]$. Hence the

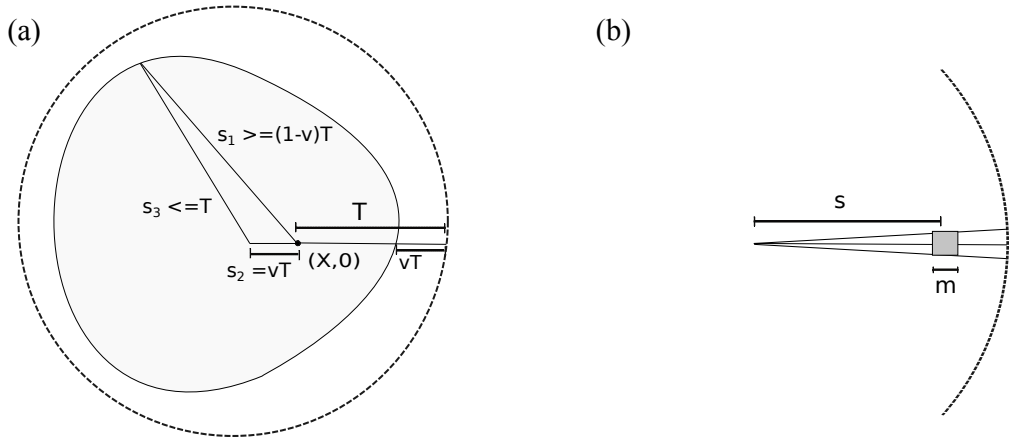

Figure 2.3: (a) The set S_T for the RET problem is shown by the gray shaded region. The dashed circle is the boundary of $\bar{S_T}$ which is a circle of radius T centered at $(X - vT, 0)$. (b) The area element ζ of length and width m in $\bar{S_T}$.

expected number and variance of targets accumulating in $A(a, b, 0, 2\pi)$ is

$$\int_a^b (r^2\lambda/vD^2)dr = (b^3 - a^3)\lambda/3vD^2.$$

Let $f_a(x)$ be the distribution of the number of accumulating targets w.r.t the radial location x. Since $\int_0^s f_a(x)dx = s^3\lambda/3vD^2$, we get $f_a(x) = \lambda x^2/vD^2$.

■

Lemma 7 (Travel time bound for RET problem) *Let targets arrive uniformly in $\mathcal{E}$ according to a Poisson arrival process of rate λ and move radially outward with speed v. Let Q be the set of targets accumulated in $\mathcal{E}$ at time t and T_d be the random variable giving the minimum amount of time required to travel to a target in Q starting from vehicle position $(X, 0)$. Then,*

$$\mathbb{E}[T_d] \geq \sqrt{\frac{\pi vD}{2\lambda}}.$$

Proof: To get a bound on the travel time, we start with defining a set S_T shown in Fig. 2.3(a), such that any target in it can be reached from the vehicle position $(X, 0)$ in

T time units or less. Mathematically,

$$S_T := \left\{(r,\theta) \in \mathcal{E} | X^2 + (r+vT)^2 - 2X(r+vT)\cos(\theta) \leq T^2\right\}.$$

Also, let $\bar{S_T} := \{(r,\theta) \in \mathcal{E} | (X - vT - r\cos\theta)^2 + (r\sin\theta)^2 \leq T^2\}$. Since the relative velocity of any target with respect to the vehicle is more than or equal to $(1-v)$, the distance s_1 of any point on the boundary of S_T from $(X,0)$ is greater than or equal to $T(1-v)$. Using the triangle inequality, the distance s_2 of that point from $(X - vT, 0)$ is less than or equal to T. Then, $S_T \subseteq \bar{S_T}$.

If T_d is the random variable giving the minimum amount of time to go from vehicle location $(X,0)$ to a target, then $T_d > T$ if S_T is empty and $\mathbb{P}[T_d > T] = \mathbb{P}[|S_T| = 0]$. Here, the notation $|S_T|$ is used to denote the number of outstanding targets in the set S_T. Further,

$$\mathbb{P}[|\bar{S_T}| = 0] = \mathbb{P}[|S_T| = 0]\mathbb{P}[|\bar{S_T} \backslash S_T| = 0] \leq \mathbb{P}[|S_T| = 0]. \tag{2.3}$$

We now calculate the probability that an infinitesimal area element ζ of length m and width m centered at $(s,0)$ shown in Fig. 2.3(b) is empty:

$$\begin{aligned}\mathbb{P}[|\zeta| = 0] &= \exp\left(-\lambda \frac{m}{v}\frac{1}{\pi D^2}\int_0^s r\theta dr\right) = \exp\left(-\lambda \frac{m}{v}\frac{1}{\pi D^2}\int_0^s r\frac{m}{s}dr\right) &(2.4)\\ &= \exp\left(\frac{-m^2}{v}\frac{\lambda s}{2\pi D^2}\right) \geq \exp\left(\frac{-m^2\lambda}{2\pi v D}\right) = \exp\left(\frac{-\lambda}{2\pi v D}\,\text{area}(\zeta)\right), &(2.5)\end{aligned}$$

where the inequality follows from the fact that $s \in [0, D]$, and the exponential function has a minimum at $s = D$. The last equality is true since $\text{area}(\zeta) = m^2$. Further, every

compact set can be written as a countable union of non-overlapping rectangles. Thus, Eq. (2.4) holds for the compact measurable set $\bar{S}_T$ as well. Then, using the results from Eq. (2.3) and Eq. (2.4),

$$\mathbb{P}[|S_T| = 0] \geq \mathbb{P}[|\bar{S}_T| = 0] \geq \exp\left(\frac{-\lambda}{2\pi v D}\,\text{area}(\bar{S}_T)\right) = \exp\left(\frac{-\lambda}{2\pi v D}\pi T^2\right),$$

and the expectation of T_d can be bounded as follows:

$$\begin{aligned}\mathbb{E}[T_d] &= \int_0^{+\infty} \mathbb{P}[T_d > T]dT = \int_0^{+\infty} \mathbb{P}[|S_T| = 0] \geq \int_0^{+\infty} \mathbb{P}[|\bar{S}_T| = 0]dT \\ &\geq \int_0^{+\infty} \exp\left(\frac{-T^2\lambda}{2vD}\right) dT \geq \frac{\sqrt{\pi}}{2}\sqrt{\frac{2vD}{\lambda}} = \sqrt{\frac{\pi v D}{2\lambda}}.\end{aligned}$$

so the result is obtained. ■

Theorem 8 (Policy Independent Upper Bound on Service Fraction) *An upper bound on the service fraction of any policy P for the RET problem satisfies*

$$\mathbb{F}_{\text{cap}}(P) \leq \sqrt{\frac{2}{\pi v \lambda D}}.$$

Proof: This follows from the fact that in order to service a fraction $c \in (0, 1]$ of targets, we require that the rate at which targets are serviced is more than the rate at which they arrive [60], i.e., $c\lambda\mathbb{E}[T] \leq 1$. Since $T > T_d$, the result now follows by using Lemma 7. ■

2.4.4 Bounds on paths and tours through targets

To distinguish static targets from moving targets, we introduce some terminology. A target moving radially outward is referred to as an escaping target. A target is said to have been 'captured' by the vehicle if the vehicle reaches the target before it escapes the

environment. The following results are used to estimate and bound the length of the path through targets in the environment.

Theorem 9 (Upper bound on path through escaping targets) *Let targets starting from* (r_i, θ_i), $i \in \{1, \ldots, N\}$ *move radially outward with speed* v. *Let* T *be the length of the path through these escaping targets in some arbitrary order* $\delta : \{1, \ldots, N\} \to \{1, \ldots, N\}$. *Let* T_s *be the length of the path through static targets located at* $(r_i + v\bar{T}, \theta_i)$, $i \in \{1, \ldots, N\}$ *processed in order* δ *and* $\bar{T} \geq T$. *Then,*

$$T \leq \frac{T_s}{1 - v}.$$

Proof: Without loss of generality, let the targets be labeled in the order in which they are processed. Let the vehicle take time T_j to service the $j-$th escaping target having serviced the $(j-1)-$th escaping target. Consider the $i-$th escaping target starting from (r_i, θ_i). The vehicle services this target at time $\sum_{j=1}^{i} T_j$. It then starts for the escaping target $i+1$ and reaches it in time T_{i+1}. Let $T_{i+1}^{'}$ be the distance between $(r_i + v\sum_{j=1}^{i+1} T_j, \theta_i)$ and $(r_{i+1} + v\sum_{j=1}^{i+1} T_j, \theta_{i+1})$. Also, let $T_{i+1}^{''}$ be the distance between $(r_i + vT, \theta_i)$ and $(r_{i+1} + vT, \theta_{i+1})$ while $T_{s,i+1}$ is the distance between $(r_i + v\bar{T}, \theta_i)$ and $(r_{i+1} + v\bar{T}, \theta_{i+1})$. Since the distance between two targets moving radially outward with the same speed is a non-decreasing function of time, $T_{i+1}^{'} \leq T_{i+1}^{''} \leq T_{s,i+1}$. Referring to Fig. 2.4, from the triangle inequality, $T_{i+1}^{'} + vT_{i+1} \geq T_{i+1}$, i.e., $T_{i+1} \leq (T_{i+1}^{'})/(1-v) \leq (T_{s,i+1})(1-v)$. Extending this to all the targets in the path,

$$T = \sum_{i=1}^{n} T_{i+1} \leq \sum_{i=1}^{n} \frac{T_{s,i+1}}{1 - v} = \frac{T_s}{1 - v}.$$

■

The upper bound on the length of the path through escaping targets is thus related

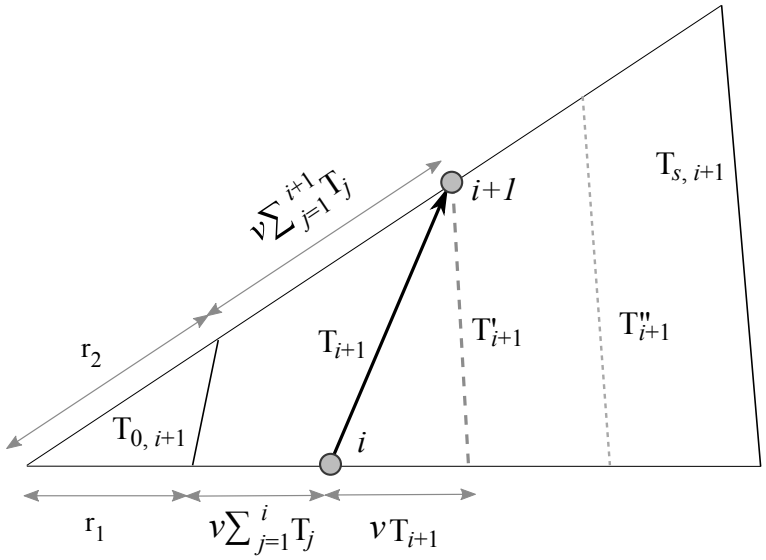

Figure 2.4: The thick line labeled T_{i+1} indicates the trajectory of the vehicle starting from the target i to service the target $i+1$. The gray circles indicate the locations at which the vehicle intercepts the targets.

to the length of the path through their static locations *in the future*.

Theorem 10 (Lower bound on path through escaping targets) *Let targets starting from* (r_i, θ_i), $i \in \{1, \ldots, N\}$ *move radially outward with speed* v. *Let* T *be the length of the path through these escaping targets in some arbitrary order* $\delta : \{1, \ldots, N\} \to \{1, \ldots, N\}$. *Let* T_0 *be the length of the path through static targets located at* (r_i, θ_i), $i \in \{1, \ldots, N\}$ *processed in order* δ. *Then,*

$$T \geq \frac{T_0}{1+v}.$$

Proof: The proof is similar to that of Theorem. 9 and is omitted for brevity. ■

Given a set K of n points, the Euclidean traveling salesperson problem (ETSP) is to determine the shortest tour, i.e., a closed path that visits each point exactly once. We now state the classic result providing a limit for the length of the ETSP through large number of points. We will leverage this result in the next section to derive tighter analytic bounds on the performance of our policies in regimes of high target arrival rates.

Theorem 11 (Asymptotic ETSP length,[96]) *If a set K of n points is distributed independently and identically in a compact set Q, then there exists a constant β such that*

$$\lim_{n\to+\infty} \frac{ETSP(K)}{\sqrt{n}} = \beta \int_Q \varphi(q)^{1/2} dq,$$

where φ is the density of the absolutely continuous part of the point distribution.

The constant β has been estimated numerically as $\beta \approx 0.7120 \pm 0.0002$ [81].

2.5 Policies

In this section, we propose three policies for the RET problem. The SAC policy is designed for low target arrival rates while the SW policy is designed for moderate target arrival rates. Finally, the SNB policy is proposed for high arrival rates.

2.5.1 Stay at Center (SAC) Policy

According to this policy, the vehicle stays at the optimal location in the disk and waits for new targets to appear in its capturable set. For $v \in [0, 0.5]$, this location is the center. The SAC policy is suitable for low target arrival rates at which the optimal vehicle location takes prominence.

Given a vehicle location $(x, 0) \in \mathcal{E}$, recall that $C(x, v, D)$ denotes the capturable set for the vehicle. Let x^* and ρ^* be defined as per Eq. (2.1). The formal description of the SAC policy is given in Algorithm 1.

Algorithm 1 has the following guarantee on capture fraction.

Theorem 12 (SAC Policy Capture Fraction) *The capture fraction of the SAC policy*

Algorithm 1: Stay At Center (SAC) policy

Given: v, D known and the vehicle placed at $(x^*, 0)$.

1 Intercept a target that appears inside $C(x^*, v, D)$;
2 Return back to $(x^*, 0)$;
3 Repeat from step 1.

satisfies

$$\mathbb{F}_{\text{cap}}(\text{SAC}) \geq \frac{\rho^*(v, D)}{2\rho^*(v, D)\lambda D + 1}.$$

If $v \in [0, 0.5]$ *so that the optimal vehicle location* $x^* = 0$*, then the above fraction becomes equal to*

$$\mathbb{F}_{\text{cap}}(\text{SAC}) \geq \frac{(1-v)^2}{2\lambda(1-v)^2 D + 1}.$$

Proof: See Appendix. ■

Remark 13 (Optimality in light load, i.e., $\lambda \to 0^+$) *In the light load regime of* $\lambda \to 0^+$*, the capture fraction achieved equals* $\rho^*(v, D)$*, which is exactly equal to the probability that a target falls within the capturable set* C *when the vehicle is located at the optimal location* $(x^*, 0)$*. Comparing with Theorem 4, we see that the SAC policy is optimal in this limiting regime.*

2.5.2 Sector-wise (SW) Policy

In the Sector-wise policy, the vehicle stays closer to the boundary and utilizes the high relative velocity of the outgoing targets. It starts every iteration at a radial location X and services the first target with the smallest clockwise angular separation in a specific subset associated with the iteration.

One such subset J_1 which the vehicle encounters in the first iteration is shown by the shaded region in Fig. 2.5. It then proceeds to the nearest location with radial coordinate X in the disk and waits for a specified time to begin its next iteration. The formal

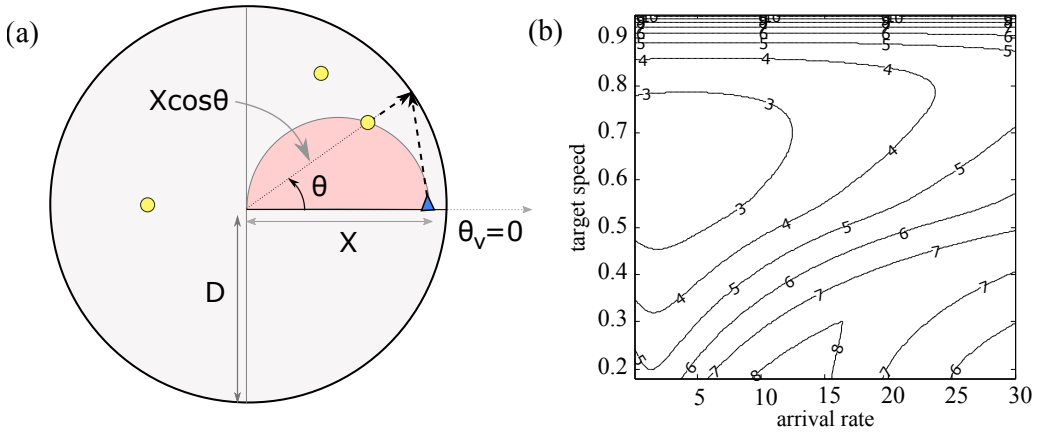

Figure 2.5: (a) Vehicle located at $p = (X, 0)$ services outstanding targets shown by the shaded region. (b) Factor of optimality of the SW policy in different parameter regimes v, λ when $D = 1$.

Algorithm 2: Sector-wise (SW) policy

Given: v, D known and the vehicle placed at $(X, 0)$.

Set $X = D\sqrt{1-v^2}$, $W = \max(0, X(1/4v - \sqrt{2}))$;
repeat
  if *there are targets in* $\mathcal{E}$ *with clockwise angular separation* $\theta < \pi/2$ *such that their radial coordinate* r *satisfies* $r \leq X\cos\theta$ **then**
    Service the target with smallest angular separation and move to nearest location in $\mathcal{E}$ with radial coordinate X ;
    Wait for time W and return to step 3.
  else
    Stay at current location.
  end
until *all targets are serviced or have escaped*;

description of the policy is given in Algorithm 2. While the SW policy is applicable in all parameter regimes of the RET problem, for a fixed speed v, it is constant factor optimal in moderate arrival regimes as established in Theorem 15. Algorithm 2 has the following guarantee on capture fraction.

Lemma 14 (SW Policy Capture Fraction) *The capture fraction of the SW policy satisfies*

$$\mathbb{F}_{\text{cap}}(\text{SW}) \geq \frac{1}{\lambda}\left(W + \frac{\pi D}{4}(3\eta_1(k) + \eta_2(k)) + \frac{8}{\lambda(1-v^2)}\eta_3(k)\right)^{-1},$$

where

$$\eta_1(k) = \left(L_{-1}(8k) - I_1(8k) - L_{-1}(20k) + I_1(20k)\right), \tag{2.6}$$

$$\eta_2(k) = \left(I_0(8k) - L_0(8k) - I_0(20k) + L_0(20k)\right), \tag{2.7}$$

$$\eta_3(k) = 1 - \pi/2\left(I_0(12k) - L_0(12k) - I_0(20k) + L_0(20k)\right), \tag{2.8}$$

$W = \max\left(0, D\sqrt{1-v^2}(1/4v - \sqrt{2})\right)$, $k = \lambda D(1-v^2)^{3/2}/72\pi v$, *and* I_0 *and* I_1 *are modified Bessel functions of the first kind and* L_0 *and* L_{-1} *are modified Struve functions [1].*

Proof: See Appendix. ■

Theorem 15 (Performance in moderate arrival rates) *For* $\lambda > \frac{7\pi v}{(1-v^2)^{3/2}D}$ *and* $v \in (1/4\sqrt{2}, 1)$*, the capture fraction of the SW policy satisfies*

$$\mathbb{F}_{\text{cap}}(\text{SW}) \geq \alpha(v)\sqrt{\frac{2}{\pi v \lambda D}},$$

where

$$\alpha(v) = \frac{\sqrt{v}}{\pi^2}\left(\left(\frac{v}{(1-v^2)^{3/2}}\right)^{1/2} + \frac{10}{3}\frac{(1-v^2)^{1/2}}{v}\right)^{-1/2}. \tag{2.9}$$

Proof: See Appendix. ■

Thus, for moderate arrival rates and $v \in (1/4\sqrt{2}, 1)$, using the result from Theorem 15 and the fundamental bound obtained in Theorem 8, the SW policy is also a constant factor policy with the factor equal to $1/\alpha(v)$.

2.5.3 Stay-Near-Boundary (SNB) Policy

We now introduce the SNB policy for the high arrival regime. In this regime, the density of targets accumulating close to the boundary of the disk is high. Hence, the distance traversed by the vehicle (and the time taken) between capturing consecutive

targets is small. Consequently, the distance by which the targets move between consecutive captures is also small. Hence, the vehicle can plan ahead and capture multiple targets in a single iteration. To determine the order of captures, it uses the solution to the Euclidean Minimum Hamiltonian Path (EMHP) problem which can be stated as follows:

> Given a set of n (stationary) points, determine the length of the shortest path which visits each point exactly once.

The SNB policy makes use of three parameters g, h and n_{cap}. At the beginning of every iteration, the vehicle computes an EMHP through the locations that the targets accumulated in $A(g, h, 0, 2\pi)$ will have after time $(D - h)$. This is done to levarage the result from Theorem 9. It uses the order obtained from the EMPH to service the first n_{cap} targets using constant bearing principle. A formal statement of the SNB policy is given in Algorithm 3.

The parameters g, h and n_{cap} are chosen in a way which ensures that the vehicle will service all the n_{cap} targets accumulated in $A(g, h, 0, 2\pi)$ at the beginning of the iteration before the last target escapes the environment. This can be achieved in the following way:

(i. parameters g and h are solutions to variables a and b respectively in the following

Optimization Problem:

$$\max_{a,b} \left(\frac{b^3 - a^3}{b^2 - a^2} \right)$$

subject to

$$\mu_A = \frac{\lambda(b^3 - a^3)}{3vD^2},$$
$$\frac{\beta}{1-v}\sqrt{\frac{6\pi}{b^3 - a^3}} \left(\frac{b^2 - a^2}{2} \right) \sqrt{\mu_A(1+v)} \leq \frac{D-b}{v},$$
$$\frac{\beta}{1+v}\sqrt{\frac{6\pi}{b^3 - a^3}} \left(\frac{b^2 - a^2}{2} \right) \sqrt{\mu_A(1-v)} \geq \frac{b-a}{v},$$
$$0 \leq a < b < D.$$

(ii. parameter n_{cap} is set as follows:

$$n_{\text{cap}} := \frac{\lambda(1-v)}{3vD^2}(h^3 - g^3).$$

Algorithm 3 has the following guarantee on the capture fraction of the RET problem.

Algorithm 3: Stay-Near-Boundary (SNB) policy

Given: g, h and n_{cap} are known and the vehicle is at $(h, 0)$.
if $A(g, h, 0, 2\pi)$ *contains outstanding targets* **then**
  $s_1 :=$ set of locations of outstanding targets in $A(g, h, 0, 2\pi)$;
  $s_2 :=$ set of their locations if they move radially outward by distance $(D - h)$;
  $\Psi :=$ order of the EMHP starting from $(D, 0)$, visiting targets in s_2 and ending at $(D, 0)$;
  service the first n_{cap} targets in s_1 in order from Ψ using constant bearing principle and return to $(h, 0)$.
end

Theorem 16 (SNB Policy Capture Fraction) *For any fixed $v \in (0, 1)$, in the limit as*

$\lambda \to +\infty$, the capture fraction of the SNB policy satisfies

$$\mathbb{F}_{\text{cap}}(\text{SNB}) \geq \frac{2p}{7\beta}\sqrt{\frac{2}{\pi\lambda v D}}$$

with probability one, where

$$p := p(D) = \begin{cases} 1, & D > 1, \\ \dfrac{5\sqrt{D}}{6}, & \textit{otherwise.} \end{cases}$$

Proof: See Appendix. ■

Corollary 17 (Performance of the SNB policy) *In the limit as $\lambda \to +\infty$ such that $\lambda > (1+v)^2/2\pi\beta^2 v(1-v)$, the SNB policy is within a factor $7\beta/2p$ of the optimal. For $D \geq 1$, this factor is ≈ 2.52.*

2.6 Simulations

The numerical performance of the SAC and SW policies for arrival rates of $\lambda = 2$ and $\lambda = 10$ respectively and all target speeds is shown in Fig. 2.6. The parameter $D = 1$ for these simulations. The mean of the capture fraction based on 1000 simulations is shown along with its standard deviation. It agrees well with the theoretical lower bounds. The theoretical bounds are still conservative. For the SAC policy, the conservativeness comes from the application of Jensen's inequality in Eq. (2.10). For the SW policy, the conservativeness of the bound is because of inequalities introduced in Eq. (2.11),(2.13) to bound integrals.

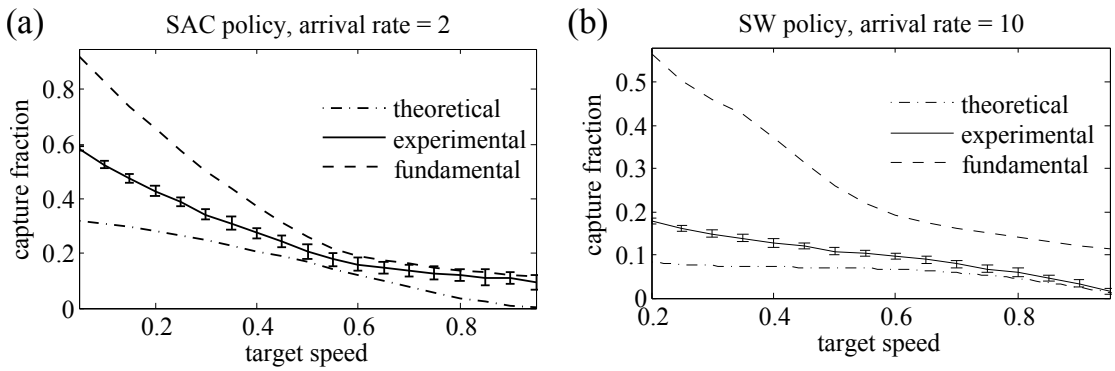

Figure 2.6: Performance of the (a) SAC and (b) SW policies for arrival rates $\lambda = 2$ and $\lambda = 10$ respectively for the RET problem with $D = 1$. The theoretical bounds are from Theorem 12 and Theorem 15 respectively.

Appendix

Proof of Theorem 12: Notice that if $m_{\text{cap}}(t) > 0$ for some $t > 0$, then

$$\limsup_{t\to+\infty} \mathbb{E}\left[\frac{m_{\text{cap}}(t)}{m_{\text{cap}}(t)+m_{\text{miss}}(t)}\right] = \limsup_{t\to+\infty} \mathbb{E}\left[\frac{1}{1+\frac{m_{\text{miss}}(t)}{m_{\text{cap}}(t)}}\right] \geq \left(1 + \limsup_{t\to+\infty} \mathbb{E}\left[\frac{m_{\text{miss}}(t)}{m_{\text{cap}}(t)}\right]\right)^{-1}, \tag{2.10}$$

where the last step comes from an application of Jensen's inequality [21]. Thus, we can determine a lower bound on the capture fraction by studying the number of targets that escape per captured target. Consider a tagged target i which falls within $C(x^*, v, D)$. The time t_i taken by vehicle to intercept target i and return to the optimal location satisfies $t_i \leq 2D$. Therefore, the number of targets that escape because the vehicle intercepts the i-th target is equal to the sum of 1) the number of targets that arrive anywhere in the environment during the time interval of t_i and 2) the number of targets that are generated outside of $C(x^*, v, D)$ while the vehicle is waiting for the next capturable target. Since the target arrival process is temporally Poisson, the expected number of targets in case 1 are given by $\lambda t_i \leq 2\lambda D$. The spatial distribution of the targets is uniform random. Further, $\text{area}(C(x^*, v, D)) = \rho^*(v, D)\pi D^2$. Therefore, the targets missed in case 2, denoted by

N_{miss} is a random variable distributed as follows.

$$N_{\text{miss}} = \begin{cases} 0, \text{ with probability } \rho^*(v,D), \\ 1, \text{ with probability } \rho^*(v,D)\Big(1-\rho^*(v,D)\Big), \\ 2, \text{ with probability } \rho^*(v,D)\Big(1-\rho^*(v,D)\Big)^2, \\ \vdots \\ k, \text{ with probability } \rho^*(v,D)\Big(1-\rho^*(v,D)\Big)^k, \\ \vdots \end{cases}$$

Therefore,

$$\begin{aligned} \mathbb{E}\left[N_{\text{miss}}\right] &= \sum_{k=1}^{\infty} k\rho^*(v,D)\Big(1-\rho^*(v,D)\Big)^k = \rho^*(v,D)\sum_{k=1}^{\infty}\Big(1-\rho^*(v,D)\Big)^k \\ &= \rho^*(v,D)\frac{1-\rho^*(v,D)}{(\rho^*(v,D))^2} = \frac{1}{\rho^*(v,D)} - 1. \end{aligned}$$

Substituting the upper bound for case 1 and the expression for case 2 in (2.10), we obtain

$$\mathbb{F}_{\text{cap}}(\text{SAC}) \geq \frac{1}{2\lambda D + \frac{1}{\rho^*(v,D)}} = \frac{\rho^*(v,D)}{2\rho^*(v,D)\lambda D + 1}.$$

Proof of Theorem 14: In the sector-wise policy, the vehicle starts every iteration at a distance $X = D\sqrt{1-v^2}$ from the center. If θ_v is the angular position of the vehicle in its i-th iteration and

$$J_i := \left\{(r,\theta) \mid 0 \leq r \leq X\cos(\theta-\theta_v), \theta-\theta_v \in [0,\pi/2]\right\},$$

then if there is an outstanding target in J_i, the vehicle services the target in J_i with the smallest angular separation from θ_v in the counterclockwise direction. The choice of X

ensures that the vehicle always services any target in J_i before it escapes the disk.

We now calculate the expectation of the time required for a single iteration of the SW policy. Without loss of generality we assume that $i = 1$, $\theta_v = 0$ initially and the vehicle is at $(X, 0)$. We also assume that the environment is unserviced. Let $K(\gamma_1, \gamma_2) := \{(r, \phi) | 0 \leq r \leq X \cos\phi, \phi \in [\gamma_1, \gamma_2]\}$ for $\gamma_1 < \gamma_2$. Also, for infinitesimal $\delta\theta$, let $\theta^+ = \theta + \delta\theta$. Then the probability of the first outstanding target in J_1 being at an angular location θ, i.e.

$$\begin{aligned}\mathbb{P}[\text{first target is in } K(\theta, \theta^+) |\ J_1 \text{ is not empty }] =& \mathbb{P}\left[|K(0,\theta)| = 0\right] \mathbb{P}\left[|K(0,\theta^+)| \neq 0\right] \\ =& \exp\left(-k\left(9\sin\theta + \sin 3\theta\right)\right) \\ & \times \left(1 - \exp\left(-k\left(9\sin\theta^+ + \sin 3\theta^+\right)\right)\right) \\ \geq& \exp(-8k\sin\theta)\left(1 - \exp(-12k\sin\theta)\right), \end{aligned} \tag{2.11}$$

where $k = \frac{\lambda X^3}{72\pi v D^2}$. A number of results are used to obtain Eq. (2.11). The first result, which is derived in the same spirit as Eq. (2.4), is that for $\alpha \in [0, 2\pi]$,

$$\mathbb{P}\left[|K(0,\alpha)| = 0\right] = \exp\left(-k\left(9\sin\alpha + \sin 3\alpha\right)\right). \tag{2.12}$$

The second supporting result is the following empirically obtained inequality for $\alpha \in [0, \pi/2]$: $12\sin\alpha \geq 9\sin\alpha + \sin 3\alpha \geq 8\sin\alpha$.

Now that we have an expectation of the first target being at an angular location θ relative to the vehicle, we calculate the time taken to capture this target. Let T_θ be the random variable denoting the time taken to start from $(X, 0)$, service a target at (r, θ) and go to (X, θ) to start the next iteration. We determine a bound on the expectation of T_θ. Once again, we assume that the environment is unserviced and note

that the probability distribution of the outstanding targets is given by $f_a(r) = \lambda r^2/vD^2$ as obtained in Lemma 6. Since $r \leq X\cos\theta$, we use Lemma 2 to obtain a lower bound on the expectation of T_θ:

$$\begin{aligned}\mathbb{E}[T_\theta] &\leq \frac{2\displaystyle\int_{r=0}^{X\cos\theta}\left(\frac{\lambda r^2}{vD^2}\right)\left(\frac{\sqrt{X^2+r^2-2Xr\cos\theta}}{\sqrt{1-v^2}}\right)dr}{\displaystyle\int_{r=0}^{X\cos\theta}\left(\frac{\lambda r^2}{vD^2}\right)dr} \\ &= \frac{2\displaystyle\int_{s=0}^{\cos\theta} X\left(\frac{\lambda s^2}{vD^2}\right)\left(\frac{\sqrt{1+s^2-2s\cos\theta}}{\sqrt{1-v^2}}\right)ds}{\displaystyle\int_{s=0}^{\cos\theta}\left(\frac{\lambda s^2}{vD^2}\right)ds} \qquad (2.13)\end{aligned}$$

$$\leq \frac{6X}{\sqrt{1-v^2}}\left(\frac{\sin\theta}{4}+\frac{1}{12}\right) =: \Gamma(\theta). \qquad (2.14)$$

The factor of two is required since the vehicle has to go to (X,θ) to start the next iteration and the time required for this is always less than or equal to the time required to service the target at (r,θ) starting from $(X,0)$. If T is the random variable denoting the time required to start from $(X,0)$, service the first target in J_1 and return to the radial location X, then using the result from Eq. (2.11),

$$\begin{aligned}\mathbb{E}[T|J_1 \text{ is not empty}] &= \int_{\theta=0}^{\pi/2}\mathbb{E}[T_\theta]\ \mathbb{P}[\text{ first target in } K(\theta,\theta^+)]d\theta \\ &\leq \int_{\theta=0}^{\pi/2}\Gamma(\theta)\left(\exp(-8k\sin\theta)\left(1-\exp(-12k\sin\theta)\right)\right)d\theta \\ &= \left(\frac{6X}{\sqrt{1-v^2}}\right)\left(\frac{\pi}{8}\eta_1(k)+\frac{\pi}{24}\eta_2(k)\right). \qquad (2.15)\end{aligned}$$

$$(2.16)$$

where the functions η_1 and η_2 are as defined in Eq. (2.6). Further, using the result from

Eq. (2.12),

$$\begin{aligned}\mathbb{P}[\ J_1 \text{ is empty }] \leq & 1 - \int_0^{\pi/2} \exp(-12k\sin\theta)\left(1-\exp(-8k\sin\theta)\right)\ d\theta \\ = & 1 - \frac{\pi}{2}\left(I_0(12k) - L_0(12k) - I_0(20k) + L_0(20k)\right) \\ = & \eta_3(k), \end{aligned} \tag{2.17}$$

and the expected time that the vehicle has to wait for a new target to appear in J_1 is less than $8/\lambda(1-v^2)$ since the area of J_1 is $(1-v^2)/8$ times the area of the disk. This is in addition to the time W that the vehicle waits at the beginning of the iteration. So, using Eq. (2.15) and Eq. (2.17), the time T taken to finish a single iteration of the SW policy has the following expectation:

$$\begin{aligned}\mathbb{E}[T] = & W + \mathbb{E}[T|\ J_1 \text{ is not empty }] + \mathbb{E}[T|\ J_1 \text{ is empty }] \\ = & W + \mathbb{E}[T|\ J_1 \text{ is not empty }] + \frac{8}{\lambda(1-v^2)}\mathbb{P}[\ J_1 \text{ is empty }] \\ \leq & W + \left(\frac{6X}{\sqrt{1-v^2}}\right)\left(\frac{\pi}{8}\eta_1(k) + \frac{\pi}{24}\eta_2(k)\right) + \left(\frac{8}{\lambda(1-v^2)}\right)\eta_3(k) \\ = & W + \frac{3\pi D}{4}\eta_1(k) + \frac{\pi D}{4}\eta_2(k) + \frac{8}{\lambda(1-v^2)}\eta_3(k). \end{aligned}$$

Then, $\mathbb{F}_{\text{cap}}(SW) \geq 1/\lambda\mathbb{E}[T]$. Finally, in the most favorable scenario for the vehicle, it intercepts each new target at the end of each quadrant at a radial location X, waits for time W and begins a new iteration. The time in which it returns to a quadrant in this manner is equal to $4\sqrt{2}X + 4W$. Since $X/v < 4\sqrt{2}X + 4W$ for all $v < 1$, the assumption that the vehicle always begins an iteration in an unserviced region holds true.

Proof of Theorem 15: From Lemma 14, we know that the capture fraction of the SW

policy satisfies

$$\mathbb{F}_{\text{cap}}(\text{SW}) \geq \frac{1}{\lambda}\left(W + \frac{\pi D}{4}(3\eta_1(k) + \eta_2(k)) + \frac{8}{\lambda(1-v^2)}\eta_3(k)\right)^{-1}. \tag{2.18}$$

When $v > 1/4\sqrt{2}$, $W = 0$. Further, when $\lambda > \frac{7\pi v}{(1-v^2)^{3/2}D}$, $k = \frac{\lambda(1-v^2)^{3/2}}{72\pi v} > 0.1$. Using upper and lower bounds on Bessel and Struve functions, the following hold true for $k > 0.1$,

$$\begin{aligned}\frac{3}{4}\eta_1(k) + \frac{1}{4}\eta_2(k) = \frac{3}{4}(L_{-1}(8k) - I_1(8k) - L_{-1}(20k) + I_1(20k)) \\ + \frac{1}{4}\left(I_0(8k) - L_0(8k) - I_0(20k) + L_0(20k)\right) \leq \frac{1}{12\sqrt{k}},\end{aligned} \tag{2.19}$$

$$\eta_3(k) = 1 - \frac{\pi}{2}\left(I_0(12k) - L_0(12k) - I_0(20k) + L_0(20k)\right) \leq \frac{5\sqrt{k}}{2}. \tag{2.20}$$

Using Eq. 2.19 and 2.20, and the result in Eq. 2.18, $\mathbb{F}_{\text{cap}}(\text{SW})$ can be bounded and the result is obtained.

Proof of Theorem 16 We start with calculating an upper bound on the length of the tour through all the targets in $A(g, h, 0, 2\pi)$. Let $Q := \{(r, \theta) \in A(g, h, 0, 2\pi)\}$ be the set of locations of targets accumulated in $A(g, h, 0, 2\pi)$ and $n = |Q|$. From Lemma 6, the normalized distribution of these targets w.r.t the radial location x is given by $f_n(x) = 3x^2/(h^3 - g^3)$ for $x \in [g, h]$. Let $\bar{Q}$ be the set of locations (s, ϕ) of these targets if they move outwards by distance d and occupy $A(g + d, h + d, 0, 2\pi)$. The normalized distribution functions of the random variables s and ϕ which denote locations of these

targets are

$$f_s(x) = \frac{3(x-d)^2}{h^3 - g^3} \quad \text{and} \quad e_\phi(y) = \frac{1}{2\pi}.$$

Using Theorem 11 and assuming that $n \to \infty$ (which we will revisit later),

$$\lim_{n \to +\infty} \frac{ETSP(\bar{Q})}{\sqrt{n}} = \beta \sqrt{\frac{6\pi}{h^3 - g^3}} \left(\frac{h^2 - g^2}{2} \right)$$

with

$$\varphi(s, \phi) = f_s(x) e_\phi(y) = \frac{3(x-d)^2}{(h^3 - g^3)} \frac{1}{2\pi}.$$

Using Chebyshev's inequality, if μ_A and σ_A are the mean and standard deviation of the random variable n, then for any fixed $v \in (0, 1)$,

$$\mathbb{P}[n < \mu_A(1+v)] \geq 1 - \sigma_A^2 / v^2 \mu_A^2.$$

Then, the condition

$$t_u := \frac{\beta}{1-v} \sqrt{\frac{6\pi}{h^3 - g^3}} \left(\frac{h^2 - g^2}{2} \right) \sqrt{\mu_A(1+v)} \leq \frac{D-g}{v} \tag{2.21}$$

from the Optimization Problem (i ensures that the n targets will be serviced before they escape the disk with at least a probability of $1 - \sigma_A^2 / v^2 \mu_A^2$. Similarly since $n_{\text{cap}} = \mu_A(1-v)$ and $v > 0$,

$$\mathbb{P}[n > \mu_A(1-v)] \geq 1 - \sigma_A^2 / v^2 \mu_A^2 \tag{2.22}$$

so that $n > n_{\text{cap}}$ and the vehicle services n_{cap} targets in an iteration with probability of at least $1 - \sigma_A^2 / v^2 \mu_A^2$. Further, the condition

$$\frac{\beta}{1+v} \sqrt{\frac{6\pi}{h^3 - g^3}} \left(\frac{h^2 - g^2}{2} \right) \sqrt{\mu_A(1-v)} \geq \frac{h-g}{v} \tag{2.23}$$

from the Optimization Problem (i ensures that with probability of at least $1-\sigma_A^2/v^2\mu_A^2$, when the vehicle starts an iteration, $A(g,h,0,2\pi)$ is unserviced. In the above inequality, the left-hand side is the lower bound on the length of the tour through $\mu_A(1-v)$ targets in $A(g,h,0,2\pi)$ obtained by using Theorem 10. We also know that

$$\mu_A = \frac{\lambda(h^3-g^3)}{3vD^2}$$

and $\sigma_A^2 = \mu_A$ from Lemma 6. When $\lambda \to +\infty$ and $v \in (0,1)$, then $\mu_A \to +\infty$ and $n_{\text{cap}} \to +\infty$ so that Eq. (2.21),(2.22) and (2.23) hold true with probability one. Then, since $n_{\text{cap}} \to +\infty$ and $n > n_{\text{cap}}$ with probability one, our earlier assumption that $n \to +\infty$ is true as well.

If $k_{\text{tot}}(i)$ and $k_{\text{cap}}(i)$ are the number of targets that have appeared and have been serviced in the $i-$th iteration of the SNB policy, and $\mathbb{F}_{\text{i}}(\text{SNB}) = \mathbb{E}\left[k_{\text{cap}}(i)/k_{\text{tot}}(i)\right]$, then since at every iteration, $k_{\text{cap}}(i) \geq n_{\text{cap}}$,

$$\mathbb{F}_{\text{i}}(\text{SNB}) \geq \mathbb{E}\left[\frac{n_{\text{cap}}}{k_{\text{tot}}(i)}\right] = n_{\text{cap}}\mathbb{E}\left[\frac{1}{k_{\text{tot}}(i)}\right] \geq \frac{n_{\text{cap}}}{\mathbb{E}[k_{\text{tot}}(i)]}, \tag{2.24}$$

where the last inequality holds true using Jensen's inequality for convex function $1/k_{\text{tot}}(i)$.

Next, when $\lambda > (1+v)^2/2\pi\beta^2 v(1-v)$, the solution to the Optimization Problem (i exists and the parameters g and h obtained by solving it satisfy

$$\frac{h^3-g^3}{h^2-g^2} \geq \frac{4Dp}{7}\frac{1-v}{\sqrt{1+v}}.$$

Now, when $\lambda \to +\infty$, for a fixed speed, the above condition on λ is met. Then, from Eq. 2.24 and using the fact that $\mathbb{E}[k_{\text{tot}}(i)] \leq \lambda t_u$,

$$\mathbb{F}_{\text{i}}(\text{SNB}) \geq \frac{2p(1-v)^2}{7(1+v)}\frac{1-v}{\beta}\sqrt{\frac{2}{\pi\lambda v}}.$$

Let the countably infinite set $Y := \{\mathbb{F}_{\mathrm{i}}(\mathrm{SNB})\ \forall i \in \mathbb{N}\}$. Also, let the uncountable set $Z := \{\mathbb{E}\left[m_{\mathrm{cap}}(t)/m_{\mathrm{tot}}(t)\right] \forall t \in \mathbb{R}_{\geq 0}\}$. Since $Y \subseteq Z$,

$$\mathbb{F}_{\mathrm{cap}}(\mathrm{SNB}) = \limsup_{t\to\infty} \mathbb{E}\left[\frac{m_{\mathrm{cap}}(t)}{m_{\mathrm{tot}}(t)}\right] \geq \limsup_{i\to\infty} \mathbb{F}_{\mathrm{i}}(\mathrm{SNB}) \geq \frac{2p(1-v)^2}{7(1+v)} \frac{1-v}{\beta} \sqrt{\frac{2}{\pi\lambda v}},$$

the result is obtained.

Chapter 3

Robotic Surveillance: Detection of Intruder Location

In this chapter we study a surveillance problem in which unknown intruder locations have to be detected in networked environments. This is achieved by designing Markov chain random walks for the surveillance vehicle which have minimum mean first passage time. Motivating examples for this problem include the monitoring of oil spills [29], the detection of forest fires [57], the tracking of border changes [97], and the periodic patrolling of an environment [39, 76]. Other applications include minimizing emergency vehicle response times [14] as well as servicing tasks in robotic warehouse management [106].

In areas of research outside of robotics, the study of the mean first passage time is of general mathematical and engineering interest. Similar to the fastest mixing Markov chain, the mean first passage time is a metric by which to gauge the performance of a random walk. The mean first passage time is also potentially useful in determining how quickly information propagates in an online network [9] or how quickly an epidemic spreads through a contact network [105].

3.1 Contributions

The contributions of this chapter [1] are as follows. First, we provide a convex optimization framework to minimize the Kemeny constant of a reversible Markov chain given the underlying graph topology of the random walk and the desired stationary distribution. Second, we extend the formulation of the *mean first passage time* to the network environments with non-homogeneous travel times, a generalization not yet looked at in the literature. We denote this extension the *weighted Kemeny constant*. Third, we derive a closed form solution for the *weighted Kemeny constant* and show its relation to the Kemeny constant. Fourth, we provide a convex optimization framework to minimize the weighted Kemeny constant of a Markov chain with desired stationary distribution. Fifth, we provide a semidefinite program (SDP) formulation for the optimization of the Kemeny constant and the weighted Kemeny constant. Finally, we look at two stochastic surveillance scenarios; in the first scenario we provide a setup in which minimizing the weighted Kemeny constant leads to the optimal Markov-chain strategy. In the second surveillance scenario we establish through numerical simulation that the Markov chain with the minimum weighted Kemeny constant performs substantially better compared with other well-known Markov chains like the the fastest mixing chain and the Metropolis-Hastings Markov chain.

3.2 Organization

We state the contributions of the chapter is Section 3.1. We then summarize the notation which we use throughout this chapter and the next. We also briefly review properties of Markov chains. In Section 3.3 we give background for the Kemeny constant and present our results for its minimization. In Section 3.4 we introduce and provide

[1] This is joint work with Rushabh Patel

detailed characterization of the weighted Kemeny constant as well as its minimization. In Section 3.5 we provide practical surveillance applications of the weighted Kemeny constant. In the final Section we present our conclusions and future research directions.

Notation

We use the notation $A = [a_{ij}]$ to denote a matrix A with the element a_{ij} in its i-th row and j-th column and, unless otherwise indicated, use bold-faced letters to denote vectors. Letting δ_{ij} denote the Kronecker delta, $A_{\rm d} = [\delta_{ij} a_{ij}]$ represents the diagonal matrix whose diagonal elements are the diagonal elements of the matrix A. The column vector of all ones and length n is denoted by $\mathbb{1}_n \in \mathbb{R}^{n\times 1}$ and I represents the identity matrix of appropriate dimension. We use diag[$\boldsymbol{b}$] to denote the diagonal matrix generated by vector $\boldsymbol{b}$ and $\mathbf{Tr}[A]$ to denote the trace of matrix A.

Properties of Markov chains

A *Markov chain* is a sequence of random variables taking value in the *finite* set $\{1, \ldots, n\}$ with the Markov property, namely that, the future state depends only on the present state; see [50, 55] for more details. Let $X_k \in \{1, \ldots, n\}$ denote the location of a random walker at time $k \in \{0, 1, 2, \ldots\}$. Some terminology for Makov chains follows:

(1) A Markov chain is *time-homogeneous* if $\mathbb{P}[X_{n+1} = j | X_n = i] = \mathbb{P}[X_n = j | X_{n-1} = i] = p_{ij}$, where $P \in \mathbb{R}^{n\times n}$ is the transition matrix of the Markov chain.

(2) The vector $\boldsymbol{\pi} \in \mathbb{R}^{n\times 1}$ is a *stationary distribution* of P if $\sum_{i=1}^{n} \boldsymbol{\pi}_i = 1$, $0 \leq \boldsymbol{\pi}_i \leq 1$ for all $i \in \{1, \ldots, n\}$ and $\boldsymbol{\pi}^T P = \boldsymbol{\pi}^T$.

(3) A time-homogeneous Markov chain is said to be *reversible* if $\boldsymbol{\pi}_i p_{ij} = \boldsymbol{\pi}_j p_{ji}$, for all $i, j \in \{1, \ldots, n\}$. For reversible Markov chains, $\boldsymbol{\pi}$ is always a steady state

distribution.

(4) A Markov chain is *irreducible* if there exists a t such that for all $i, j \in \{1, \ldots, n\}$, $(P^t)_{ij} > 0$.

(5) If the Markov chain is *irreducible*, then there is a unique stationary distribution $\boldsymbol{\pi}$, and the corresponding eigenvalues of the transition matrix, λ_i for $i \in \{1, \ldots, n\}$, are such that $\lambda_1 = 1$, $|\lambda_i| \leq 1$ and $\lambda_i \neq 1$ for $i \in \{2, \ldots, n\}$.

For further details on irreducible matrices and about results (4) and (5) see [71, Chapter 8]. In this and the next chapter we consider finite irreducible time-homogeneous Markov chains.

3.3 The Kemeny constant and its minimization

Consider a undirected weighted graph $\mathcal{G} = (\mathcal{V}, E, P)$ with node set $\mathcal{V} := \{1, \ldots, n\}$, edge set $E \subset \mathcal{V} \times \mathcal{V}$, and weight matrix $P = [p_{ij}]$ with the property that $p_{ij} \geq 0$ if $(i, j) \in E$ and $p_{ij} = 0$ otherwise. We interpret the weight of edge (i, j) as the probability of moving along that edge. Therefore, element p_{ij} in the matrix represents the probability with which the random walk visits node j from node i. Throughout this document we assume that the underlying undirected graph $(\mathcal{V}, E)$ associated to the transition probabilities P is connected.

In this section we look into a discrete-time random walk defined by a Markov chain on a graph $\mathcal{G}$. At each time step (*hop*) of the random walk we move to a new node or stay at the current node according to the transition probabilities described by a transition matrix P as discussed above. We do this with three objectives in mind. The first objective is to analyze the random walk and characterize the average visit time between nodes in the graph. The second objective is to minimize the average visit time between any two

nodes and the final is to achieve a long term (infinite horizon) visit frequency $\boldsymbol{\pi}_i$ at node i. Here, the frequency $\boldsymbol{\pi}_i$ is the ratio of visits to node i divided by the total number of visits to all nodes in the graph. Throughout this chapter, we describe the random walk using realizations of a Markov chain with transition matrix $P = [p_{ij}]$.

3.3.1 The mean first passage time for a weighted graph

Let $X_k \in \{1, \ldots, n\}$ denote the location of the random walker at time $k \in \{0, 1, 2, \ldots\}$. For any two nodes $i, j \in \{1, \ldots, n\}$, the *first passage time from i to j*, denoted by T_{ij}, is the first time that the random walker starting at node i at time 0 reaches node j, that is,

$$T_{ij} = \min\{k \geq 1 \mid X_k = j \text{ given that } X_0 = i\}.$$

It is convenient to introduce the shorthand $m_{ij} = \mathbb{E}[T_{ij}]$, and to define the *mean first passage time matrix* M to have entries m_{ij}, for $i, j \in \{1, \ldots, n\}$. The *mean first passage time from start node i*, denoted by $\boldsymbol{k}_i$, is the expected first passage time from node i to a random node selected according to the stationary distribution $\boldsymbol{\pi}$, i.e.,

$$\boldsymbol{k}_i = \sum_{j=1}^{n} m_{ij} \boldsymbol{\pi}_j.$$

It is well known [52] that the mean first passage time from a start node is independent of the start node, that is, $\boldsymbol{k}_i = \boldsymbol{k}_j$ for all $i, j \in \{1, \ldots, n\}$. Accordingly, we let $K = \boldsymbol{k}_i$, for all $i \in \{1, \ldots, n\}$, denote the *mean first passage time*, also known as the Kemeny constant, of the Markov chain.

Next, we provide formulas for these quantities. By definition, the first passage time

from i to j satisfies the recursive formula:

$$T_{ij} = \begin{cases} 1, & \text{with probability } p_{ij}, \\ T_{kj} + 1, & \text{with probability } p_{ik}, k \neq j. \end{cases}$$

Taking the expectation, we compute

$$m_{ij} = p_{ij} + \sum_{k=1, k\neq j}^{n} p_{ik}(m_{kj} + 1) = 1 + \sum_{k=1, k\neq j}^{n} p_{ik} m_{kj},$$

or in matrix notation,

$$(I - P)M = \mathbb{1}_n \mathbb{1}_n^T - PM_{\mathrm{d}}, \tag{3.1}$$

where P is the transition matrix of the Markov chain. If the Markov chain is irreducible with stationary distribution $\boldsymbol{\pi}$, then one can show $M_{\mathrm{d}} = \mathrm{diag}[\{1/\boldsymbol{\pi}_1, \ldots, 1/\boldsymbol{\pi}_n\}]$, and

$$\boldsymbol{\pi}^T M \boldsymbol{\pi} = \sum_{i=1}^{n} \boldsymbol{\pi}_i \sum_{j=1}^{n} \boldsymbol{\pi}_j m_{ij} = \sum_{i=1}^{n} \boldsymbol{\pi}_i \boldsymbol{k}_i = K.$$

Clearly, the Kemeny constant can be written as the function $P \mapsto K(P)$, however, to ease notation we simply write K and use $K(P)$ only when we wish to emphasize the constant's dependence on P.

The Kemeny constant $K = \boldsymbol{\pi}^T M \boldsymbol{\pi}$ can be computed from equation (3.1) or can be expressed as a function of the eigenvalues of the transition matrix P as is stated in the following theorem [52].

Theorem 18 *(Kemeny constant of an irreducible Markov chain): Consider a Markov chain with an irreducible transition matrix P with eigenvalues $\lambda_1 = 1$ and λ_i, $i \in$*

$\{2, \ldots, n\}$. *The Kemeny constant of the Markov chain is given by*

$$K = 1 + \sum_{i=2}^{n} \frac{1}{1 - \lambda_i}.$$

Using Theorem 18, we derive the following equivalent expression for *reversible* Markov chains in terms of the trace of a symmetric positive definite matrix. Before stating our result, we first introduce some notation. Given a stationary distribution vector $\boldsymbol{\pi} \in \mathbb{R}^{n\times 1}$ for a Markov chain with transition matrix $P \in \mathbb{R}^{n\times n}$, we define the matrix $\Pi \in \mathbb{R}^{n\times n}$ as $\Pi = \text{diag}[\boldsymbol{\pi}]$ and the vector $\boldsymbol{q} \in \mathbb{R}^{n\times 1}$ as $\boldsymbol{q}^T = (\sqrt{\boldsymbol{\pi}_1}, \ldots, \sqrt{\boldsymbol{\pi}_n})$. We are now ready to state our first result.

Theorem 19 *(Kemeny constant of a reversible irreducible Markov chain): The Kemeny constant of a reversible irreducible Markov chain with transition matrix P and stationary distribution $\boldsymbol{\pi}$ is given by*

$$K = \mathbf{Tr}\left[(I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)^{-1}\right]. \tag{3.2}$$

Proof: We start by noting that P is an irreducible row-stochastic matrix therefore the eigenvalues of P are $\{\lambda_1 = 1, \lambda_2, \ldots, \lambda_n\}$, where $|\lambda_i| \leq 1$ and $\lambda_i \neq 1$ for $i \in \{2, \ldots, n\}$. It follows that the eigenvalues of $(I - P)$ are $\{0, 1 - \lambda_2, \ldots, 1 - \lambda_n\}$. Since P is irreducible and reversible, there exists a stationary distribution $\boldsymbol{\pi} \in \mathbb{R}^n_{>0}$ implying Π is invertible and that $\Pi^{1/2}(I - P)\Pi^{-1/2} = I - \Pi^{1/2} P \Pi^{-1/2}$ is symmetric. It can easily be verified that $I - P$ and $I - \Pi^{1/2} P \Pi^{-1/2}$ have the same eigenvalues and that $\boldsymbol{q}$ is the eigenvector associated with the eigenvalue at 0. Next, notice the matrix $(I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)$ is symmetric and that $(I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)\boldsymbol{q} = \boldsymbol{q}$. Therefore, $(I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)$ has an eigenvalue at 1 associated with the vector $\boldsymbol{q}$. Since $(I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)$ is symmetric, the eigenvectors corresponding to different eigenvalues are orthogonal;

implying for eigenvector $\boldsymbol{v} \neq \boldsymbol{q}$ that $(I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)\boldsymbol{v} = (I - \Pi^{1/2} P \Pi^{-1/2})\boldsymbol{v}$ since the eigenvalue at 1 is simple. Therefore, the eigenvalues of $(I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)$ are $\{1, 1-\lambda_2, \ldots, 1-\lambda_n\}$. Thus, $K = \mathbf{Tr}\left[(I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)^{-1}\right] = 1 + 1/(1-\lambda_2) + \ldots + 1/(1-\lambda_n) = K$. ■

Given the above result, we are now ready to state our first problem of interest.

Problem 1 *(Optimizing the Kemeny constant of a reversible Markov chain): Given the stationary distribution $\boldsymbol{\pi}$ and graph $\mathcal{G}$ with vertex set $\mathcal{V}$ and edge set E, determine the transition probabilities $P = [p_{ij}]$ solving:*

$$\begin{aligned} \text{minimize} \quad & \mathbf{Tr}\left[(I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)^{-1}\right] \\ \text{subject to} \quad & \sum_{j=1}^{n} p_{ij} = 1, \ \textit{for each } i \in \{1, \ldots, n\} \\ & \boldsymbol{\pi}_i p_{ij} = \boldsymbol{\pi}_j p_{ji}, \ \textit{for each } (i,j) \in E \\ & 0 \leq p_{ij} \leq 1, \ \textit{for each } (i,j) \in E \\ & p_{ij} = 0, \ \textit{for each } (i,j) \notin E. \end{aligned} \tag{3.3}$$

Remark 20 *All feasible solutions P to Problem 1 are inherently irreducible transition matrices: when P is not irreducible, the matrix $(I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)$ is not invertible. Moreover, a feasible point always exists since the Metropolis-Hastings algorithm applied to any irreducible transition matrix associated with $\mathcal{G}$, generates a reversible transition matrix which is irreducible and satisfies the stationary distribution constraint [47].*

The following theorem establishes the convexity of the Kemeny constant for transition matrices with fixed stationary distribution.

Theorem 21 (Convexity of Problem 1) *Let $\mathcal{P}_{\boldsymbol{\pi}}$ denote the set of matrices associated to irreducible reversible Markov chains with stationary distribution $\boldsymbol{\pi}$. Then, $\mathcal{P}_{\boldsymbol{\pi}}$ is a convex set and $P \mapsto K(P)$ is a convex function over $\mathcal{P}_{\boldsymbol{\pi}}$.*

Proof: Let $\mathcal{S}$ denote the set of symmetric positive definite matrices, for any stationary distribution $\boldsymbol{\pi} \in \mathbb{R}^n_{>0}$, denote the set $\mathcal{S}_{\mathcal{P},\boldsymbol{\pi}} := \{I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T \mid P \in \mathcal{P}_{\boldsymbol{\pi}}\}$. We begin by showing that $\mathcal{P}_{\boldsymbol{\pi}}$ is a convex set. Let $P_1, P_2 \in \mathcal{P}_{\boldsymbol{\pi}}$, then $\mathcal{P}_{\boldsymbol{\pi}}$ is convex if for an arbitrary $\theta \in [0,1]$ that

$$\theta P_1 + (1-\theta) P_2 = P_3 \in \mathcal{P}_{\boldsymbol{\pi}}. \tag{3.4}$$

Pre and post multiplying (3.4) by $\Pi^{1/2}$ and $\Pi^{-1/2}$, respectively, we have that $\theta \Pi^{1/2} P_1 \Pi^{-1/2} + (1-\theta)\Pi^{1/2} P_2 \Pi^{-1/2} = \Pi^{1/2} P_3 \Pi^{-1/2}$. Then $\Pi^{1/2} P_3 \Pi^{-1/2}$ is symmetric since $\Pi^{1/2} P_1 \Pi^{-1/2}$ and $\Pi^{1/2} P_2 \Pi^{-1/2}$ are symmetric. Pre multiplying (3.4) by $\boldsymbol{\pi}^T$ we easily verify that the stationary distribution P_3 is $\boldsymbol{\pi}^T$ and similarly, post multiplying by $\mathbb{1}_n$ verifies that P_3 is row stochastic. Finally taking the left hand side of (3.4) to the n-th power gives $(\theta P_1 + (1-\theta)P_2)^n = \theta^n P_1^n + (1-\theta)^n P_2^n + \zeta$, where ζ denotes the sum of all other terms in the expansion and has the property $\zeta_{ij} \geq 0$ for all i, j since P_1 and P_2 are non-negative element-wise matrices. Moreover from irreducibility, there exists a sufficiently large N such that for $n > N$, $(P_1^n)_{ij} > 0$ and $(P_2^n)_{ij} > 0$ for all i, j, which implies $(P_3^n)_{ij} > 0$, therefore $P_3 \in \mathcal{P}_{\boldsymbol{\pi}}$ and $\mathcal{P}_{\boldsymbol{\pi}}$ is convex.

Next we show that $\mathcal{S}_{\mathcal{P},\boldsymbol{\pi}} \subset \mathcal{S}$. From the proof of Theorem 19 we have for $P \in \mathcal{P}_{\boldsymbol{\pi}}$ that $I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T$ has eigenvalues $\{1, 1-\lambda_2, \ldots, 1-\lambda_n\}$, where λ_i for $i \in \{1, \ldots, n\}$ are the eigenvalues of P, where $\lambda_i \leq |1|$ for all i and $\lambda_i \neq 1$ for $i \in \{2, \ldots, n\}$. Therefore, all eigenvalues of $I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T$ are strictly greater than zero. Finally, since P is reversible $\Pi^{1/2} P \Pi^{-1/2}$ is symmetric implying $(I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)^T = I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T$ and so $\mathcal{S}_{\mathcal{P},\boldsymbol{\pi}} \subset \mathcal{S}$.

Finally, define the mapping $g : \mathcal{P}_{\boldsymbol{\pi}} \mapsto \mathcal{S}_{\mathcal{P},\boldsymbol{\pi}}$ by $g(X) = I - \Pi^{1/2} X \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T$. This is an affine mapping from the convex set $\mathcal{P}_{\boldsymbol{\pi}}$ to a subset of $\mathcal{S}$. From [41] we know that $\mathbf{Tr}[X^{-1}]$ is convex for $X \in \mathcal{S}$, therefore the composition with the affine mapping

$g : \mathcal{P}_{\boldsymbol{\pi}} \mapsto \mathcal{S}_{\mathcal{P},\boldsymbol{\pi}} \subset \mathcal{S}$, $\mathbf{Tr}[g(X)^{-1}]$ is also convex [19, Chapter 3.2.2]. ■

Problem 1 includes constraints on the stationary distribution of the transition matrix, a notion which has not been looked at in the literature before. The author of [58] provides bounds to determine the set of transition matrices such that K is minimized and [52] gives special matrices for which the optimal Kemeny constant can be found, but these are all approached for the general setting with no constraint on the actual stationary distribution. In real-world settings, constraints on the stationary distribution are important and have many practical interpretations. For example, it is often desirable to visit certain regions more frequently than other based on each region's relative importance.

3.3.2 SDP framework for optimizing the Kemeny constant

Here we show how Problem 1 can be equivalently rewritten as an SDP by introducing a symmetric slack matrix.

Problem 2 *(Optimizing the Kemeny constant of a reversible Markov chain (SDP)): Given the stationary distribution $\boldsymbol{\pi}$ and graph $\mathcal{G}$ with vertex set $\mathcal{V}$ and edge set E, determine $X = [x_{ij}]$ and the transition probabilities $P = [p_{ij}]$ solving:*

$$
\begin{aligned}
\text{minimize} \quad & \mathbf{Tr}[X] \\
\text{subject to} \quad & \begin{bmatrix} I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T & I \\ I & X \end{bmatrix} \succeq 0 \\
& \sum_{j=1}^{n} p_{ij} = 1, \ \textit{for each } i \in \{1, \ldots, n\} \\
& \boldsymbol{\pi}_i p_{ij} = \boldsymbol{\pi}_j p_{ji}, \ \textit{for each } (i,j) \in E \\
& 0 \leq p_{ij} \leq 1, \ \textit{for each } (i,j) \in E \\
& p_{ij} = 0, \ \textit{for each } (i,j) \notin E.
\end{aligned}
$$

The first inequality constraint in Problem 2 represents a linear matrix inequality (LMI) and denotes that the matrix is positive semidefinite. Since the matrix in the LMI has off-diagonal entries equal to the identity matrix, it holds true if and only if X is positive definite and $(I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T) - X^{-1}$ is positive semidefinite [3, Theorem 1]. This implies $(I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)$ is positive definite and that $X \succeq (I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)^{-1}$. Therefore, the SDP given by Problem 2 minimizes the Kemeny constant.

3.4 The weighted Kemeny constant and its minimization

In most practical applications, distance/time traveled and service costs/times are important factors in the model of the system. We incorporate these concepts by allowing for an additional set of weighted edges in our graph in addition to the edge weights which describe the transition probabilities. Such a system can be represented by the doubly-weighted graph $\mathcal{G} = (\mathcal{V}, E, P, D)$, where $D = [d_{ij}]$ is a weight matrix with the properties that: if $(i, i) \in E$, then $d_{ii} \geq 0$; if $(i, j) \in E$, $i \neq j$, then $d_{ij} > 0$; and if $(i, j) \notin E$, then $d_{ij} = 0$. The weighted adjacency matrix $P = [p_{ij}]$ has the same interpretation as before as an irreducible row-stochastic transition matrix P which governs the random walk on the graph. An example of a doubly-weighted graph is shown in Figure 3.1. In the following, we will interpret d_{ij}, $i \neq j$ as the time to travel between two nodes, i and j, in the graph and d_{ii} as the service time at node i. We discuss another motivating example and interpretation for d_{ij} in a later section.

Recall that $X_k = i$ denotes that the random walker is at node i at time k. If a sample trajectory of the random walk is $X_0 = i$, $X_1 = j$, $X_2 = k$, then the time instant at which a random walker arrives in state X_2 is $d_{ij} + d_{jk}$. Thus the time interval between two

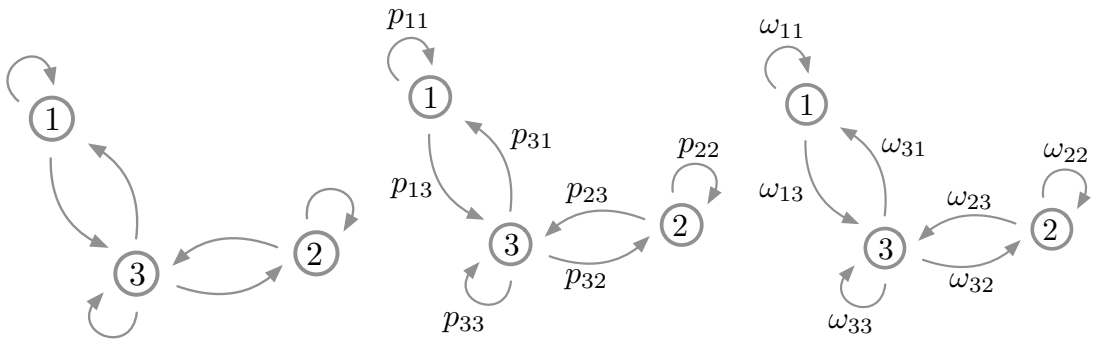

Figure 3.1: Example of a doubly-weighted graph $\mathcal{G} = (\mathcal{V}, E, P, D)$ with three nodes: (a) shows the edge set, E, allowed for the graph with three nodes, (b) shows the probabilities, p_{ij} to move along each edge, and (c) shows the time (i.e., distance traveled), d_{ij} to move along each edge.

consecutive steps of this random walk depends on the weighted adjacency matrix, D, of the graph and is not constant.

In the following analysis, we look at several characterization and optimization objectives: The first involves extending the notion of the Kemeny constant to doubly-weighted graphs and providing a detailed characterization of this extension. The second involves the minimization of the mean first passage time of a doubly-weighted graph and the third involves characterization and minimization of the mean time to execute a single *hop*. The first and second objectives are motivated by the need to minimize visit times to nodes in the graph, and the third is motivated by the desire to minimize resource consumption when moving between nodes. We seek to design transition matrices P with stationary distribution $\boldsymbol{\pi}$ which optimize each problem. We start with the first objective.

3.4.1 The mean first passage time for a doubly-weighted graph

The mean first passage time for the Markov chain on a weighted graph $\mathcal{G} = (\mathcal{V}, E, P)$ by definition, is simply its Kemeny constant. Recall that the mean first passage time for node i, defined by $\boldsymbol{k}_i$, is determined by taking the expectation over the first passage

times m_{ij}, from node i to all other nodes j. We present an analogous notion of the first passage time between two nodes on a doubly-weighted graph. We start with defining the first passage time random variable for a random walk on a doubly-weighted graph and provide a recursive formulation for its expectation.

As in Section 3.3.1, for any two nodes $i, j \in \{1, \ldots, n\}$, the *first passage time from i to j* is the first time that the random walker starting at node i at time 0 reaches node j, that is,

$$T_{ij} = \min\Big\{ \sum_{n=0}^{k-1} d_{X_n, X_{n+1}}, \text{ for } k \geq 1 \mid X_k = j \text{ given that } X_0 = i \Big\}.$$

Lemma 22 *(First passage time for a doubly-weighted graph): Let $n_{ij} = \mathbb{E}[T_{ij}]$ denote the mean first passage time to go from i to j for a graph with weight matrix D and transition matrix P. Then*

$$n_{ij} = p_{ij}(d_{ij}) + \sum_{k \neq j} p_{ik}(n_{kj} + d_{ik}), \tag{3.5}$$

or, in matrix notation,

$$(I - P)N = (P \circ D)\mathbb{1}_n \mathbb{1}_n^T - PN_{\mathrm{d}}, \tag{3.6}$$

where $(P \circ D)$ is the element-wise product between P and D and where $N_{\mathrm{d}} = [\delta_{ij} n_{ij}]$.

Proof: By its definition, the first passage time satisfies the recursive formula:

$$T_{ij} = \begin{cases} d_{ij}, & \text{with probability } p_{ij}, \\ d_{ik} + T_{kj}, & \text{with probability } p_{ik}, k \neq j. \end{cases} \tag{3.7}$$

Therefore, the results follows from taking the expectation:

$$\mathbb{E}[T_{ij}] = d_{ij}p_{ij} + \sum_{k \neq j} p_{ik}(\mathbb{E}[T_{kj}] + d_{ik}).$$

■

The matrix N, which we call the *mean first passage time matrix for a doubly-weighted graph* thus satisfies an equation similar to (3.1) for the passage time matrix M of a graph with a single weight matrix, the transition matrix P. In fact, we see that equation (3.6) is equivalent to (3.1) when $d_{ij} = 1$ for all $(i, j) \in E$ (i.e., for an unweighted graph).

The random variable tracking the time interval between consecutive visits to a node has been referred to as the *refresh time* of that node [76] and n_{ii} is the expected value of the refresh time for the random walk. We now obtain a relation between $\boldsymbol{\pi}$ and the refresh times n_{ii}.

Theorem 23 (Refresh times for doubly-weighted graphs) *Consider a Markov chain on a doubly-weighted graph $\mathcal{G} = (\mathcal{V}, E, P, D)$ with stationary distribution $\boldsymbol{\pi}$ and associated mean first passage time matrix N. The refresh time for node i is given by $n_{ii} = (\boldsymbol{\pi}^T(P \circ D)\mathbb{1}_n)/\boldsymbol{\pi}_i$, implying that*

$$N_{\mathrm{d}} = \boldsymbol{\pi}^T(P \circ D)\mathbb{1}_n M_{\mathrm{d}}. \tag{3.8}$$

Proof: The stationary distribution of the transition matrix P is $\boldsymbol{\pi} \in \mathbb{R}^{n \times 1}$. Therefore, premultiplying equation (3.6) by $\boldsymbol{\pi}^T$, we obtain

$$0 = \boldsymbol{\pi}^T(P \circ D)\mathbb{1}_n\mathbb{1}_n^T - \boldsymbol{\pi}^T N_{\mathrm{d}},$$

where the left hand side of equation (3.6) is zero since $\pi^T(I - P) = \pi^T - \pi^T = 0$. Now

we have that $(\boldsymbol{\pi}^T(P \circ W)\mathbb{1}_n)\mathbb{1}_n^T = \boldsymbol{\pi}^T N_{\mathrm{d}}$. Since N_d is a diagonal matrix and $\boldsymbol{\pi}^T(P \circ D)\mathbb{1}_n$ is a scalar, we get that $\boldsymbol{\pi}_i n_{ii} = \boldsymbol{\pi}^T(P \circ D)\mathbb{1}_n$. Thus, dividing by $\boldsymbol{\pi}_i$ we have that $n_{ii} = \boldsymbol{\pi}^T(P \circ D)\mathbb{1}_n/\boldsymbol{\pi}_i$, and in matrix form $N_d = \boldsymbol{\pi}^T(P \circ D)\mathbb{1}_n \mathrm{diag}(\{1/\boldsymbol{\pi}_1, \ldots, 1\ \boldsymbol{\pi}_n\}) = \boldsymbol{\pi}^T(P \circ D)\mathbb{1}_n M_d$. ■

This theorem implies that the refresh time n_{ii} of the random walk is directly proportional to the visit frequencies $1/\boldsymbol{\pi}_i$. Therefore, the relative visit frequencies of one node compared to another are not a function of the weight matrix D and only depend on the stationary distribution of the transition matrix P.

We now investigate the properties of the mean first passage time of the weighted random walk. The mean first passage time for a doubly-weighted graph $\mathcal{G} = (\mathcal{V}, E, P, D)$ with associated passage times matrix N is given by $K_D = \boldsymbol{\pi}^T \boldsymbol{k}_D$, where $\boldsymbol{k}_D = N\boldsymbol{\pi}$ is the vector of *first passage times* and the $i-$th entry $\boldsymbol{k}_{D,i}$ in $\boldsymbol{k}_D$ denotes the mean time to go from i to any other node. We refer to the mean first passage time for a doubly-weighted graph, K_D, as the *weighted Kemeny constant*. We now provide an analytic expression for the vector $\boldsymbol{k}_D \in \mathbb{R}^{n\times 1}$.

Lemma 24 *(First passage times for a doubly-weighted graph): Given a Markov chain on a doubly-weighted graph $\mathcal{G} = (\mathcal{V}, E, P, D)$ with stationary distribution $\boldsymbol{\pi}$ and associated first passage time matrix N, the following equality holds:*

$$(I - P)\boldsymbol{k}_D = (P \circ D)\mathbb{1}_n - \mathbb{1}_n \boldsymbol{\pi}^T (P \circ D)\mathbb{1}_n, \tag{3.9}$$

where $\boldsymbol{k}_D = N\boldsymbol{\pi}$.

Proof: Post multiplying equation (3.6) on both sides by $\boldsymbol{\pi}$ gives

$$\begin{aligned}(I-P)N\boldsymbol{\pi} =&(P\circ D)\mathbb{1}_n\mathbb{1}_n^T\boldsymbol{\pi} - PN_{\mathrm{d}}\boldsymbol{\pi},\\ (I-P)\boldsymbol{k}_D =&(P\circ D)\mathbb{1}_n - P(\boldsymbol{\pi}^T(P\circ D)\mathbb{1}_n)\mathbb{1}_n\\ =&(P\circ D)\mathbb{1}_n - \mathbb{1}_n\boldsymbol{\pi}^T(P\circ D)\mathbb{1}_n.\end{aligned}$$

■

The right hand side of (3.9) gives the insight that, in general, $\boldsymbol{k}_{D,i} \neq \boldsymbol{k}_{D,j}$ on the doubly-weighted graph, unlike the counterpart for the single-weighted graph (where $\boldsymbol{k}_i = K$ for all $i \in \{1,\ldots,n\}$). Interestingly enough however, there does exist a relation between the weighted Kemeny constant K_D and the Kemeny constant K as is stated in the following theorem, whose proof is postponed to the Appendix.

Theorem 25 *(Weighted Kemeny constant of a Markov chain): For the doubly-weighted graph $\mathcal{G} = (\mathcal{V}, E, P, D)$, the weighted Kemeny constant K_D of the Markov chain is given by*

$$K_D = \boldsymbol{\pi}^T(P\circ D)\mathbb{1}_n K, \tag{3.10}$$

where K is the Kemeny constant associated with the irreducible transition matrix P with stationary distribution $\boldsymbol{\pi}$.

Remark 26 *The expected number of hops to go from one node to another in a Markov chain with transition matrix P is its Kemeny constant. The expected distance travelled (and hence time taken) executing one hop is $\sum_i \boldsymbol{\pi}_i \sum_j p_{ij}d_{ij} = \boldsymbol{\pi}(P\circ D)\mathbb{1}_n$. Hence, it is consistent with intuition that the expected time to travel from one node to another should be $K\boldsymbol{\pi}^T(P\circ D)\mathbb{1}_n$ as is formally shown in the appendix.*

Given the above results, we are now ready to state another problem of interest.

Problem 3 *(Optimizing the weighted Kemeny constant of a reversible Markov chain): Given the stationary distribution $\boldsymbol{\pi}$ and graph $\mathcal{G}$ with vertex set $\mathcal{V}$, edge set E and weight matrix D, determine the transition probabilities $P = [p_{ij}]$ solving:*

$$\begin{aligned} \text{minimize} \quad & \left(\boldsymbol{\pi}^T (P \circ D)\mathbb{1}_n\right)\left(\mathbf{Tr}\left[(I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)^{-1}\right]\right) \\ \text{subject to} \quad & \sum_{j=1}^{n} p_{ij} = 1, \ \textit{for each } i \in \{1, \ldots, n\} \\ & \boldsymbol{\pi}_i p_{ij} = \boldsymbol{\pi}_j p_{ji}, \ \textit{for each } (i,j) \in E \\ & 0 \leq p_{ij} \leq 1, \ \textit{for each } (i,j) \in E \\ & p_{ij} = 0, \ \textit{for each } (i,j) \notin E. \end{aligned}$$

The following theorem establishes the convexity of the weighted Kemeny constant for transition matrices with fixed stationary distribution.

Theorem 27 (Convexity of Problem 3) *Given the graph $\mathcal{G}$ with vertex set $\mathcal{V}$, edge set E and weight matrix D, let $\mathcal{P}_{\mathcal{G},\boldsymbol{\pi}}$ denote the set of matrices associated with $\mathcal{G}$ that are irreducible reversible Markov chains with stationary distribution $\boldsymbol{\pi}$. Then, $\mathcal{P}_{\mathcal{G},\boldsymbol{\pi}}$ is a convex set and $P \mapsto \boldsymbol{\pi}^T (P \circ D)\mathbb{1}_n K(P)$ is a convex function over $\mathcal{P}_{\mathcal{G},\boldsymbol{\pi}}$.*

Proof: Let $\mathcal{S}$ denote the set of symmetric positive definite matrices, for any stationary distribution $\boldsymbol{\pi} \in \mathbb{R}^n_{>0}$, denote the set $\mathcal{S}_{\mathcal{G},\mathcal{P},\boldsymbol{\pi}} := \{I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T \mid P \in \mathcal{P}_{\mathcal{G},\boldsymbol{\pi}}\}$. The proof of convexity of the set $\mathcal{P}_{\mathcal{G},\boldsymbol{\pi}}$ is similar to that of the proof of $\mathcal{P}_{\boldsymbol{\pi}}$ in Theorem 21 and so is omitted for brevity. Then from the proof of Theorem 21 we know there exists an affine mapping $g(X) : \mathcal{P}_{\mathcal{G},\boldsymbol{\pi}} \mapsto \mathcal{S}_{\mathcal{G},\mathcal{P},\boldsymbol{\pi}}$ given by $g(X) = I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T$. We know from [41] that $f(X) = \mathbf{Tr}[X^{-1}]$ is convex, therefore the perspective function $h(X,t) = \{tf(X/t) \mid t > 0\}$ is also convex [19, Chapter 3.2.6]. Moreover the composition of $h(X,t)$

with the affine mapping $g(X)$, $h(g(X),t)$ is also convex. Let $t = (\boldsymbol{\pi}^T(X \circ D)\mathbb{1}_n)^{1/2}$, and notice that $\boldsymbol{\pi}^T(X \circ D)\mathbb{1}_n > 0$ for $X \in \mathcal{P}_{\mathcal{G},\boldsymbol{\pi}}$ and therefore $t > 0$. Also notice that for a constant $k \in \mathbb{R}^n_{>0}$ and matrix $X \in \mathbb{R}^{n\times n}$ that $\mathbf{Tr}[(\frac{X}{k})^{-1}] = k\mathbf{Tr}[X^{-1}]$. Then $h(g(X),t) = t\mathbf{Tr}[(\frac{g(X)}{t})^{-1}] = t^2\mathbf{Tr}[(g(X)^{-1}] = \boldsymbol{\pi}^T(X\circ D)\mathbb{1}_n\mathbf{Tr}[(I-\Pi^{1/2}X\Pi^{-1/2}+\boldsymbol{q}\boldsymbol{q}^T)^{-1}]$ for $X \in \mathcal{P}_{\mathcal{G},\boldsymbol{\pi}}$. ■

3.4.2 SDP framework for optimizing the weighted Kemeny constant

In a similar fashion to Problem 1, we can formulate Problem 3 as an SDP by introducing the symmetric slack matrix $X \in \mathbb{R}^{n\times n}$ and the scalar variable t as is shown in the following.

Problem 4 *(Optimizing the weighted Kemeny constant of a reversible Markov chain (SDP)): Given the stationary distribution $\boldsymbol{\pi}$ and graph $\mathcal{G}$ with vertex set $\mathcal{V}$, edge set E*

and weight matrix D, determine $Y = [y_{ij}]$, X and t solving:

$$\begin{aligned}
\text{minimize} \quad & \mathbf{Tr}[X] \\
\text{subject to} \quad & \begin{bmatrix} t(I + \boldsymbol{q}\boldsymbol{q}^T) - \Pi^{1/2} Y \Pi^{-1/2} & I \\ I & X \end{bmatrix} \succeq 0 \\
& \sum_{j=1}^{n} y_{ij} = t, \ \textit{for each } i \in \{1, \ldots, n\} \\
& \boldsymbol{\pi}_i y_{ij} = \boldsymbol{\pi}_j y_{ji}, \ \textit{for each } (i,j) \in E \\
& 0 \leq y_{ij} \leq t, \ \textit{for each } (i,j) \in E \\
& y_{ij} = 0, \ \textit{for each } (i,j) \notin E \\
& \boldsymbol{\pi}^T (Y \circ D) \mathbb{1}_n = 1 \\
& t \geq 0.
\end{aligned}$$

Then, the transition matrix P is given by $P = Y/t$.

As in Problem 2, the first inequality constraint in Problem 4 represents an LMI. Before noting when the LMI holds, first note that the constraints in Problem 4 imply that $Pt = Y$ and that $t = \frac{1}{\boldsymbol{\pi}^T (P \circ D) \mathbb{1}_n}$. Hence, using a similar argument as in Problem 2, the LMI constraint holds true if and only if $X \succeq \boldsymbol{\pi}^T (P \circ D) \mathbb{1}_n (I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)^{-1}$ where X and $\pi^T (P \circ D) \mathbb{1}_n (I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)^{-1}$ are both positive definite, and therefore the SDP given by Problem 4 minimizes the weighted Kemeny constant.

3.4.3 Minimizing single hop distance

We now look at the objective of minimizing the mean time for a single *hop* of a random walk. At time k, let S_{ij} be the time required to transition from i to j in a single

hop along an edge of length d_{ij}. Then,

$$\begin{aligned}\mathbb{E}[S] = \sum_{i=1}^{n}\sum_{j=1}^{n} p_{ij}S_{ij} &= \sum_{i=1}^{n}\mathbb{P}[X_k = i]\sum_{j=1}^{n} d_{ij}\mathbb{P}[X_{k+1} = j] \\ &= \sum_{i=1}^{n}\sum_{j=1}^{n} \boldsymbol{\pi}_i d_{ij} p_{ij} = \boldsymbol{\pi}^T (P \circ D)\mathbb{1}_n. \end{aligned} \tag{3.11}$$

The function $\boldsymbol{\pi}^T(P \circ D)\mathbb{1}_n$ is clearly convex in P. If one assumes that $d_{ii} = 0$ for all $i \in \{1, \ldots, n\}$, then minimizing (3.11) over P yields the trivial solution $P = I$. We can take into account both the single hop distance as well as the Kemeny constant to design a Markov chain as follows.

Problem 5 *(Optimizing Kemeny constant and mean distance): Given the stationary distribution $\boldsymbol{\pi}$ and graph $\mathcal{G}$ with vertex set $\mathcal{V}$, edge set E and weight matrix D, and given user specified constant $\alpha \in [0, 1]$, determine the transition probabilities $P = [p_{ij}]$ solving:*

$$\begin{aligned} \text{minimize} \quad & \alpha \mathbf{Tr}\left[(I - \Pi^{1/2} P \Pi^{-1/2} + \boldsymbol{q}\boldsymbol{q}^T)^{-1}\right] \\ & + (1-\alpha)\boldsymbol{\pi}^T (P \circ D)\mathbb{1}_n \\ \text{subject to} \quad & \sum_{j=1}^{n} p_{ij} = 1, \ \textit{for each } i \in \{1, \ldots, n\} \\ & \boldsymbol{\pi}_i p_{ij} = \boldsymbol{\pi}_j p_{ji}, \ \textit{for each } (i,j) \in E \\ & 0 \leq p_{ij} \leq 1, \ \textit{for each } (i,j) \in E \\ & p_{ij} = 0, \ \textit{for each } (i,j) \notin E. \end{aligned}$$

This problem is convex since the sum of two convex problems is convex, moreover, it can be extended to an SDP utilizing the LMI defined in Problem 2. In the context where $d_{ii} = 0$ for all $i \in \{1, \ldots, n\}$, varying the parameter α can be used to control the connectivity of the Markov chain; the choice $\alpha = 1$ ensures connectivity, and the choice $\alpha = 0$ minimizes the single hop distance while making the graph disconnected.

3.5 Applications of the mean first passage time to surveillance

The results on mean first passage time for doubly-weighted graphs (i.e., the weighted Kemeny constant) presented in this work provide a general framework which can potentially be applied to the analysis and design in a myriad of fields. We focus on one application in particular; the intruder detection and surveillance problem. We look at two variations of this problem:

Scenario I In practical stochastic intruder detection and surveillance scenarios, there is often a desire to surveil some regions more than others (i.e.,have a pre- specified stationary distribution) while simultaneously minimizing the time any one region has to wait before it is serviced. For this setup, in every step of the random walk, the agent must move to a new region and execute its surveillance task. Assuming we are working on a doubly-weighted graph described by $\mathcal{G} = (\mathcal{V}, E, P, D)$, let us also assume there is a fixed persistent intruder in the environment and it takes s_i time for an agent to determine if the intruder is in region $i \in \mathcal{V}$. Denote the time to move from region i to region j by d_{ij}, where we can assume, without loss of generality, that $d_{ii} = 0$. Then, we can define the weight corresponding to the edge from i to j as $d_{ij} = d_{ij} + s_j$. In this scenario we wish to minimize the expected time to capture the persistent intruder when no prior knowledge of their position is known.

Scenario II In this scenario we consider the intruder detection problem and adopt a similar setup to Scenario I, however, we now assume a set of intruders are distributed throughout the environment. Each intruder performs a malicious activity in its host region for a fixed duration of time, which we call the *intruder life-time*, followed instan-

taneously by another intruder. The intruder is caught only if the agent is in the same region as the intruder for any duration of the *intruder life-time*. For simplicity only a single intruder appears at a time.

In the following subsections we analyze the performance of various stochastic surveillance policies as applied to Scenario I and Scenario II described above. More specifically, we gauge the performance of other well-known Markov chain design algorithms against the algorithms presented in this chapter.

3.5.1 Optimal strategy for Scenario I

In the context of Scenario I the weighted Kemeny constant of the agent's transition matrix, P, corresponds to the average time it takes to capture an intruder regardless of where the agent and intruder are in the environment. Therefore by definition of the Kemeny constant, we have the following immediate corollary for Scenario I.

Corollary 28 (Optimal surveillance and service strategy) *The transition matrix P which has minimal mean first passage time is the optimal strategy for the intruder detection problem described by* Scenario I.

This tells us that if we restrict ourselves to reversible Markov chains, then not only is the chain with minimal mean first passage time optimal, but given the results from Section 3.3 and 3.4, we can also optimally design this chain.

3.5.2 Numerical analysis of Scenario II

In Scenario II the transition matrix with minimum mean first passage time is not guaranteed to be the optimal policy, and thus to gauge its performance compared to other policies we analyze both homogeneous (uniform service/travel times) and heterogeneous environment cases. To compare performance we generate a random walk for the

environment using the Metropolis-Hastings, fastest mixing Markov chain (FMMC) [18], and Kemeny constant algorithms. While game theoretic frameworks [27, 17] also generate stochastic policies, they are based on assumptions on the intruder behavior. We avoid such assumptions here and, therefore, omit them from our comparative analysis.

We first look at the homogeneous case which is described by the discretized environment shown in Figure 3.2. We assume that a single surveillance agent executes a random walk in the environment, spending 1 time unit in each region, and that the agent transitions between two connected regions instantaneously. Also, we assume a uniform stationary distribution on the transition matrix (each node in the region must be visited with equal likelihood). The Markov chain generated by the Metropolis-Hastings algorithm is generated by applying the algorithm to the random walk described by $p_{ij} = 1/d_i$ for $i \neq j$ and $p_{ij} = 0$ for $i = j$, where d_i is the degree of a node i (excluding self-loops) [47]. The *intruder life-time* is set to 66 time units and 500 intruders appear per simulation run (the sequence in which the intruders appear is determined before each simulation run), for a total simulation run of $33,000$ time units. As stated in the scenario description, the intruder is caught if the surveillance agent is in the same region as the intruder for any portion of the *intruder life-time*. Table 3.1 summarizes the statistical performance of each algorithm after 200 runs of the simulation and justifies our use of the Kemeny constant algorithm as a valid surveillance strategy; the Kemeny constant algorithm captures intruders more frequently than the other two algorithms, and its worst case performance is still better than the worst case performance of the other two algorithms. Although results for an intruder life-time of only 66 time units are presented here, we have found that the Kemeny constant algorithm always outperforms the other two algorithms or is equivalent; the algorithms become equivalent in the limiting case, when the intruder life-times are so low that no intruder can be caught, or when the intruder-life times are so large that the intruder is always caught.

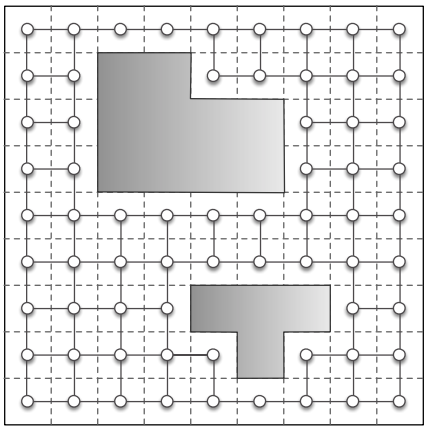

Figure 3.2: Environment with two obstacle represented by an unweighted graph.

Algorithm	**Min**	**Mean**	**Max**	**StdDev**	K
Kemeny constant	26.6	32.4	38.2	2.1	207
FMMC	24.6	29.8	34.4	1.9	236
Metropolis-Hastings	24.8	31.1	37	2.1	231

Table 3.1: Statistics on the percentage of intruders caught in 200 simulation runs for the environment in Fig. 3.2.

For the heterogeneous case, we work with the environment shown in Figure 3.3. In this environment the time taken by the agent to travel an edge is no longer instantaneous and has travel weights as shown in the figure. Once in a new region, the agent is required to spend 1 time unit examining the region for malicious activities. We again assume that each node in the region must be visited with equal likelihood. We again also assume an intruder is caught if the surveillance agent is in the same region as a intruder for any portion of the *intruder life-time*, but now set the *intruder life-time* to 11 time units with a intruder appearing 500 times (total of 5500 time units per simulation run). Since the design of the FMMC and Metropolis-Hastings algorithms do not inherently account for non-uniform travel and service times, we also compare the performance of the random walk generated by the weighted Kemeny constant algorithm against the random walk generated by solving Problem 5 with $\alpha = 0.1$ (smaller α emphasizes minimizing the length of

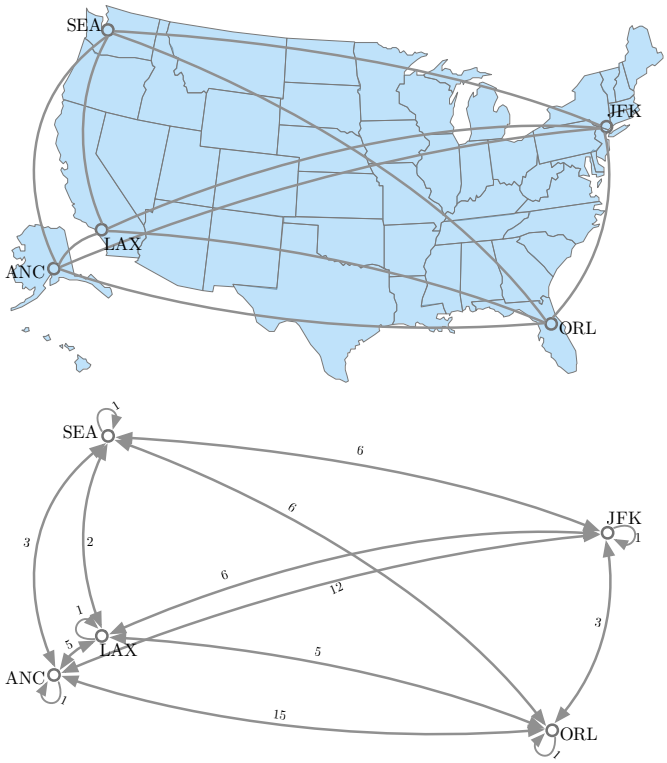

Figure 3.3: Various airport hub locations (top), and the corresponding weight map (bottom). Edge weights between two hubs account for travel time between hubs plus required service time once at hub. Self loops have no travel time so encompass only service time required at hub.

the average edge traveled in the graph). Table 3.2 summarizes the statistical performance of each algorithm after 200 runs of the simulation. The weighted Kemeny constant algorithm's performance compared to the FMMC and Metropolis-Hastings stochastic policies in this scenario is significantly better than what was seen in the first scenario. We also note that the random walk policy determined by solving Problem 5 performs comparably to the weighted Kemeny constant policy. This is to be expected since the Metropolis-Hastings and FMMC stochastic policies do not account for heterogeneous travel/service times on the graph. To get a better understanding of each algorithm's performance in this intruder scenario, the simulation is run for different intruder life-times, the results of which can be seen in Figure 3.4. There are several key items worth noting from the

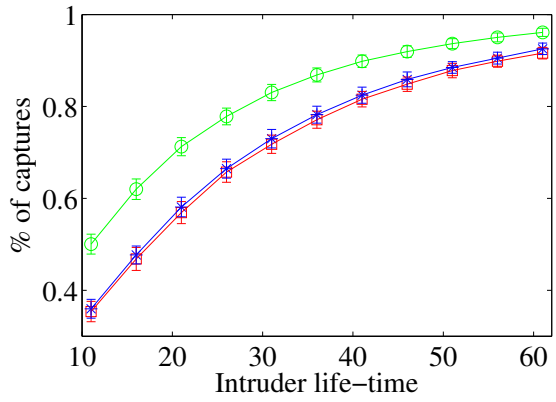

Figure 3.4: Percentage of intruders detected for varying intruder life-times by a surveillance agent executing a random walk according to the Markov chain generated by the mean first-passage time algorithm (circle), FMMC algorithm (square), M-H algorithm (asterisk), and the Markov chain generated by solving Problem 5 (diamond). Average points and standard deviation error bars are taken over 200 runs, where the intruder appears 500 times for each run.

Algorithm	**Min**	**Mean**	**Max**	**StdDev**	K_D
Weighted Kemeny	44	50.1	56	2.2	19.5
Kemeny+Mean Dist.	40.6	47.1	53	2.2	23.1
FMMC	29.8	35.4	40.4	2.2	26.2
Metropolis-Hastings	30.4	36	41.6	2.1	26.5

Table 3.2: Statistics on the percentage of intruders caught in 200 simulation runs for the environment in Fig. 3.3.

simulation. First, we see that the weighted Kemeny constant algorithm significantly outperforms the other algorithms for a large range of intruder life-times. This matches our intuition since the algorithm inherently minimizes average travel time between nodes. Second, notice that the Markov chain generated by solving Problem 5 (with $\alpha = 0.1$) performs well for small intruder life-times but its performance plateaus quickly. This is due to the fact that the transition matrix generated by solving Problem 5 forces agents to stay at a given node rather than jump nodes; as one would suspect, once intruder life-times increase, a strategy which places emphasis on an agent that moves between regions will begin to perform relatively better. Finally, observe that as intruder life-time

increases, the algorithms' capture rates start to converge. As in the homogeneous case, this is due to the fact that once the intruder's life-time is long enough, the agent will almost surely reach the intruder regardless of the policy it follows.

To solve for the Markov chains with minimal Kemeny constant (Problem 2 and Problem 4) and with fastest mixing rate, we use CVX, a Matlab-based package for convex programs [45]. The execution time to solve each Markov chain for the examples described above takes on the order of a couple seconds using a computer with a 1.3 GHz processor.

Appendix

Proof of Theorem 25: Let $\beta = \boldsymbol{\pi}^T(P \circ D)\mathbb{1}_n$, then from Theorem 23 we have that N_{d} from (3.6) can be written as βM_{d}. Now from Theorem 30 the general solution to (3.6) is

$$N = G((P \circ D)\mathbb{1}_n\mathbb{1}_n^T - \beta P M_{\mathrm{d}}) + (I - G(I - P))U, \tag{3.12}$$

where G is a generalized inverse of $(I-P)$ (see Theorem 32) and U is an arbitrary matrix as long as the consistency condition

$$\big(I - (I - P)G\big)\big((P \circ D)\mathbb{1}_n\mathbb{1}_n^T - \beta P M_{\mathrm{d}}\big) = 0 \tag{3.13}$$

holds. Substituting (3.18) from Lemma 33 in for $(I - P)G$ in (3.13) gives that

$$\begin{aligned}
&\big(I - (I - P)G\big)\big((P \circ D)\mathbb{1}_n\mathbb{1}_n^T - \beta P M_{\mathrm{d}}\big) \\
&\qquad = \frac{\boldsymbol{t}\boldsymbol{\pi}^T}{\boldsymbol{\pi}^T\boldsymbol{t}}((P \circ D)\mathbb{1}_n\mathbb{1}_n^T - \beta P M_{\mathrm{d}}), \\
&\qquad = \frac{\boldsymbol{t}}{\boldsymbol{\pi}^T\boldsymbol{t}}(\boldsymbol{\pi}^T(P \circ D)\mathbb{1}_n\mathbb{1}_n^T - \beta\boldsymbol{\pi}^T P M_{\mathrm{d}}), \\
&\qquad = \frac{\boldsymbol{t}}{\boldsymbol{\pi}^T\boldsymbol{t}}(\beta\mathbb{1}_n^T - \beta\mathbb{1}_n^T) = 0,
\end{aligned}$$

and so we have that the system of equations is consistent. This implies we can design U to reduce (3.12). We start by seeing how the second term in (3.12) can be reduced. Using (3.19) from Lemma 33 we have that $(I - G(I - P))U = \frac{\mathbb{1}_n \boldsymbol{u}^T}{\boldsymbol{u}^T \mathbb{1}_n} U = \mathbb{1}_n \boldsymbol{h}^T$, where $\boldsymbol{h}^T = \frac{\boldsymbol{u}^T U}{\boldsymbol{u}^T \mathbb{1}_n}$. Hence, we can re-write (3.12) as

$$N = G((P \circ D)\mathbb{1}_n \mathbb{1}_n^T - \beta P M_{\mathrm{d}}) + \mathbb{1}_n \boldsymbol{h}^T, \tag{3.14}$$

designing U reduces to designing the n elements of $\boldsymbol{h}$. Let $H = \mathrm{diag}[\boldsymbol{h}]$, then $\mathbb{1}_n \boldsymbol{h}^T = \mathbb{1}_n \mathbb{1}_n^T H$. Also, let $\Xi = \mathbb{1}_n \boldsymbol{\pi}^T$, where $\Xi_{\mathrm{d}} = \mathrm{diag}[\boldsymbol{\pi}]$. Utilizing these expressions in (3.14) and taking the diagonal elements gives

$$\begin{aligned}
&\big(N = G((P \circ D)\mathbb{1}_n \mathbb{1}_n^T - \beta P M_{\mathrm{d}}) + \mathbb{1}_n \mathbb{1}_n^T H\big)_{\mathrm{d}}, \\
\implies &\beta M_{\mathrm{d}} = (G(P \circ D)\Xi)_{\mathrm{d}} M_{\mathrm{d}} - \beta (GP)_{\mathrm{d}} M_{\mathrm{d}} + H, \\
\implies &H = \beta M_{\mathrm{d}} - (G(P \circ D)\Xi)_{\mathrm{d}} M_{\mathrm{d}} + \beta (GP)_{\mathrm{d}} M_{\mathrm{d}},
\end{aligned}$$

where we use Lemma 31 to get the initial diagonal reduction. Substituting the expression for H into (3.14), and recalling that $\mathbb{1}_n \boldsymbol{h}^T = \mathbb{1}_n \mathbb{1}_n^T H$ gives

$$N = \big(G(P \circ D)\Xi - \mathbb{1}_n \mathbb{1}_n^T (G(P \circ D)\Xi)_{\mathrm{d}} + \beta(\mathbb{1}_n \mathbb{1}_n^T (GP)_{\mathrm{d}} - GP + \mathbb{1}_n \mathbb{1}_n^T)\big) M_{\mathrm{d}}, \tag{3.15}$$

where we use the fact that $\mathbb{1}_n \mathbb{1}_n^T = \Xi M_{\mathrm{d}}$. Now from (3.19) we have that $I - G - GP = \frac{\mathbb{1}_n \boldsymbol{u}^T}{\boldsymbol{u}^T \mathbb{1}_n}$. Notice that $\mathbb{1}_n \mathbb{1}_n^T (I - G - GP)_{\mathrm{d}} = \mathbb{1}_n \mathbb{1}_n^T (\frac{\mathbb{1}_n \boldsymbol{u}^T}{\boldsymbol{u}^T \mathbb{1}_n})_{\mathrm{d}} = \frac{\mathbb{1}_n \boldsymbol{u}^T}{\boldsymbol{u}^T \mathbb{1}_n}$ and so this implies that $\mathbb{1}_n \mathbb{1}_n^T - \mathbb{1}_n \mathbb{1}_n^T G_{\mathrm{d}} + \mathbb{1}_n \mathbb{1}_n^T (GP)_{\mathrm{d}} = I - G + GP$, which implies that $\mathbb{1}_n \mathbb{1}_n^T + \mathbb{1}_n \mathbb{1}_n^T (GP)_{\mathrm{d}} - GP = I - G + \mathbb{1}_n \mathbb{1}_n^T G_{\mathrm{d}}$. Substituting this into (3.15) gives the following reduced form.

$$N = \big(G(P \circ D)\Xi - \mathbb{1}_n \mathbb{1}_n^T (G(P \circ D)\Xi)_{\mathrm{d}} + \beta(\mathbb{1}_n \mathbb{1}_n^T G_{\mathrm{d}} + I - G)\big) M_{\mathrm{d}}. \tag{3.16}$$

Observing the definition of the generalized inverse, G, given by Theorem 15 part (ii) and recalling that $\Xi = \mathbb{1}_n \boldsymbol{\pi}^T$, we can rewrite the first term on the right hand side of (16) as $G(P \circ D)\Xi = (I - P + \boldsymbol{t}\boldsymbol{u}^T)^{-1}(P \circ D)\mathbb{1}_n \boldsymbol{\pi}^T$. Substituting (3.20) in for the right hand side with $\boldsymbol{t} = (P \circ D)\mathbb{1}_n$ gives $G(P \circ D)\Xi = \frac{\mathbb{1}_n \boldsymbol{\pi}^T}{\boldsymbol{u}^T \mathbb{1}_n} = \frac{1}{\boldsymbol{u}^T \mathbb{1}_n}\Xi$, and so $\mathbb{1}_n \mathbb{1}_n^T (G(P \circ D)\Xi)_{\mathrm{d}} = \mathbb{1}_n \mathbb{1}_n^T (\frac{1}{\boldsymbol{u}^T \mathbb{1}_n}\Xi)_{\mathrm{d}} = \frac{1}{\boldsymbol{u}^T \mathbb{1}_n}\Xi = G(P \circ D)\Xi$. Therefore, the first two terms in (3.16) cancel giving the equality

$$N = \beta(\mathbb{1}_n \mathbb{1}_n^T G_{\mathrm{d}} + I - G)M_{\mathrm{d}}. \tag{3.17}$$

We have already defined $\boldsymbol{t}$ in the generalized inverse G but not $\boldsymbol{u}$. Let $\boldsymbol{u} = \boldsymbol{\pi}$ and multiply the right hand side of (3.17) by $\boldsymbol{\pi}$ and the left hand side by $\boldsymbol{\pi}^T$. Utilizing equality (3.21) from Lemma 33 gives

$$\begin{aligned}\boldsymbol{\pi}^T N \boldsymbol{\pi} &= \boldsymbol{\pi}^T \beta(\mathbb{1}_n \mathbb{1}_n^T G_{\mathrm{d}} + I - G)M_{\mathrm{d}}\boldsymbol{\pi} = \beta(\mathbb{1}_n^T G_{\mathrm{d}} + \boldsymbol{\pi}^T - \frac{\boldsymbol{\pi}^T}{\beta})\mathbb{1}_n \\ &= \beta(\mathbb{1}_n^T G_{\mathrm{d}} \mathbb{1}_n + 1 - \frac{1}{\beta}) = \beta(\mathbf{Tr}[G] + 1) - 1.\end{aligned}$$

Noting that the eigenvalue at 1 for an irreducible row-stochastic matrix is unique, it can be easily verified using the orthogonality property of left and right eigenvectors that the eigenvalues of G^{-1} are $\bar{\lambda}_i = (1 - \lambda_i)$ for $i \in \{2, \ldots, n\}$, where λ_i are eigenvalues of P and $\lambda_i \neq 1$. Therefore, it only remains to find $\bar{\lambda}_1$. Taking the trace of G^{-1} gives $\mathbf{Tr}[I - P + (P \circ D)\mathbb{1}_n \boldsymbol{\pi}^T] = \mathbf{Tr}[I - P] + \mathbf{Tr}[(P \circ D)\mathbb{1}_n \boldsymbol{\pi}^T] = \sum_{i=1}^n (1 - \lambda_i) + \boldsymbol{\pi}^T (P \circ D)\mathbb{1}_n$, which implies that $\bar{\lambda}_1 = \boldsymbol{\pi}^T (P \circ D)\mathbb{1}_n = \beta$. Therefore, $\beta(\mathbf{Tr}[G] + 1) - 1 = \beta(\frac{1}{\beta} + \sum_{i=2}^n \frac{1}{1-\lambda_i} + 1) - 1 = \beta(1 + \sum_{i=2}^n \frac{1}{1-\lambda_i})$.

Supplemental Material

For completeness, we include the following results which are needed in the proof of Theorem 25. We begin with some standard results on generalized inverses. For more details refer to [50, Chapter 4] or [49] .

Definition 29 (Generalized inverse) *A* generalized inverse *of an $m \times n$ matrix A is defined as any $n \times m$ matrix A^- that has the property*

$$AA^-A = A.$$

It should be noted that a generalized inverse always exists, although it is not always unique. However, for non-singular matrices the generalized inverse is unique and corresponds to the usual notion of a matrix inverse. The following theorems summarize practical considerations when working with generalized inverses.

Theorem 30 *The equation $A\boldsymbol{x} = \boldsymbol{b}$ admits a solution if and only if every generalized inverse A^- satisfies*

$$AA^-\boldsymbol{b} = \boldsymbol{b}.$$

Then, we say $A\boldsymbol{x} = \boldsymbol{b}$ is consistent *and all its general solutions are given by*

$$\boldsymbol{x} = A^-\boldsymbol{b} + (A^-A - I)\boldsymbol{z},$$

where $\boldsymbol{z}$ is an arbitrary vector. Moreover, a necessary and sufficient condition for the equation $AX = C$ to be consistent is that $(I - AA^-)C = 0$, in which case the general

solution is given by

$$X = A^{-}C + (I - A^{-}A)U,$$

where U is an arbitrary matrix.

The next two results come from [51, Chapter 7].

Lemma 31 (Diagonal matrix properties) *For $\boldsymbol{\pi}$ with positive non-zero elements, let $\mathbb{1}_n\boldsymbol{\pi}^T = \Xi$, where $\Xi_{\mathrm{d}} = diag[\boldsymbol{\pi}]$. Also, let Λ be any diagonal matrix, X any square matrix of same dimensions as Λ, and $D = (\Xi_{\mathrm{d}})^{-1}$, then*

(i.) $(X\Lambda)_{\mathrm{d}} = (X_{\mathrm{d}})\Lambda$, *and*

(ii.) $(X\mathbb{1}_n\mathbb{1}_n^T)_{\mathrm{d}} = (X\Xi)_{\mathrm{d}}D$, *and*

(iii.) $\mathbb{1}_n\mathbb{1}_n^T\Xi_{\mathrm{d}} = \Xi$.

Theorem 32 (Generalized inverse of $I - P$) *Let $P \in \mathbb{R}^{n\times n}$ be the transition matrix of a irreducible Markov Chain with stationary distribution $\boldsymbol{\pi}$. Let $\boldsymbol{u}, \boldsymbol{t} \in \mathbb{R}^n$ be any vectors such that $\boldsymbol{u}^T\mathbb{1}_n \neq 0$ and $\boldsymbol{\pi}^T\boldsymbol{t} \neq 0$, then*

(i.) $I - P + \boldsymbol{t}\boldsymbol{u}^T$ *is nonsingular, and*

(ii.) $(I - P + \boldsymbol{t}\boldsymbol{u}^T)^{-1}$ *is a generalized inverse of $I - P$.*

Lemma 33 (Properties of the generalized inverse of $I - P$) *Let $\mathcal{G} = (\mathcal{V}, E, P, D)$ be a doubly-weighted graph with associated weight matrix D and irreducible transition matrix P with stationary distribution $\boldsymbol{\pi}$. Also let $G = (I - P + \boldsymbol{t}\boldsymbol{u}^T)^{-1}$ denote the generalized inverse of $(I - P)$, then the following relations hold.*

$$(I - P)G = I - \frac{\boldsymbol{t}\boldsymbol{\pi}^T}{\boldsymbol{\pi}^T\boldsymbol{t}}, \tag{3.18}$$

$$G(I - P) = I - \frac{\mathbb{1}_n \boldsymbol{u}^T}{\boldsymbol{u}^T \mathbb{1}_n}, \text{ and} \tag{3.19}$$

$$\frac{\mathbb{1}_n}{\boldsymbol{u}^T \mathbb{1}_n} = G\boldsymbol{t}. \tag{3.20}$$

If $\boldsymbol{t} = (P \circ D)\mathbb{1}_n$ *and* $\boldsymbol{u}^T = \boldsymbol{\pi}^T$ *then*

$$\boldsymbol{\pi}^T G = \frac{\boldsymbol{\pi}^T}{\boldsymbol{\pi}^T (P \circ D)\mathbb{1}_n} \tag{3.21}$$

Proof: First, notice that $(I - P + \boldsymbol{t}\boldsymbol{u}^T)(I - P + \boldsymbol{t}\boldsymbol{u}^T)^{-1} = I$ implies that

$$(I - P)G = I - \boldsymbol{t}\boldsymbol{u}^T G. \tag{3.22}$$

Multiplying both sides on the left by $\boldsymbol{\pi}^T$ and noting that $\boldsymbol{\pi}^T(I - P) = 0$ gives that $\boldsymbol{\pi}^T = (\boldsymbol{\pi}^T \boldsymbol{t})\boldsymbol{u}^T G$. Dividing through by $(\boldsymbol{\pi}^T \boldsymbol{t})$ gives

$$\frac{\boldsymbol{\pi}^T}{\boldsymbol{\pi}^T \boldsymbol{t}} = \boldsymbol{u}^T G, \tag{3.23}$$

and substituting (3.23) into (3.22) gives (3.18).

Following a similar procedure as before with $(I - P + \boldsymbol{t}\boldsymbol{u}^T)^{-1}(I - P + \boldsymbol{t}\boldsymbol{u}^T) = I$, where we now multiply both sides on the right by $\mathbb{1}_n$ and note that $(I - P)\mathbb{1}_n = 0$ results in (3.20), which after appropriate substitution gives (3.19).

For the proof of equality (3.21), first we check that $\boldsymbol{t} = (P \circ D)\mathbb{1}_n$ and $\boldsymbol{u}^T = \boldsymbol{\pi}^T$ satisfy the conditions of Theorem 32. The definition of D guarantees that $P \circ D$ has at least one non-zero element which implies $\boldsymbol{\pi}^T \boldsymbol{t} = \boldsymbol{\pi}^T (P \circ D)\mathbb{1}_n \neq 0$. Also, $u^T \mathbb{1}_n = \boldsymbol{\pi}^T \mathbb{1}_n = 1$. Now substituting $\boldsymbol{u}$ and $\boldsymbol{t}$ into (3.23) gives (3.21). ■

Chapter 4

Robotic Surveillance: Quickest Anomaly Detection

In this chapter we study a surveillance problem in which anomalies in regions of a networked environment have to be detected based on noisy observations from the regions as soon as possible after their occurence. Motivating applications for this setup are surveillance tasks like detection of wilde fires, oil spills and other environmental monitoring tasks in which large number of regions have to be monitored under extreme sensor and modeling uncertainties. In many such scenarios, limited number of vehicles have to monitor a large number of regions and the vehicles have to be deployed in a manner which is most conducive to quick anomaly detection.

The framework for the surveillance problem studied in this chapter was introduced in [95]. The authors in [95] proposed a Markov chain based routing policy termed the *Ensemble CUSUM Algorithm* for this purpose. The Markov chains that they considered were required to have transition matrices with identical columns.

4.1 Contributions

We revisit the surveillance problem studied in [95]. The authors in [95] considered an all to all graph topology more suitable for aerial vehicles. We extend their setup to weighted graphs with arbitrary topologies, which we broadly refer to as robotic roadmaps. Further, we only keep the assumption of irreducibility on the Markov chains and look at a wider class of Markov chains than the one considered in [95]. We determine an expression for the average detection delay in the generalized setting and find that it depends on the first passage times of the Markov chain corresponding to the routing policy. We then frame an optimization problem to find the Markov chain corresponding to the *optimal policy* which minimizes the detection delay. We also provide an upper bound on the minimum detection delay and frame an optimization problem to minimize the upper bound. We prove that the upper bound optimization problem is convex and provide a semidefinite program (SDP) formulation to solve it and obtain the corresponding *efficient policy*. Using an illustrative example, we validate our expression for the detection delay and also surmise that the efficient policy provides a detection delay close to that of the optimal policy.

4.2 Organization

In Section 4.3, we describe the setup of the surveillance problem, formally define the *quickest detection task* and state the *Ensemble CUSUM Algorithm* to address the task. In Section 4.4, we review results on the CUSUM algorithm and the mean first passage time of Markov chain random walks on graphs which will be used to analyze the Ensemble CUSUM Algorithm. In Section 4.5, we analyze the performance of the Ensemble CUSUM Algorithm and provide an upper bound on its performance. In Section 4.6, we present

numerical simulations which validate our findings.

4.3 Problem Setup

We first describe our model for the environment and the mathematical model used for simulating the presence of anomalies in the environment.

4.3.1 Environment

A networked environment similar to that in Chapter 3 is consired in this chapter as well and is reviewed here for sake of completeness. The environment is modeled as a graph $G = (V, E)$ with node set $V := \{1, \ldots, n\}$ and edge set $E \subseteq V \times V$. The nodes in the graph correspond to the regions in the environment and the edges correspond to the interconnections between them. The time taken to travel from region i to the neighboring region j is d_{ij} and travel time matrix $D = [d_{ij}] \in \mathbb{R}^{n \times n}$ with the property that $d_{ij} \geq 0$ if $(i, j) \in E$ and $d_{ij} = 0$ otherwise.

The level of importance w_i is assigned to region i and $w = [w_i] \in \mathbb{R}^{n \times 1}$ is the referred to as the *priority vector*. Without loss of generality, $w^T \mathbb{1}_n = 1$. The environment can thus be described by the 4-tuple: $\mathcal{E} = \langle V, E, D, w \rangle$. An example of the environment and the graph corresponding to it is shown in Fig. 4.1 and Fig. 4.2 respectively.

4.3.2 Observations in Environment

When the surveillance vehicle visits a region in the environment, it makes an observation about the region. Based on all the observations made in the region up to that point, it predicts the presence of anomalies in the region.

Let the set of observations made by the surveillance vehicle at the region k be

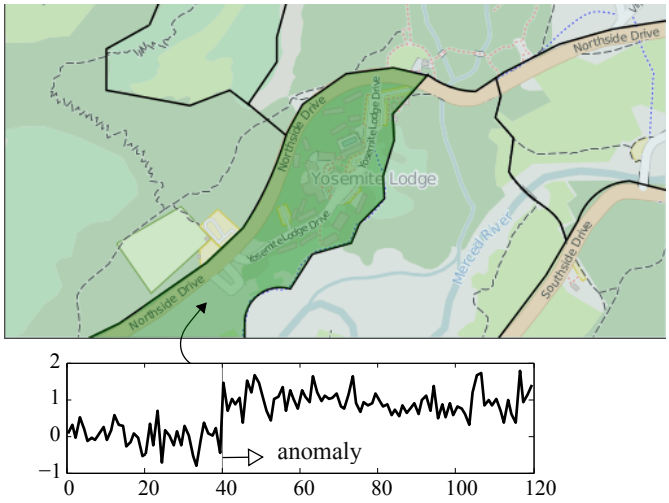

Figure 4.1: The environment is an area separated into seven regions of interest. Observations made in the highlighted region change after an anomaly occurs. The aim of the surveillance vehicle is to detect this change as soon as possible.

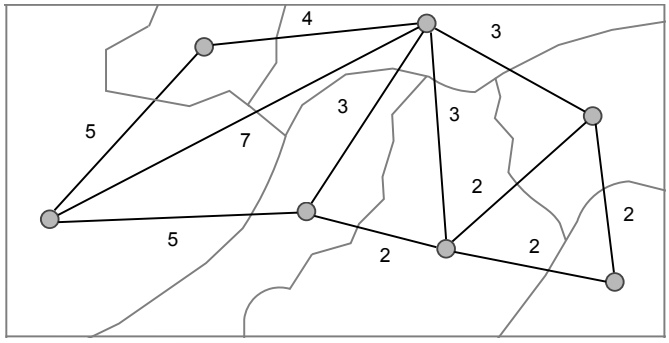

Figure 4.2: The robotic roadmap corresponding to the environment can be represented by a graph. The edge weights of the graph represent travel times between neighboring regions.

$\{y_{k,1}, y_{k,2}, \ldots\}$. If an anomaly occurs in the region at some iteration v, then the observations $\{y_{k,1}, \ldots y_{k,v}\}$ are i.i.d. with probability density function f_k^0 and the observations $\{y_{k,v}, y_{k,v+1}, \ldots\}$ are i.i.d. with probability density function f_k^1. We use the notation $\mathcal{D}(f_k^1, f_k^0)$ to denote the Kullback-Leibler divergence of f_k^0 from f_k^1 and also denote $\mathcal{D}_k := \mathcal{D}(f_k^1, f_k^0)$ for convenience. We now describe the *spatial quickest detection task* and quantify it.

4.3.3 Quickest Detection of Anomalies

The surveillance vehicle adopts a policy described by the tuple $\mathcal{P} = \langle P, q \rangle$. It moves in the environment $\mathcal{E} = \langle V, E, D, w \rangle$ according to a Markov chain with stationary distribution $q = [q_i] \in \mathbb{R}^{n \times 1}$ and transition matrix $P = [p_{ij}] \in \mathbb{R}^{n \times n}$. If $(i, j) \in E$, then $p_{ij} \geq 0$ and $p_{ij} = 0$ otherwise.

The aim of the vehicle is to detect anomalies in a region based on observations made in that region in least amount of time possible. More specifically, using a routing policy $\mathcal{P}$, it is required to minimize the average detection delay defined below.

Definition 34 (Average Detection Delay) *Let the vehicle service the environment $\mathcal{E} = \langle V, E, D, w \rangle$ using policy $\mathcal{P}$ for $k \in \{1, \ldots, n\}$ and let $\delta_k(\mathcal{P})$ be the delay in detecting an anomaly at region k. Then the task of the vehicle is to minimize the* average detection delay *$\delta_{avg}(\mathcal{P})$ given by*

$$\delta_{avg}(\mathcal{P}) = \sum_{k=1}^{n} w_k \mathbb{E}[\delta_k(\mathcal{P})]. \tag{4.1}$$

4.3.4 Ensemble CUSUM Algorithm

The surveillance vehicle visits the regions in $\mathcal{E}$ according to a realization of the Markov chain with stationary distribution q and transition matrix P. When the vehicle is in a particular region of the environment, it runs a local version of the *CUSUM algorithm*. We refer to the n parallel CUSUM algorithms by *Ensemble CUSUM Algorithm* (Algorithm 4). We wish to find the surveillance policy $\mathcal{P} = \langle P, q \rangle$ for the environment $\mathcal{E} = \langle V, E, D, w \rangle$ which minimizes the average detection delay $\delta_{\text{avg}}(\mathcal{P})$ defined in equation (4.1) in the previous section.

Remark 35 *(Service times): The service times required for conducting surveillance in different regions are not modeled in the problem setup. However, they can be incorporated*

Algorithm 4: Ensemble CUSUM Algorithm

Given: Policy $\mathcal{P} = \langle P, q \rangle$, threshold η, initial state x.
Set: $\Lambda_{k,0} = 0$ for $k \in \{1, \ldots, n\}$, local variable $\tau = 1$ for all regions.

1 Make observation $y_{k,\tau}$ at region x. ;
2 $\Lambda_{x,\tau} = \left(\Lambda_{x,\tau-1} + \log \frac{f_x^1(y_{x,\tau})}{f_x^0(y_{x,\tau})}\right)^+$;
3 **if** $\Lambda_{x,\tau} > \eta$ **then**
4 | Declare an anomaly at region x.;
5 | Set $\Lambda_{x,\tau} = 0$;
6 **end**
7 Set $\tau \leftarrow \tau + 1$ for x;
8 Select $x \leftarrow z$ with probability $P(x, z)$;
9 Repeat from step 1.

in a straighforward manner. If $v \in \mathbb{R}^{n\times 1}$ is the constant vector of service times, they can be accounted for by modifying the travel time matrix to $\bar{D} := D + \mathbb{1}_n v^T$.

Remark 36 *(Knowledge of probability density functions): The probability density functions in the absence and presence of anomalies are assumed to be known to the surveillance vehicle. In a scenario where the probability density functions are not known, the CUSUM algoithm can be replaced by the minimax robust quickest change detection algorithm [103] and the results presented in this chapter can be extended to apply to that scenario as well.*

4.4 Preliminary Results

We will now state some preliminary results which will be used in analysing the Ensemble CUSUM Algorithm. We will start by reviewing some performance guarantees on the CUSUM algorithm.

4.4.1 CUSUM Algorithm

The CUSUM algorithm is designed for quick prediction of anomalies while at the same time, avoiding making *false alarms* [88]. In the CUSUM algorithm, at each iteration $\tau \in \mathbb{N}$ made in region k, (i) observation $y_{k,\tau}$ is collected, (ii) the statistic

$$\Lambda_{k,\tau} = \left(\Lambda_{k,\tau-1} + \log \frac{f_k^1(y_{k,\tau})}{f_k^0(y_{k,\tau})} \right)^+$$

with $\Lambda_{k,0} = 0$ is computed and (iii) a change is declared if $\Lambda_{k,\tau} > \eta$. Let O_k be the observation at which an anomaly is declared at region k. For a given threshold η, the expectation of O_k conditioned on the presence of an anomaly, i.e. the worst expected number of observations of the CUSUM algorithm is

$$\mathbb{E}_{f_k^1}(O_k) \approx \frac{e^{-\eta} + \eta - 1}{\mathcal{D}(f_k^1, f_k^0)} = \frac{\bar{\eta}}{\mathcal{D}_k}, \tag{4.2}$$

where $\bar{\eta} = e^{-\eta} + \eta - 1$, and the expectation of O_k conditioned on the absence of an anomaly, i.e. the false alarm rate for CUSUM algorithm is

$$\mathbb{E}_{f_k^0}(O_k) \approx \frac{e^{\eta} - \eta - 1}{\mathcal{D}(f_k^0, f_k^1)} \tag{4.3}$$

The approximations in equations (4.2,4.3) are referred to as the Walds approximations [88]. For large values of the threshold η, these approximations are known to be accurate. We also set

$$s_k := \frac{\bar{\eta}}{\mathcal{D}_k}, \tag{4.4}$$

and $s = [s_k] \in \mathbb{R}^{n \times 1}$, referring to it as the *vector of CUSUM samples*. Given $\bar{\eta}$ and $\mathcal{D}_k$, the constant s_k is the expected number of visits to region k required to detect an anomaly

in that region.

The expression for the average detection delay of the Ensemble CUSUM Algorithm (Algorithm 4) depends on the property of Markov chains called the first passage time stated in Chapter 3.

4.5 Performance of the Ensemble CUSUM Algorithm

We are now ready to state our main results on the average detection delay $\delta_{\text{avg}}(\mathcal{P})$ of the Ensemble CUSUM Algorithm (Algorithm 4).

Theorem 37 *(Performance of the Ensemble CUSUM Algorithm): For a single vehicle conducting surveillance of the environment $\mathcal{E} = \langle V, E, D, w\rangle$ according to the Ensemble CUSUM Algorithm (Algorithm 4) using the policy $\mathcal{P} = \langle P, q\rangle$,*

(i) the expected detection delay $\mathbb{E}[\delta_k(\mathcal{P})]$ at region k satisfies

$$\mathbb{E}[\delta_k(\mathcal{P})] = \sum_{i=1}^{n} q_i n_{ik} + (s_k - 1)\, n_{kk}, \tag{4.5}$$

(ii) the average detection delay $\delta_{avg}(\mathcal{P})$ over the entire environment satisfies

$$\delta_{avg}(\mathcal{P}) = \sum_{k=1}^{n} w_k \left(\sum_{i=1}^{n} q_i n_{ik} + (s_k - 1)\, n_{kk} \right), \tag{4.6}$$

where $N = [n_{ij}] \in \mathbb{R}^{n\times n}$ is the first passage time matrix for the irreducible Markov chain with transition matrix $P \in \mathbb{R}^{n\times n}$ and stationary distribution $q \in \mathbb{R}^{n\times 1}$ and the constant $s \in \mathbb{R}^{n\times 1}$ is the vector of CUSUM samples.

Proof: Let $\tau \in \{1, \ldots, O_k\}$ be the iterations at which the vehicle visits region region k and sends information about it to the control center. Let O_k be the iteration at

which an anomaly is detected in region k. The observation made at region k at the τ-th iteration in that region is denoted by $y_{k,\tau}$. Let the log likelihood ratio calculated by the local CUSUM algorithm for that iteration be $\epsilon_{k,\tau}$. Then,

$$\epsilon_{k,\tau} = \log \frac{f_k^1(y_{k,\tau})}{f_k^0(y_{k,\tau})}.$$

Conditioned on the presence of an anomaly, $\{\epsilon_{k,\tau}\}_{\tau_k \in \mathbb{N}}$ are i.i.d. and $\mathbb{E}_{f_k^1}[\epsilon_{k,\tau}] = \mathcal{D}_k$. Then, referring to result summarized in equation (4.2), $\mathbb{E}_{f_k^1}[O_k] = \bar{\eta}/\mathcal{D}_k$. Thus, the expected time it takes for the Ensemble CUSUM Algorithm to make the O_k-th observation at region k is essentially the expected detection delay $\delta_k(\mathcal{P})$ at region k. We will now devote our attention to computing the expectation of $\delta_k(\mathcal{P})$.

Let t_k^0 be the time at which the vehicle starts the CUSUM algorithm. Let $\{t_k^1, t_k^2, \ldots t_k^{O_k}\}$ be the time instant at which it leaves region k, having serviced it, and $\Delta t_k^i = t_k^{i+1} - t_k^i$ for $i = \{0, 1, 2, \ldots\}$. Then, the detection delay $\delta_k(P) = t_k^{O_k} = \sum_{i=0}^{O_k} \Delta t_k^i$. The expectation of $\delta_k(P)$ can be computed:

$$\begin{aligned}
\mathbb{E}[\delta_k(P)] =& \mathbb{E}\left[\sum_{i=0}^{O_k-1} \Delta t_k^i\right] = \mathbb{E}[\Delta t_k^0] + \mathbb{E}\left[\sum_{i=1}^{O_k-1} \Delta t_k^i\right], \\
=& \mathbb{E}[\Delta t_k^0] + (\mathbb{E}[O_k] - 1)\mathbb{E}[\Delta t_k^i] \qquad (4.7) \\
=& \sum_{i=1}^{n} q_i n_{ik} + \left(\frac{\bar{\eta}}{\mathcal{D}_k} - 1\right) n_{kk}. \qquad (4.8)
\end{aligned}$$

Equation (4.7) comes from the application of Wald's identity. Notice that $\mathbb{E}[\Delta t_k^0]$ is the expected time to start from any node and visit node k for the first time and given $i > 0$, $\mathbb{E}[\Delta t_k^i]$ is the expected time taken to return to node k. Recollect that n_{ij} is the expected time for the vehicle to start from node i to visits node j for the first time. Hence, we can conclude that $\mathbb{E}[\Delta t_k^0] = \sum_{i=1}^{n} q_i n_{ik}$ and $\mathbb{E}[\Delta t_k^i] = n_{kk}$ for $i > 0$ to obtain equation (4.8).

Using the definition of s from equation (4.4), the first result follows. Next, using the definition of $\delta_{\text{avg}}(\mathcal{P})$ from equation (4.1), the second result follows. ■

Thus, the average detection delay depends on the first passage times between nodes of the graph representing the environment $\mathcal{E}$. We now present a modified expression for $\delta_{\text{avg}}(\mathcal{P})$, removing the dependence of the first passage times, in the following theorem. The proof of the theorem is postponed to Appendix.

Theorem 38 *(Average detection delay): For a single vehicle conducting surveillance of the environment $\mathcal{E} = \langle V, E, D, w\rangle$ according to the Ensemble CUSUM Algorithm (Algorithm 4) using the policy $\mathcal{P} = \langle P, q\rangle$,*

$$\delta_{avg}(\mathcal{P}) = \beta \mathbb{1}_n^T[(((I-P)+(P\circ D)\mathbb{1}_n q^T))^{-1}\circ I](r\cdot w) + (\beta-1) + \beta(s-\mathbb{1}_n)^T(r\cdot w), \tag{4.9}$$

where $r \in \mathbb{R}^{n\times 1}$ with $r\cdot q = \mathbb{1}_n$, $I \in \mathbb{R}^{n\times n}$ is the identity matrix, $\beta = q^T(P\circ D)\mathbb{1}_n$ and the constant $s \in \mathbb{R}^{n\times 1}$ is the vector of CUSUM samples.

In our setup, the environment can have an arbitrary graph topology and the routing policy can also take an arbitrary form adhering to the restrictions imposed by the graph topology. A specific simplification, where the environment is an all to all graph and where the transition matrix for the routing policy has the form $P = \mathbb{1}_n q^T$ is explored in [95]. While they provide algorithms to optimize the transition matrix q in the simplified setup, we consider the more generalized problem. Specifically, our goal is to find policy $\mathcal{P} = \langle P, q\rangle$ for the vehicle such that $\delta_{\text{avg}}(\mathcal{P})$ is minimized. This can be framed as the following optimization problem:

Problem 6 *(Minimizing the average detection delay): Given the environment $\mathcal{E} = \langle V, E, D, w\rangle$ and the constant vector of CUSUM samples $s \in \mathbb{R}^{n\times 1}$, determine the*

stationary distribution $q = [q_i] \in \mathbb{R}^{n\times 1}$ *and transition probabilities* $P = [p_{ij}] \in \mathbb{R}^{n\times n}$ *solving:*

$$\begin{aligned}
\text{minimize} \quad & \beta \mathbb{1}_n^T[(((I-P)+(P\circ D)\mathbb{1}_n q^T))^{-1}\circ I](r\cdot w) \\
& + (\beta-1) + \beta(s-\mathbb{1}_n)^T(r\cdot w) \\
\text{subject to} \quad & P\mathbb{1}_n = \mathbb{1}_n, \ \textit{for each } i \in \{1,\dots,n\} \\
& 0 \le p_{ij} \le 1, \ \textit{for each } (i,j) \in E \\
& p_{ij} = 0, \ \textit{for each } (i,j) \notin E \\
& q^T P = q^T, \ \textit{for each } (i,j) \in E \\
& q^T\mathbb{1}_n = 1, q_i \ge 0, \ \textit{for each } i \in \{1,\dots,n\} \\
& P \ \textit{irreducible}, \\
& \beta = q^T(P\circ D)\mathbb{1}_n, \ r\cdot q = \mathbb{1}_n.
\end{aligned}$$

The above optimization problem contains the constraint that the transition matrix be irreducible. Since it is hard to enforce the irreducibility constraint during each step of an iterative optimization algorithm, our approach is to relax the irreducibility constraint and verify that the final solution satisfies the constraint. A Markov chain that is not irreducible contains multiple communicating classes, making the first passage time between at least one pair of regions infinite. Since the average detection delay depends on the first passage times of the chain, the outcome where the final solution is a reducible chain would drive up the cost function of the optimization problem, making such an outcome highly unlikely. Because of this reason, the relaxation of the irreducibility constraint works very well in practice.

Let $\mathcal{P}^*$ be the solution to Problem 6 and let $\delta^*_{\text{avg}} := \delta_{\text{avg}}(\mathcal{P}^*)$. The cost function of this optimization problem is not a convex function of P and q. Moreover, one of

the constraints is also nonlinear. We now devote some attention to determining an upper bound on δ^*_{avg}, and frame an optimization problem to minimize it. We start with evaluating policies of the form $\mathcal{P}_w = \langle P_w, w\rangle$, i.e., where the Markov chain corresponding to the policy has the stationary distribution w. We leverage the result known on the weighted sum of the first passage times from Theorem 25 to simplify expressions for the detection delay of the Ensemble CUSUM algorithm in this case.

Theorem 39 *(Upper bound on average detection delay): For a single vehicle conducting surveillance of the environment $\mathcal{E} = \langle V, E, D, w\rangle$ according to the Ensemble CUSUM Algorithm (Algorithm 4) using the policy $\mathcal{P}_w = \langle P_w, w\rangle$,*

$$\delta_{avg}(\mathcal{P}_w) = \left(w^T(P_w \circ D)\mathbb{1}_n\right)\left(\sum_{i=2}^{n}\frac{1}{1-\lambda_i(P_w)} + (s-\mathbb{1}_n)^T\mathbb{1}_n\right)$$

for all $P_w \in S_w$, where S_w is the set of transition matrices corresponding to irreducible Markov chains with stationary distribution w, $\{\lambda_1(P_w), \ldots, \lambda_n(P_w)\}$ are the eigenvalues of P_w with $\lambda_1(P_w) = 1$, and the constant $s \in \mathbb{R}^{n\times 1}$ is the vector of CUSUM samples.

Proof: We start with the expression for $\delta_{\text{avg}}(\mathcal{P})$ obtained in Theorem 37. In matrix form, equation (4.6) can be rewritten as

$$\delta_{\text{avg}}(\mathcal{P}) = q^T N w + (s-\mathbb{1}_n)^T N_d w, \tag{4.10}$$

where $\mathcal{P} = \langle P, q\rangle$. Setting the variable q to w, and using the result from Theorem 25, as well as the result from Theorem. 23 (ii), equation (4.10) can be simplified:

$$\begin{aligned}\delta_{\text{avg}}(\mathcal{P}_w) =& w^T N w + (s-\mathbb{1}_n)^T N_d w,\\ =& \left(w^T(P_w \circ D)\mathbb{1}_n\right)\left(\sum_{i=2}^{n}\frac{1}{1-\lambda_i(P_w)} + (s-\mathbb{1}_n)^T\mathbb{1}_n\right),\end{aligned}$$

■

We can make the upper bound obtained on the optimal detection delay tighter by choosing $P_{\text{ub}} \in S_w$ which minimizes the average detection delay. The following optimization problem can be framed to find the matrix P_{ub}.

Problem 7 *(Minimizing the upper bound on optimal average detection delay): Given the environment $\mathcal{E} = \langle V, E, D, w \rangle$, the vector of CUSUM samples $s \in \mathbb{R}^{n\times 1}$ and the stationary distribution $w \in \mathbb{R}^{n\times 1}$, determine the transition probabilities $P = [p_{ij}] \in \mathbb{R}^{n\times n}$ solving:*

$$\begin{aligned} \text{minimize} \quad & \left(w^T(P \circ D)\mathbb{1}_n\right)\left(\sum_{i=2}^{n} \frac{1}{1-\lambda_i(P)} + (s-\mathbb{1}_n)^T\mathbb{1}_n\right) \\ \text{subject to} \quad & P\mathbb{1}_n = \mathbb{1}_n, \ \textit{for each } i \in \{1,\ldots,n\} \\ & 0 \le p_{ij} \le 1, \ \textit{for each } (i,j) \in E \\ & p_{ij} = 0, \ \textit{for each } (i,j) \notin E \\ & w_i p_{ij} = w_j p_{ji}, \ \textit{for each } (i,j) \in E. \end{aligned} \tag{4.11}$$

Note that the above optimization problem also involves the restriction of non-reversibility on the transition matrix P as denoted by the last equality.

Theorem 40 *(Convexity of Optimization Problem 7): Let S_w be the set of transition matrices associated with irreducible non-reversible Markov chains on graph $G = (V, E)$ and having the stationary distribution w. Then, the Optimization Problem 7 is convex.*

Proof: From Theorem 39, the cost function $f(P_w)$ of the Optimization Problem 7 can be written down as:

$$f(P_w) = \left(w^T(P_w \circ D)\mathbb{1}_n\right)\left(\sum_{i=2}^{n} \frac{1}{1-\lambda_i(P_w)}\right) + \left(w^T(P_w \circ D)\mathbb{1}_n\right)(s-\mathbb{1}_n)^T\mathbb{1}_n. \tag{4.12}$$

The first term in equation (4.12) is the mean first passage time of the Markov chain as defined in Theorem 25. The mean first passage time is a convex function over the set S_w (refer to [78] for the proof). Moreoever, the second term in equation (4.12) is an affine function over the set S_w. Since the positive weighted sum of convex and affine functions is convex, the function $P_w \mapsto f(P_w)$ is convex over the set S_w. The set S_w is also convex and the constraints of the problem are affine. Hence, the optimization problem is convex.

■

The Optimization Problem 7 can be written as a semidefinite program. In order to do this, the expression for the detection delay is rewritten in terms of the trace of a matrix:

$$\begin{aligned}\delta_{\text{avg}}(\mathcal{P}_w) &= \left(w^T(P_w \circ D)\mathbb{1}_n\right)\left(\sum_{i=2}^{n}\frac{1}{1-\lambda_i(P_w)} + (s-\mathbb{1}_n)^T\mathbb{1}_n\right),\\ &= \left(w^T(P_w \circ D)\mathbb{1}_n\right)\mathbf{Tr}\left[(I - W^{1/2}P_wW^{-1/2} + w_cw_c^T)^{-1}\right]\\ &\quad + \left(w^T(P_w \circ D)\mathbb{1}_n\right)(s-\mathbb{1}_n)^T\mathbb{1}_n,\end{aligned}$$

where $W = \text{diag}[w]$ and the column vector $w_c = (\sqrt{w_1}, \ldots, \sqrt{w_n})^T$. The first equation comes from Theorem 39 and the first part of the second equation is because of a relation between the trace of a function of P_w and its eigenvalues [78]. Using this form for $\delta_{\text{avg}}(\mathcal{P}_w)$, we can now formulate an SDP as shown below.

Problem 8 *(Minimizing the upper bound on the optimal average detection delay (SDP)): Given the environment $\mathcal{E} = \langle V, E, D, w\rangle$ and vector of CUSUM samples $s \in \mathbb{R}^{n\times 1}$, with $W = diag[w]$ and $w_c = (\sqrt{w_1}, \ldots, \sqrt{w_n})^T$, determine $Y = [y_{ij}] \in \mathbb{R}^{n\times n}$, $X \in \mathbb{R}^{n\times n}$, $t \in \mathbb{R}$*

and $u \in \mathbb{R}$ solving:

$$\begin{aligned}
&\text{minimize} \quad \mathbf{Tr}[X] + u(s^T \mathbb{1}_n) \\
&\text{subject to} \\
&\qquad \begin{bmatrix} t(I + w_c w_c^T) - W^{1/2} Y W^{-1/2} & I \\ I & X \end{bmatrix} > 0 \\
&\qquad \begin{bmatrix} t & 1 \\ 1 & u \end{bmatrix} > 0 \\
&\qquad \sum_{j=1}^{n} y_{ij} = t, \ \textit{for each } i \in \{1, \ldots, n\} \\
&\qquad w_i y_{ij} = w_j y_{ji}, \ \textit{for each } (i,j) \in E \\
&\qquad 0 \leq y_{ij} \leq t, \ \textit{for each } (i,j) \in E \\
&\qquad y_{ij} = 0, \ \textit{for each } (i,j) \notin E \\
&\qquad w^T (Y \circ W) \mathbb{1}_n = 1 \\
&\qquad t \geq 0.
\end{aligned}$$

Then, the transition matrix P_w is given by $P_w = Y/t$.

Let P_{ub} be the solution to the Optimization Problem 7. We refer to the policy $\mathcal{P}_{\text{ub}} = \langle P_{\text{ub}}, w \rangle$ as the *efficient policy* for convenience.

4.6 Numerical Simulations

We now study the spatial quickest detection task for a specific environment. In particular, we are interested in examining the efficiency of the upper bound service policy $\mathcal{P}_{\text{ub}} = \langle P_{\text{ub}}, w \rangle$ compared to the optimal policy $\mathcal{P}^*$ which minimizes the average detection

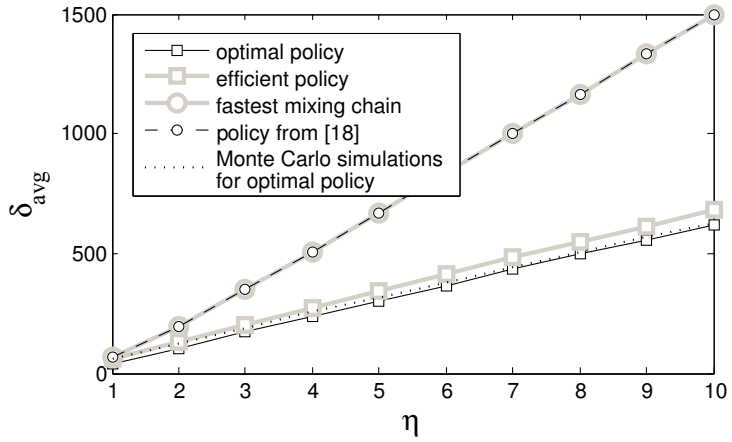

Figure 4.3: Variation of the average detection delay using the optimal policy δ^*_{avg} (black squares), the efficient policy δ_{ub} (grey squares), the policy based on the fastest mixing non-reversible Markov chain with a uniform stationary distribution (grey circles) and the policy in [95] (black circles) with respect to the threshold η of the CUSUM algorithm. Expected detection delay for the optimal policy using Monte Carlo Simulations (dashed lines).

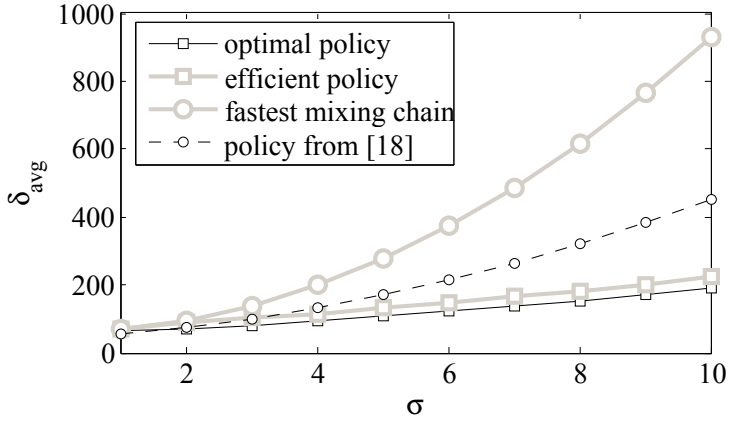

Figure 4.4: Average detection delay using the optimal policy δ^*_{avg} (black squares) and the efficient policy δ_{ub} (grey squares) are compared with the average detection delay obtained using the policy based on the fastest mixing non-reversible Markov chain (grey circles) with a uniform stationary distribution and the policy from [95] (black circles) for various levels on noise in observations made in the second region.

delay. We also compare these two policies to some candidate policies (namely the policy based on the fastest mixing non-reversible Markov chain [18] and a policy proposed in [95]).

Environment and modeling of anomalies: The environment (Fig. 4.1) is an

area separated into seven regions of interest. The weighted graph corresponding to this environment is shown in Fig. 4.2. The edge weights of this graph represent the travel times between neighboring regions. All regions in the environment have equal priority, so that $w = \mathbb{1}_n$, and the service time required to make an observation in each of the regions is one time unit. The probability density functions of the observations made in the environment in the absence and presence of anomalies are normal distributions $f_k^0 = \mathcal{N}(0,1)$ and $f_k^1 = \mathcal{N}(1,1)$ respectively for $k \in \{1,\dots,n\}$.

Computation of service policies: The Optimization Problem 6 to determine the optimal policy $\mathcal{P}^*$ is non-convex with nonlinear constraints. We solve it using the sqp algorithm in Matlab and verify that the solutions obtained are at least local minima. This is done by ensuring that the solutions satisfies 1. the regularity condition and 2. the Karush-Kuhn-Tucker (KKT) conditions necessary for the solution to optimal. On the other hand, the Optimization Problem 7 to compute the Markov chain corresponding to the upper bound service policy $\mathcal{P}_{\text{ub}}$ is convex and can be written as a semidefinite program. It is solved using CVX, a Matlab-based package for convex programs [45]. The fastest mixing non-reversible Markov chain is also computed by solving a semidefinite program in CVX. The policy proposed in [95] is stated as follows: $\mathcal{P}^\dagger = \langle P^\dagger, q^\dagger \rangle$ where

$$q_k^\dagger = \frac{\sqrt{w_k/\mathcal{D}_k}}{\sum_{j=1}^n \sqrt{w_j/\mathcal{D}_j}},\ k \in \{1,\dots,n\}$$

and $P^\dagger$ is the fastest mixing non-reversible Markov chain with stationary distribution $q^\dagger$.

Validation of theoretical expressions: We start with comparing the theoretical expression for the average detection delay δ_{avg} in the environment obtained in Theorem 38 (black squares) to the expected detection delay computed through Monte-Carlo simulations (dotted lines) in Fig 4.3 for the optimal policy. The gap between the theoretical and the numerically obtained values is attributed to Wald's approximation introduced in

equation (4.2).

Comparison of performances of service policies: We first compare variation in the performance of various service policies with respect to different thresholds η of the CUSUM algorithm in Fig. 4.3(a). The average detection delay δ_{ub} obtained using the efficient policy $\mathcal{P}_{\text{ub}}$ (grey squares) is close to the optimal average detection delay δ^*_{avg} (black squares) for lower values of the threshold η. The gap observed between the optimal solution and the upper bound can be attributed to two factors: freedom to choose any stationary distribution as well as relaxation of the nonreversibility constraint for computing the optimal solution. In comparison, the performance of the fastest mixing non-reversible Markov chain with stationary distribution $w = \mathbb{1}_n$ (grey circles) and the policy proposed in Theorem 6 in [95] (black circles) is much poorer. This is expected since the efficient policy is guaranteed to perform better in comparison to the fastest mixing non-reversible Markov chain with the same stationary distribution.

Next, we study the effect of the variation in the probability density functions of observations on the performance of the service policies. We consider a scenario where the second region, which is a residential area in our illustration, is affected by noisy observations. While the probability distribution functions for observations in all the other regions remain same, they are different for the second region: $f_2^0 = \mathcal{N}(0, \sigma)$ and $f_2^1 = \mathcal{N}(1, \sigma)$. The average detection delays for the various policies considered in the chapter are compared for different values of σ in Fig. 4.4. The performance of the efficient policy (grey squares) is very close to the optimal performance (black squares) for a wide range of σ in this case. In comparison, the performances of the policy based on the fastest mixing non-reversible chain with stationary distribution $w = \mathbb{1}_n$ (grey circles) and the policy from [95] (black circles) are much poorer. They also get worse for noisier observations.

Appendix

Proof of Theorem 38: We start from the expression for $\delta_{\text{avg}}(\mathcal{P})$ obtained in Theorem 37. Using the definition of s stated in the statement of the theorem, and equation (4.6), the expression for $\delta_{\text{avg}}(\mathcal{P})$ can be written in matrix form as follows:

$$\delta_{\text{avg}}(\mathcal{P}) = q^T N w + (s - \mathbb{1}_n)^T N_d w. \tag{4.13}$$

We first work towards simplifying the first term in equation (4.13). The first passage time matrix satisfies equation (3.6). Using this and the identity from Lemma 33, and the assumption that $\mathbb{1}_n^T w = 1$, the term $q^T N w$ can be simplified:

$$\begin{aligned} q^T N w =& q^T \beta(\mathbb{1}_n \mathbb{1}_n^T G_d + I - G) Q^{-1} w, = \beta(\mathbb{1}_n^T G_d + q^T(I - G)) Q^{-1} w, \\ =& \beta(\mathbb{1}_n^T G_d + q^T - \frac{q^T}{\beta}) Q^{-1} w = \beta \mathbb{1}_n^T G_d Q^{-1} w + (\beta - 1) q^T Q^{-1} w, \\ =& \beta \mathbb{1}_n^T G_d Q^{-1} w + (\beta - 1)\mathbb{1}_n^T w = \beta \mathbb{1}_n^T G_d Q^{-1} w + (\beta - 1). \end{aligned} \tag{4.14}$$

where $Q = \text{diag}[q]$. Looking at the first term in equation (4.14),

$$\begin{aligned} \mathbb{1}_n^T G_d Q^{-1} w =& \mathbb{1}_n^T [(((I - P) + (P \circ D)\mathbb{1}_n q^T))^{-1} \circ Q^{-1}] w, \\ =& \mathbb{1}_n^T [(((I - P) + (P \circ D)\mathbb{1}_n q^T))^{-1} \circ I](r \cdot w). \end{aligned}$$

Substituting $N_d w = \beta Q^{-1} w = \beta(r \circ w)$ from equation (3.8) into the second term in equation (4.13), the result follows.

Chapter 5

Synchronization of Beads on a Ring

We now consider the problem of synchronization of n robotic agents that control their motion on a ring and that communicate when in close proximity of each other. One application of this setup is as a boundary-patrolling algorithm which has numerous applications, e.g. in monitoring of spreading fires, toxic-area containment and clean-up, and the sensing of sharp temperature gradient surfaces in the sea. These algorithms require sporadic communication among agents, which have to optimally divide the task among themselves without the intervention of a supervisor.

We therefore pose the question: can n intelligent agents (or beads), capable of controlling their motion, autonomously organize themselves so that each one sweeps a sector of the ring and impacts with the neighboring beads always at the boundaries of the sector? In other words, can they reach a periodic orbit and get in sync?

Apart from boundary-guarding the problem is also of general interest: If the n agents control their motion to simulate n beads sliding on a frictionless hoop, then we know that their dynamics is very rich. In fact, in [30], the authors study extensively the case of $n = 3$ and prove the existence of periodic as well as chaotic orbits.

5.1 Contributions

We state the contributions of this chapter [1] now. We design a distributed algorithm that allows a collection of beads to reach synchronization and that is robust to failure of beads. The algorithm requires the beads to slowdown and speedup immediately prior to and after impact respectively; accordingly, we refer to the algorithm as the "slowdown, impact and speedup algorithm." The beads can be deployed with arbitrary initial positions and speeds. At the desired steady state, every bead sweeps a sector of equal length, and neighboring beads meet always at the same point. If n is even, the beads all travel at the same speed, while if n is odd, the beads travel at the same average speed. Two beads exchange information only when they impact. We provide a sufficient condition on the initial positions of the beads to guarantee converegence of beads. Extensive simulations show that synchronization is reached in general, even when the assumptions are not satisfied. Moreover, our algorithm confers certain robustness properties on the emerging synchronized behavior, which is of interest for any control system.

5.2 Organization

This chapter is organized as follows. Section 5.1 states the contributions of the work. We then introduce the notation employed and describe in detail what is meant by agent or bead synchronization on a circular boundary in Section 5.3. The discrete-time synchronization algorithm is presented in Section 5.4. A set of preliminary results on which the main theorems build upon is presented in 5.5. The main results that allow us to analyze the algorithm are included in 5.6. Finally, we present simulations in Section 5.7 showing that convergence of the algorithm is indeed possible in most general cases.

[1] This work is in collaboration with Dr. Sara Susca and Prof. Sonia Martínez

Notation

On the ring or 1-sphere $\mathbb{S}^1$, by convention, let us define positions as angles measured counterclockwise from the positive horizontal axis. The *counterclockwise distance between two angles* $\mathrm{dist}_{\mathsf{cc}} : \mathbb{S}^1 \times \mathbb{S}^1 \to [0, 2\pi)$ is the path length from an angle to the other traveling counterclockwise. The column vector with entries all equal to 1 is $\mathbb{1}_n \in \mathbb{R}^n$. When working with indices in $\{1, \ldots, n\}$, we use the identifications $0 \equiv n$ and $n + 1 \equiv 1$.

5.3 Model and problem statement

Here we model a network of agents moving on a ring and we state our stabilization problem for certain interesting periodic modes. First, we propose our agents model with motion control, sensing and communication capabilities. The agents are at arbitrary positions $\theta_i \in \mathbb{S}^1$, $i \in \{1, \ldots, n\}$, at initial time and ordered counterclockwise.

Each agent controls its motion according to $\dot{\theta}_i(t) = u_i(t)$, where u_i is a bounded control signal. Each agent senses its own position on the ring and senses/distinguishes impacts with its counterclockwise and clockwise neighbors. However, agents do not need to have knowledge of their absolute positions in a global reference frame. Similarly, agents do not need to know the total number of beads n and the circle length. Each agent is equipped with a short-range communication device; for simplicity, we assume two agents communicate only when they are at the same position. In other words, two agents have *communication impacts* when they move to a coincident position. The algorithm can be implemented over anonymous agents; that is, agents lacking an identifier that can distinguish them from each other. However, for simplicity in formulating the problem, we make use of indices $i \in \{1, \ldots, n\}$ and make use of coordinates in a global reference frame. Finally, each agent is equipped with a processor, capable of storing quantities in memory and performing computations.

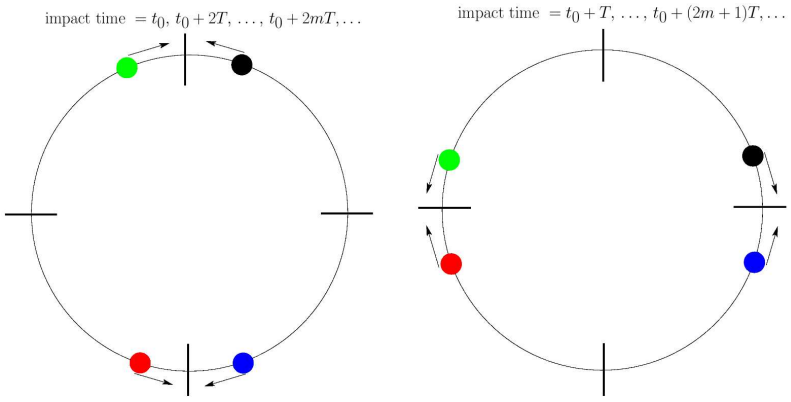

Figure 5.1: The figure shows a collection of four beads moving in balanced synchronization.

Next, we describe some interesting periodic trajectories for n beads moving on a ring. It is our objective, in the following sections, to design a motion control and communication algorithm to render such trajectories attractive.

Definition 41 (Balanced synchronization) *Consider a collection of n beads moving on a ring. The collection of beads is* balanced synchronized *with period T, if (i) any two neighboring beads impact always at the same point, (ii) the time interval between two consecutive impacts, involving the same beads, has duration T, and (iii) all the beads impact simultaneously. In other words, in a balanced synchronized collection, each bead sweeps an arc of length $2\pi/n$ at constant speed $2\frac{2\pi}{nT}$.*

An example of a collection of four beads in sync is shown in Figure 5.1: each bead sweeps an arc at the boundaries of which it impacts with one of its neighbors and the impacts happen simultaneously.

If n is odd, then balanced synchronization cannot be reached. Therefore, we give the following weaker synchronization notion, reachable also for odd n.

Definition 42 (Unbalanced synchronization) *Consider a collection of n beads moving on a ring. The collection of beads is* unbalanced synchronized *with period T, if (i)*

any two beads impact always at the same point and (ii) the time interval between two consecutive impacts, involving the same beads, has duration T. As before, in an unbalanced synchronized collection, each bead sweeps an arc of length $2\pi/n$ at average speed $2\frac{2\pi}{nT}$.

5.4 Synchronization algorithm

In this section we describe an algorithm that allows a collection of n agents to achieve balanced synchronization (for n even) and unbalanced synchronization (for n odd). We begin with an informal description for the case when n is even:

> Each agent changes its direction of motion when it impacts another agent with opposing velocity. Each agent maintains an estimate of the arc it eventually sweeps when the network asymptotically achieves balanced synchronization. This estimate is updated according to an averaging law at each communication impact (so that all estimated arcs converge to pairwise contiguous arcs of equal length). A similar averaging law is applied to the agent's speed to ensure that all agents' speeds converge to a common nominal value. To synchronize the back-and-forth motion inside the arcs, each agent travels at nominal speed while inside its arc, slows down when moving away from it, and speeds up when moving towards it after an impact.

We refer to this strategy as to the Slowdown, Impact and Speedup Algorithm, abbreviated as the SIS Algorithm. To provide a formal description, we begin by defining all variables that each agent maintains in its memory and we later state how these variables are updated as time evolves and communication impacts take place.

Algorithm Variables

Each agent $i \in \{1, \dots, n\}$ maintains in memory the following tuple:

$$\begin{aligned} v_i &\in \mathbb{R}_{>0}, && \text{the nominal speed,} \\ d_i &\in \{-1, +1\}, && \text{the direction of motion,} \\ a_i &\in \{-1, +1\}, && \text{the moving-away flag,} \\ \ell_i &\in \mathbb{S}^1, && \text{the arc lower boundary, and} \\ u_i &\in \mathbb{S}^1, && \text{the arc upper boundary.} \end{aligned}$$

Regarding initialization, we allow $v_i(0)$, $d_i(0)$ to be arbitrary and we set $\ell_i(0) = u_i(0) := \theta_i(0)$, and $a_i(0) := d_i(0)$.

Given these definitions, it is convenient to introduce the following notation and nomenclature. First, we define the ith processor state $x_i := (v_i, d_i, a_i, \ell_i, u_i)$ and call (θ_i, x_i) the ith agent state. Next, we associate an arc of the ring to each bead. This arc is the fraction of the ring that each bead eventually sweeps when balanced synchrony (as in Definition (41)) is asymptotically reached. To each bead i, we associate a *desired sweeping arc* defined by

$$\mathrm{Arc}(\ell_i, u_i) = \{\theta \in \mathbb{S}^1 \mid \mathrm{dist}_{\mathsf{cc}}(\ell_i, \theta) \leq \mathrm{dist}_{\mathsf{cc}}(\ell_i, u_i)\}.$$

This quantity will also be denoted by $\mathcal{D}_i$ for convenience henceforth.

Algorithm Rules

The algorithm rules specify how the agents move in continuous time and how they update their processor states when certain events happen.

First, at all time $t \geq 0$, each bead sets its velocity $\dot{\theta}_i$ depending on whether the bead is traveling inside its desired sweeping arc, or, if outside the sweeping arc, depending on whether it is moving away from or towards the sweeping arc. Specifically, given two scalar gains $\frac{1}{2} < f < 1$ and $h = \frac{f}{2f-1} > 1$, we set

$$\dot{\theta}_i(t) := d_i(t)v_i(t) \cdot \begin{cases} 1, & \text{if } \theta_i(t) \in \mathcal{D}_i, \\ f, & \text{if } \theta_i(t) \notin \mathcal{D}_i \text{ and } d_i(t) = a_i(t), \\ h, & \text{if } \theta_i(t) \notin \mathcal{D}_i \text{ and } d_i(t) = -a_i(t). \end{cases}$$

Second, the ith processor state changes only when one of the following two events occurs: an *Impact Event* which takes place with either bead $i-1$ or with bead $i+1$, or a *Crossing Event* which takes place when bead i crosses either ℓ_i or u_i while leaving its desired sweeping arc.

(Impact Event) If at time t an impact occurs for bead i with either bead $i+1$ or $i-1$, then: (1) the two beads exchange through communication their processors states, and (2) with this information, each bead updates its memory as follows. We define an impact between beads i and $i+1$ to be of *head-to-tail type* if $d_i(t) = d_{i+1}(t)$, and of *head-to-head*

type if instead $d_i(t) = -d_{i+1}(t)$. The ith processor state is updated according to:

$$v_i(t^+) := \begin{cases} \frac{1}{2}\big(v_i(t) + v_{i-1}(t)\big), & \text{if the impact occurs with } i-1, \\ \frac{1}{2}\big(v_i(t) + v_{i+1}(t)\big), & \text{if the impact occurs with } i+1, \end{cases} \tag{5.1}$$

$$d_i(t^+) := \begin{cases} -d_i(t), & \text{if the impact is head-to-head type,} \\ d_i(t), & \text{otherwise,} \end{cases} \tag{5.2}$$

$$a_i(t^+) := a_i(t),$$

$$\ell_i(t^+) := \begin{cases} C_i(t) - \frac{1}{2}\operatorname{dist}_{\mathsf{cc}}(C_{i-1}(t), C_i(t)), & \text{if the impact occurs with } i-1, \\ \ell_i(t), & \text{if the impact occurs with } i+1, \end{cases} \tag{5.3}$$

$$u_i(t^+) := \begin{cases} u_i(t), & \text{if the impact occurs with } i-1, \\ C_i(t) + \frac{1}{2}\operatorname{dist}_{\mathsf{cc}}(C_i(t), C_{i+1}(t)), & \text{if the impact occurs with } i+1, \end{cases} \tag{5.4}$$

where the upper-script + indicates the variable value right after the impact, and where we define the *center* $C_i \in \mathbb{S}^1$ of the desired sweeping arc $\mathcal{D}_i$ by $C_i = \ell_i + \operatorname{dist}_{\mathsf{cc}}(\ell_i, u_i)/2$.

Note that, after an impact between beads i and $i-1$, we have $\ell_{i-1}(t^+) = u_i(t^+)$ because they both are defined as the midpoint of the arc from $C_{i-1}(t)$ to $C_i(t)$.

(Crossing Event) The memory of each bead i is updated also when the agent crosses either $\ell_i(t)$ or $u_i(t)$ while leaving its desired sweeping arc. The nominal speed v_i, the direction d_i and the boundary of the sweeping arc ℓ_i and u_i do not change,

$$v_i(t^+) := v_i(t), \quad d_i(t^+) := d_i(t), \quad \ell_i(t^+) := \ell_i(t), \quad u_i(t^+) := u_i(t),$$

The flag a_i is updated as follows:

$$a_i(t^+) := d_i(t).$$

Here the upper-script + indicates the value of the memory right after bead i crosses the boundary of its desired sweeping arc.

Only two-way impacts have been considered in the above algorithm. However, impacts between three or more beads can be assumed to be a sequence of two-way impacts separated by infinitesimal times. By default, impacts between beads with smaller indices can be addressed first. Although the order in which they are addressed affects the subsequent motion of the beads, it does not affect the convergence results of the SIS Algorithm.

5.5 Preliminary results

In this section we prove some preliminary results before we can prove the correctness of the SIS Algorithm. We begin with an important characterization of initial states.

Definition 43 (Admissible balanced and unbalanced configurations) *A state* $\{(\theta_i, x_i)\}_{i\in\{1,\dots,n\}}$ *is*

(1) directionally balanced *if* $\sum_{i=1}^{n} d_i = 0$

(2) directionally D-unbalanced *for* $D \in \{-n+1, \dots, n-1\} \setminus \{0\}$*, if* $\sum_{i=1}^{n} d_i = D$.

Furthermore, a state has an admissible configuration *if in addition to being* directionally balanced *or* D-unbalanced*, for all* $i, j \in \{1, \dots, n\}$ *and* $j \neq i$, $\theta_i \neq \theta_j$. *The set of* admissible balanced configurations*, and* admissible D-unbalanced configurations *are denoted by* $\mathcal{A}_{0-\text{bal}}$*, and* $\mathcal{A}_{D-\text{unbal}}$ *respectively.*

Note that $\{(\theta_i, x_i)\}_{i\in\{1,\dots,n\}} \in \mathcal{A}_{0-\text{bal}}$ if and only if n is even and $n/2$ beads are moving clockwise and $n/2$ are moving counterclockwise.

Next, at each time $t \geq 0$, we define the *impact graph* $\mathcal{G}(t)$ as the undirected graph with vertex set $\{1, \ldots, n\}$ and with edge set defined by the following rule: the pair (i, j) is an edge in $\mathcal{G}(t)$ if the beads i and j collide at time t.

Proposition 1 (Uniform connectivity of impact graphs) *Along the trajectories of the* SIS ALGORITHM, *with* $\{(\theta_i(0), x_i(0))\}_{i\in\{1,\ldots,n\}} \in \mathcal{A}_{0-\text{bal}} \bigcup \mathcal{A}_{D-\text{unbal}}$, *for all* $t_0 \geq 0$ *the graph* $\bigcup_{t\in[t_0, t_0+2\pi/(f v_{\min})]} \mathcal{G}(t)$ *is connected.*

The proof of Proposition 1 builds up on the following facts.

Lemma 44 (Properties) *Along the trajectories of the* SIS ALGORITHM, *with* $\{(\theta_i(0), x_i(0))\}_{i\in\{1,\ldots,n\}} \in \mathcal{A}_{0-\text{bal}} \bigcup \mathcal{A}_{D-\text{unbal}}$*:*

(1) $\sum_{i=1}^{n} d_i(t)$ *is constant,*

(2) any two desired sweeping arcs are disjoint or share at most a boundary point, furthermore their label index increases in the counterclockwise direction, i.e., $u_i(t) = \ell_{i+1}(t)$,

(3) the order of the beads is preserved, i.e., for all $i, j \in \{1, \ldots, n\}$, $j \notin \{i, i+1\}$, *and* $t \geq 0$, *we have* $\text{dist}_{\mathsf{cc}}(\theta_i(t), \theta_{i+1}(t)) \leq \text{dist}_{\mathsf{cc}}(\theta_i(t), \theta_j(t))$. *Therefore, a bead* i *can impact only its immediate neighbors* $i-1$ *and* $i+1$.

Proof: See Appendix ■

Lemma 45 (Impacts in bounded interval) *Let* $v_{\min} = \min_{i\in\{1,\ldots,n\}} v_i(0)$. *Along the trajectories of the* SIS ALGORITHM, *with* $\{(\theta_i(0), x_i(0))\}_{i\in\{1,\ldots,n\}} \in \mathcal{A}_{0-\text{bal}} \bigcup \mathcal{A}_{D-\text{unbal}}$, *for all* $i \in \{1, \ldots, n\}$ *and for all* $t_0 > 0$, *bead* i *impacts at least once with both its neighbors* $i-1$ *and* $i+1$ *across the interval* $[t_0, t_0 + \frac{2\pi}{f v_{\min}}]$.

Proof: See Appendix ■

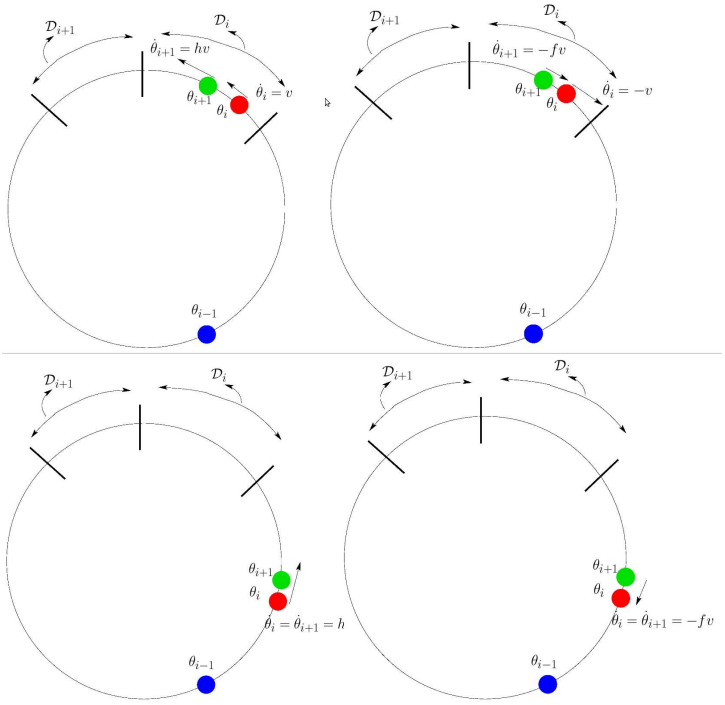

Figure 5.2: This figure shows that, regardless from where and with which velocities beads i and $i+1$ impact, the order of the beads is preserved. The velocities in the figure are the velocities after the impact. The speed v is just the average value of v_i and v_{i+1} before the impact.

Proof: [of Proposition 1] Because of Lemma 45, for all i and for all t_0 there exist t_1 and $t_2 \in [t_0, t_0 + \frac{2\pi}{f v_{\min}}]$ such that $\mathcal{G}(t_1)$ and $\mathcal{G}(t_2)$ have respectively an edge between vertices i and $i+1$ and between vertices i and $i-1$. Therefore, the graph $\bigcup_{t\in[t_0,t_0+\frac{2\pi}{f\bar{v}(0)}]} \mathcal{G}(t)$ contains the ring graph. ■

5.6 Convergence analysis

In the first part of this section we prove that the nominal speeds v_i of all the beads will asymptotically be equal to the average of their initial values, and that the desired sweeping arc will asymptotically attain a length $2\pi/N$. In the second and third part of this section we show that SIS ALGORITHM enables the beads to reach balanced synchrony if n is even and unbalanced synchrony if n is odd. We begin our convergence analysis with a useful result that combines known facts from [40, 102, 72]. Given a symmetric stochastic matrix $F \in \mathbb{R}^{N\times N}$, its *associated graph* has node set $\{1, \ldots, N\}$ and edge set defined as follows: (i, j) is an edge if and only if $F_{ij} > 0$.

Theorem 46 (Average Consensus Dynamics) *Consider a sequence of symmetric stochastic matrices $\{F(\ell) \mid \ell \in \mathbb{Z}_{\geq 0}\} \subset \mathbb{R}^{N\times N}$ and the dynamical system*

$$x(\ell + 1) = F(\ell)x(\ell).$$

Let $G(\ell)$ be the graph associated with $F(\ell)$. Assume that

(A1) $G(\ell)$ has a self loop at each node,

(A2) Each non-zero edge weight $F_{ij}(\ell)$, including the self-loops weights $F_{ii}(\ell)$, is larger than a constant $\alpha > 0$, and

(A3) The graph $\cup_{\tau\geq\ell} G(\tau)$ is connected for all $\ell \in \mathbb{Z}_{\geq 0}$.

Then the system is said to achieve average consensus with

$$\lim_{\ell \to +\infty} x(\ell) = \Big(\frac{1}{N}\sum_{i=1}^{N} x_i(0)\Big)\mathbb{1}_N.$$

We also define some terminology associated with movement of the beads on the ring. Let the k^{th} impact between beads i and $i+1$ occur at the instant I_i^k. Let $I^k = [I_1^k, \ldots, I_n^k]^T \in \mathbb{R}^n$. Let us also define the k^{th} *passage time* P_i^k as the instant at which bead i passes by the center of its desired sweeping arc after its k^{th} but before its $(k+1)^{\text{th}}$ impact. Let $P^k = [P_1^k, \ldots, P_n^k]^T \in \mathbb{R}^n$.

5.6.1 Convergence of nominal speed and desired sweeping arc

We start by proving that all nominal speeds v_i converge to being equal to the average of their initial values.

Lemma 47 (Speed convergence) *Let $v(t) = [v_1(t), \ldots, v_n(t)]^T \in \mathbb{R}^n$. Along the trajectories of the* SIS Algorithm, *with $\{(\theta_i(0), x_i(0))\}_{i\in\{1,\ldots,n\}} \in \mathcal{A}_{0-\text{bal}} \bigcup \mathcal{A}_{D-\text{unbal}}$:*

$$\lim_{t \to +\infty} v(t) = \frac{\mathbb{1}_n^T v(0)}{n}\mathbb{1}_n.$$

Proof: For all $i \in \{1, \ldots, n\}$, define $A_i \in \mathbb{R}^{n\times n}$ by:

$$[A_i]_{lm} = \begin{cases} \frac{1}{2}, & \text{if } (l,m) \in \{(i,i),(i,i+1),(i+1,i),(i+1,i+1)\}, \\ 1, & \text{if } l = m \text{ and } l \notin \{i, i+1\}, \\ 0, & \text{otherwise}. \end{cases}$$

Because of equation (5.1),

$$v(I_i^k + \varepsilon) = A_i v(I_i^k).$$

where $I_i^k + \varepsilon$ is the time instant just after the impact. This can be extended to account for more than one two-way impacts taking place at the same instant. For example, if two separate impacts occur between beads i and $i+1$ as well as j and $j+1$ at I_i^k, then $v(I_i^k + \varepsilon) = A_i A_j v(I_i^k)$.

Therefore, the dynamics of $v(t)$ is the average consensus dynamics with matrices A_i. Proposition 1 ensures that the sequence of impact graphs at impact instants is uniformly jointly connected. Therefore, the assumptions of Theorem 46 are satisfied and we know that all velocities $v_i(t)$ converge to $\frac{1}{n}\sum_{i=1}^n v_i(0)$. ■

We now prove that the desired sweeping arcs converge asymptotically to a stationary configuration in which all sweeping arcs have length $2\pi/n$.

Lemma 48 (Convergence of desired sweeping arc) *Let $L_i(t) = \mathrm{dist}_{\mathsf{cc}}(\ell_i(t), u_i(t))$ be the length of the desired sweeping arc $\mathcal{D}_i$ for $i \in \{1, \ldots, n\}$, and $L(t) = [L_1(t), \ldots, L_n(t)]^T \in \mathbb{R}^n$. Along the trajectories of the* SIS ALGORITHM, *with $\{(\theta_i(0), x_i(0))\}_{i\in\{1,\ldots,n\}} \in \mathcal{A}_{0-\mathrm{bal}} \bigcup \mathcal{A}_{D-\mathrm{unbal}}$, the arc lengths and the arcs converge, that is,*

$$\lim_{t\to+\infty} L(t) = \frac{2\pi}{n}\mathbb{1}_n,$$

and the limits $\lim_{t\to+\infty} \ell_i(t)$ and $\lim_{t\to+\infty} u_i(t)$ exist and are finite.

Proof: For all $i \in \{1, \dots, n\}$, define $B_i \in \mathbb{R}^{n \times n}$ by:

$$[B_i]_{lm} = \begin{cases} \frac{3}{4}, & \text{if } (l,m) \in \{(i,i),(i+1,i+1)\}, \\ \frac{1}{4}, & \text{if } (l,m) \in \{(i,i+1),(i+1,i)\}, \\ 1, & \text{if } l = m \notin \{i, i+1\}, \\ 0, & \text{otherwise}. \end{cases}$$

From equations (5.3) and (5.4), an impact between i and $i+1$ at time t causes

$$\begin{aligned} L_i(I_i^k + \varepsilon) &= \frac{3}{4} L_i(I_i^k) + \frac{1}{4} L_{i+1}(I_i^k), \\ L_{i+1}(I_i^k + \varepsilon) &= \frac{1}{4} L_i(I_i^k) + \frac{3}{4} L_{i+1}(I_i^k). \end{aligned}$$

Therefore, if at time t an impact between i and $i+1$ occurs and no other impact occurs, then $L(I_i^k + \varepsilon) = B_i L(I_i^k)$. Analogously to the proof of Lemma 47, the dynamics of $L(t)$ is the average consensus dynamics with matrices B_i. Proposition 1 ensures that the sequence of impact graphs at impact instants is uniformly jointly connected. Therefore, the assumptions of Theorem 46 are satisfied and we know that all lengths $L_i(t)$ converge to $\frac{1}{n}\sum_{i=1}^n L_i(0) = \frac{2\pi}{n}$. To prove that the limits of the arc boundaries $\ell_i(t)$ and $u_i(t)$ exist and are finite, it suffices to notice that (i) at each impact the arc boundaries change by an amount proportional to the difference between arc lengths, and (ii) every exponentially decaying sequence is summable. ■

5.6.2 Balanced synchrony

We now prove that the SIS Algorithm steers the collection of beads to be in balanced synchrony for a set of initial conditions contained in $\mathcal{A}_{0-\text{bal}}$, under certain

assumptions. Although convergence to balanced synchronization is proved only locally, simulations shown in Section 5.7 suggest that indeed the set of initial conditions for which the balanced synchronization is reached is quite large and may be equal to $\mathcal{A}_{0-\text{bal}}$.

Theorem 49 (Balanced synchrony convergence) *Consider an even number n of beads with an initial condition contained in $\mathcal{A}_{0-\text{bal}}$ and executing the* SIS ALGORITHM. *Assume that*

(A4) The desired sweeping arcs for each agent are already the desired steady-state regions of equal length $2\pi/n$ and the nominal velocity of each agent has the same value $\overline{v}$. Since the SIS ALGORITHM *makes sweeping regions and nominal velocities reach these common values for any initial condition in $\mathcal{A}_{0-\text{bal}}$, we can do this without loss of generality.*

(A5) $d_{2i}(0) = -d_{2i-1}(0)$ for $i \in \{1, \ldots, n/2\}$, i.e., consecutive beads move in opposite directions.

(A6) The initial condition satisfies $|P_i^1 - P_j^1| \leq \delta_{pb}$ for i, $j \in \{1, \ldots, n\}$, where $\delta_{pb} = \frac{\pi}{n\overline{v}}\left(\frac{1+f}{f}\right)$.

Then

$$\lim_{k \to +\infty} P^k = \frac{\mathbb{1}_n^T P^k}{n} \mathbb{1}_n.$$

Proof: See Appendix. ■

5.6.3 Unbalanced synchrony

We now prove that the SIS ALGORITHM steers the collection of beads to be in unbalanced synchrony for a set of initial conditions contained entirely in $\mathcal{A}_{D-\text{unbal}}$ with $D = \pm 1$. We first start by proving that there exists an orbit along which the beads can reach unbalanced synchrony.

Theorem 50 (Existence of periodic orbit for 1-unbalanced collections: sufficiency) *Given $D \in \{-1, +1\}$, assume that $\{(\theta_i(0), x_i(0))\}_{i\in\{1,\ldots,n\}} \in \mathcal{A}_{D-\text{unbal}}$, $\frac{1}{2} < f < \frac{n}{1+n}$, and that, for $i \in \{1, \ldots, n\}$, $v_i(t) = v_i(0) = \overline{v}$, $\ell_i(t) = \ell_i(0)$, $u_i(t) = u_i(0)$ with $\ell_i(0) = u_{i-1}(0)$ and with $\text{dist}_{\mathsf{cc}}(\ell_i(0), \ell_{i+1}(0)) = \frac{2\pi}{n}$. Then*

(i) there exists a periodic orbit for the SIS ALGORITHM *in which the beads are in unbalanced synchrony with period $2\frac{2\pi}{n}\frac{1}{\overline{v}}$; and*

(ii) along this orbit each bead i impacts its neighboring bead $i-1$ at position $\ell_i(0) + D\delta$, where $\delta = \frac{2\pi}{n^2}\frac{f}{1-f} < \frac{2\pi}{n}$.

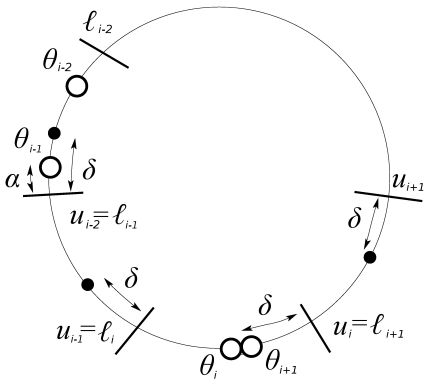

Figure 5.3: This figure shows the periodic orbit described in Theorem 50. The white circles are the positions of beads. The black dots are the locations of the impacts for any two neighboring beads. Note that bead $i-1$ and $i-2$ are moving towards each other and so are beads i and $i+1$.

Remark 51 (Impacts order in 1-unbalanced synchrony) *It is useful to take note of the order in which the impacts happen in a D-unbalanced collection of beads that reach unbalanced synchrony, where $D \in \{-1, +1\}$. As we will see in the proof of Theorem 50, if $\sum_{i=1}^{n} d_i(0) = -1$ and i and $i+1$ have just met, then the next impact will be between $i-1$ and $i-2$ and so on until i meets $i+1$ again and the periodic orbit is complete. More concisely, if the first two beads to impact are i and $i+1$, then the k^{th} impact happens between $(i-3Dk) \mod n$ and $(i+1-3Dk) \mod n$. Therefore, if $\sum_{i=1}^{n} d_i(0) = -1$, then*

the impacts happen in a counterclockwise fashion; on the other hand, if $\sum_{i=1}^{n} d_i(0) = +1$*, then the impacts happen in a clockwise fashion. Let us illustrate the idea using a the graph* $\mathcal{G}(t)$ *introduced in Proposition 1. We recall that the graph* $\mathcal{G}(t)$ *has as vertex set* $\{1, \ldots, n\}$ *and edge from* i *to* $i+1$ *if and only if the beads* i *and* $i+1$ *collide at time* t*. Figure 5.4 shows* $\mathcal{G}(t)$ *for* $t \in [I_{1,2}, I_{1,2} + 2\frac{2\pi}{n}\frac{1}{\bar{v}}]$ *and the time at which the impacts happen for* $n = 5$*.* □

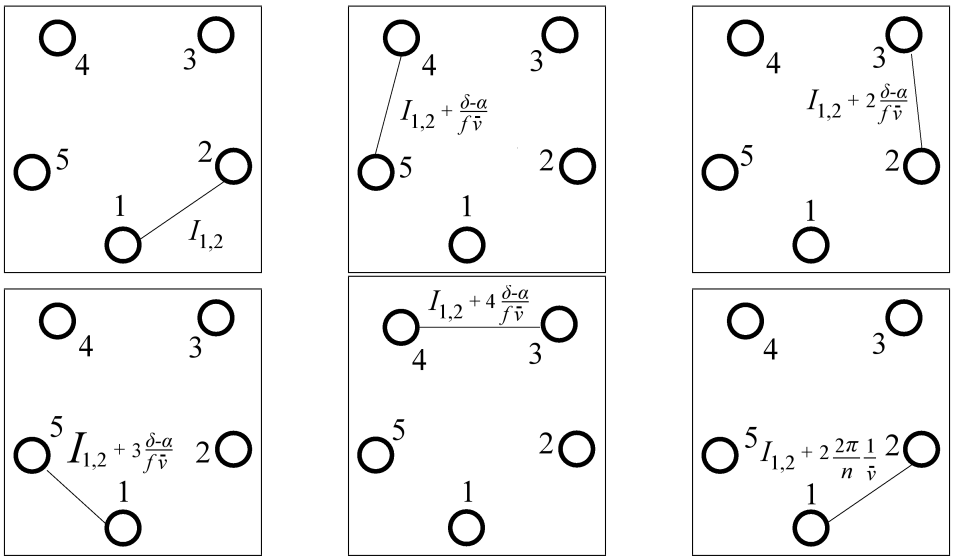

Figure 5.4: This figure illustrates $\mathcal{G}(t)$ for $t \in [I_{1,2}, I_{1,2} + 2\frac{2\pi}{n}\frac{1}{\bar{v}}]$ and the time at which each edge appears for $n = 5$ and $\sum_{i=1}^{n} d_i(0) = -1$ when unbalanced synchrony is reached.

Proof: [of Theorem 50] See Appendix. ■

It turns out that $f < \frac{n}{1+n}$ is not only sufficient but also necessary for the existence of the periodic orbit described in part (ii) of Theorem 50.

Theorem 52 (Existence of periodic orbit for 1-unbalanced collections: necessity) *Given* $D \in \{-1, +1\}$*, assume that* $\{(\theta_i(0), x_i(0))\}_{i \in \{1,\ldots,n\}} \in \mathcal{A}_{D-\text{unbal}}$*, and that, for* $i \in \{1, \ldots, N\}$*,* $v_i(t) = v_i(0) = \overline{v}$*,* $\ell_i(t) = \ell_i(0)$*,* $u_i(t) = u_i(0)$ *with* $\ell_i(0) = u_{i-1}(0)$ *and with* $\text{dist}_{\mathsf{cc}}(\ell_i(0), \ell_{i+1}(0)) = \frac{2\pi}{N}$*. If along the trajectories of the* SIS ALGORITHM *the un-*

balanced synchrony is reached, that is, beads i and $i-1$ always meet at $\ell_i(t) + D\delta$ with $\delta < \frac{2\pi}{n}$ and the period of the orbit is $2\frac{2\pi}{n}\frac{1}{\overline{v}}$, then $f < \frac{n}{1+n}$.

Proof: See Appendix. ■

A natural question to ask is if there exists a periodic orbit for the SIS ALGORITHM when $\{(\theta_i(0), x_i(0))\}_{i\in\{1,\ldots,N\}} \in \mathcal{A}_{D-\text{unbal}}$ and $|D| > 1$. To answer this question, we extend the result of Theorem 52 to the more general case of D-unbalanced collections of beads.

Theorem 53 (Existence of a periodic orbit: necessity) *Let $\{(\theta_i(0), x_i(0))\}_{i\in\{1,\ldots,n\}} \in \mathcal{A}_{D-\text{unbal}}$ and $|D| > 1$. If along the trajectories of the* SIS ALGORITHM *the unbalanced synchrony is reached and bead i meets bead $i-1$ at location $\ell_i(t) + \frac{D}{|D|}\delta$ with $\delta < \frac{2\pi}{n}$, then $f < \frac{n/|D|}{1+n/|D|}$.*

Proof: See Appendix. ■

We now prove that the SIS ALGORITHM steers the collection of beads to be in unbalanced synchrony for a set of initial conditions contained in $\mathcal{A}_{D-\text{unbal}}$, under certain assumptions.

In particular we prove that the interval between two consecutive times each bead passes by a point while moving in the same direction asymptotically approaches $2\frac{2\pi}{n}\frac{1}{\overline{v}}$, which is the period of the periodic orbit. This is just a consequence of the definition of unbalanced synchrony.

Theorem 54 (1-unbalanced synchrony convergence) *Consider n beads executing the* SIS ALGORITHM, *with n being odd. Let $\delta = \frac{2\pi}{n^2}\frac{f}{1-f} < \frac{2\pi}{n}$, and $\tilde{C}_i$ be the center of the counterclockwise arc* $\text{Arc}(\ell_i(0) + D\delta, u_i(0) + D\delta)$ *for all $i \in \{1, \ldots, n\}$. Further, assume that*

(A7) The desired sweeping arcs for each agent are already the desired steady-state regions of equal length $2\pi/n$ and the nominal velocity of each agent has the same value $\overline{v}$.

Since the initial condition is in $\mathcal{A}_{D-\text{unbal}}$, we can do this without loss of generality.

(A8) $D \in \{-1, +1\}$

(A9) The initial condition is such that $|P_i^1 - P_j^1| \leq \delta_{pub}$ *for* $i,\ j \in \{1, \ldots, n\}$ *where* $\delta_{pub} = \frac{1}{\overline{v}}(\ \delta + \frac{\pi}{n})\ (\ \frac{1-f}{f})$.

Then, along the trajectories of the SIS Algorithm*:*

$$\lim_{k \to +\infty} P^{2k} - P^{2(k-1)} = \mathbb{1}_n \frac{2}{\overline{v}} \frac{2\pi}{n},$$

that is, the collection of beads asymptotically reaches unbalanced synchrony.

Proof: See Appendix. ■

5.7 Simulations

In this section we present numerical simulations obtained by implementing the SIS Algorithm on balanced and unbalanced collection of beads. Based on the simulations we formulate four conjectures.

5.7.1 Balanced collection of beads

As we have seen in Section 5.6.2, it can be proved that the SIS Algorithm allows the beads to get in sync if for all $i \in \{1, \ldots, N\}$, $v_i(0) = \overline{v} > 0$, $\text{dist}_{\mathsf{cc}}(\ell_i(0), \ell_{i+1}(0)) = \frac{2\pi}{n}$, $\text{dist}_{\mathsf{cc}}(\ell_i(0), u_i(0)) = \frac{2\pi}{n}$, and $d_i(0) = -d_j(0)$ for $j \in \{i-1, i+1\}$. Extensive simulations

suggest that the basin of attraction of the periodic orbit is indeed much larger; we state this observation as a conjecture.

Conjecture 1 (Balanced collection: global basin of attraction) *Given initial conditions* $\{(\theta_i(0), x_i(0))\}_{i\in\{1,\dots,n\}} \in \mathcal{A}_{0-\mathrm{bal}}$*, let* P_i^k *be the last instant at which bead* i *passed by the center of its desired sweeping arc before time* t *and let* $P^k = [P_1^k, \dots, P_n^k]^T \in \mathbb{R}^n$*. Then, along the trajectories of the* SIS ALGORITHM*:*

$$\lim_{k\to+\infty} P^k = \frac{\mathbb{1}_n^T P^k}{n}\mathbb{1}_n.$$

In what follows we present the simulation results obtained by implementing the SIS ALGORITHM with $n = 8$ beads, when beads are randomly positioned on $\mathbb{S}^1$, $v_i(0)$ uniformly distributed in $]0, 1]$, $d_1(0) = d_2(0) = d_4(0) = d_6(0) = +1$ and $f = 0.7$.

Figure 5.5(a) shows the positions of the eight beads vs time. Consecutive beads do not move in opposite directions initially, as is assumed in Assumption (A5) for the validity of Theorem 49. They also do not possess same initial nominal speeds, as is necessary according to Assumption (A4). Since beads $i = 3, 4, 6$ and 7 impact neighboring beads even before they pass through the centers of their respective desired sweeping arcs after their first impacts, clearly Assumption (A6) is also not satisfied.

In spite of none of the assumptions being satisfied, each bead meets its neighbor at the same location on the circle asymptotically, reaching synchrony. The beads also attain the same nominal speed asymptotically. In Figure 5.5(b), the positions and the desired sweeping arc boundaries for bead $i = 5$ are illustrated. The solid line represents $\theta_5(t)$, the dash-dot line represents $\ell_5(t)$, and the thicker solid line represents $u_5(t)$. The distance $\mathrm{dist}_{\mathsf{cc}}(\ell_5(t), u_5(t))$ asymptotically approaches $360/N = 45$ degrees.

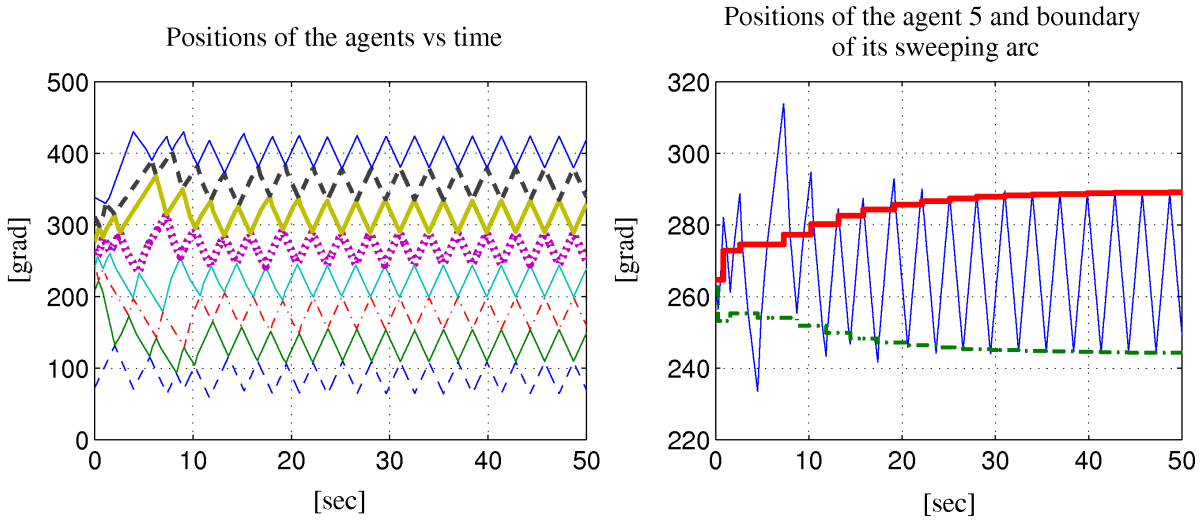

Figure 5.5: The SIS ALGORITHM is implemented with $n = 8$ beads, which are randomly positioned on $\mathbb{S}^1$, $v_i(0)$ is uniformly distributed in $]0,1]$, $d_1(0) = d_2(0) = d_4(0) = d_6(0) = +1$, and $f = 0.7$. (a) shows positions of beads vs time. Beads $2,4,6,8$ are represented by solid lines, while the dash line, dash-dot line, point line, and thicker dash line represent the positions of beads $1,3,5,7$. (b) shows $\theta_5(t)$ (solid line), $u_5(t)$ (thicker solid line), and $\ell_5(t)$ (dash-dot line).

5.7.2 Unbalanced collection of beads

In Theorem 54 we have proved that if $\{(\theta_i(0), x_i(0))\}_{i\in\{1,\dots,n\}} \in \mathcal{A}_{D-\text{unbal}}$ with $D \in \{-1,+1\}$, and if the collection of beads is close to unbalanced synchrony, then the SIS ALGORITHM steers the collection to synchrony. Also in this case, extensive simulations suggest that the basin of attraction of the periodic orbit is larger.

Conjecture 2 (1-unbalanced collection: global basin of attraction) *Given initial conditions* $\{(\theta_i(0), x_i(0))\}_{i\in\{1,\dots,n\}} \in \mathcal{A}_{D-\text{unbal}}$ *with* $D \in \{-1,+1\}$, *let* $\delta = \frac{2\pi}{n^2}\frac{f}{1-f} < \frac{2\pi}{n}$, *and let* $\tilde{C}_i(t)$ *be the center of the counterclockwise arc* $\text{Arc}(\ell_i(t) + D\delta, u_i(t) + D\delta)$ *for all* $i \in \{1,\dots,n\}$. *Let* P_i^k *be the instant at which bead* i *passes by* $\tilde{C}_i$ *for the* k^{th} *time and let* $P^k = [P_1^k, \dots, P_n^k]^T \in \mathbb{R}^n$. *Then, along the trajectories of the* SIS ALGORITHM*:*

$$\lim_{k\to+\infty} P^{2k} - P^{2(k-1)} = \mathbb{1}_n \frac{2}{\overline{v}} \frac{2\pi}{n},$$

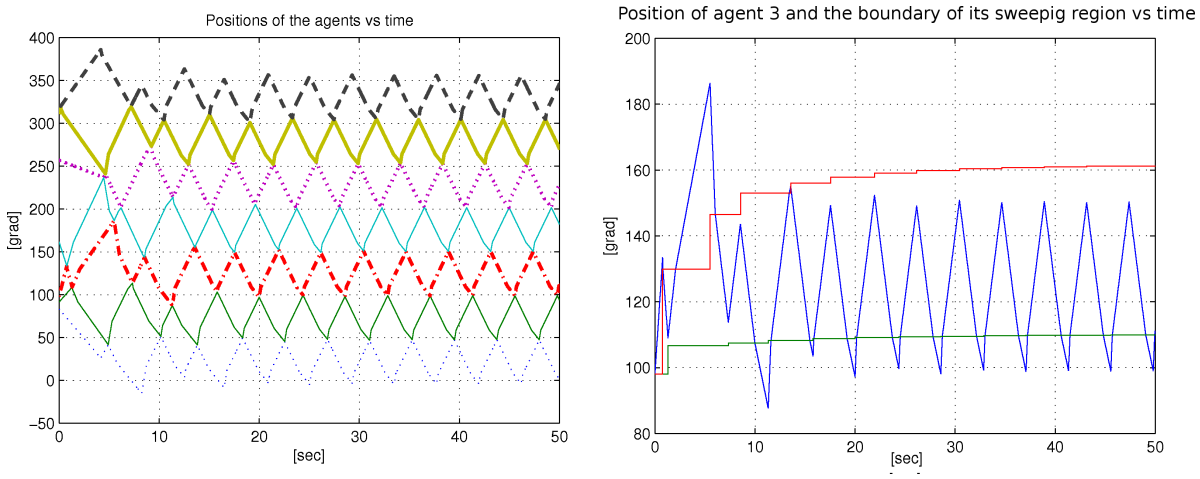

Figure 5.6: The SIS ALGORITHM is implemented for $n = 7$ beads. The beads are randomly positioned on $\mathbb{S}^1$, $v_i(0)$ is uniformly distributed in $]0, 1]$, $d_1(0) = d_4(0) = d_5(0) = d_7(0) = -1$, and $f = 0.6$. (a) shows θ_i vs time. Beads $2, 4, 6$ are represented by solid lines, while the dash line, dash-dot line, point line, and thicker dash line represent the positions of beads $1, 3, 5, 7$. (b) shows $\theta_3(t)$ (solid line), $u_3(t)$ (thicker solid line), and $\ell_3(t)$ (dash-dot line).

that is, the collection of beads asymptotically reaches unbalanced synchrony.

In what follows we present the simulation results obtained by implementing the SIS ALGORITHM with $n = 7$ beads, the beads are randomly positioned on $\mathbb{S}^1$, $v_i(0)$ uniformly distributed in $]0, 1]$, $d_1(0) = d_4(0) = d_5(0) = d_7(0) = -1$, that is the collection of beads is D-unbalanced with $D = -1$, and $f = 0.6$. Note that $f < \frac{n}{1+n} = \frac{7}{8}$. Figure 5.6(a) shows the positions of the seven beads vs time. Clearly, asymptotically each bead meets its neighbor at the same location on the circle, reaching synchrony. In Figure 5.6(b), the positions and the desired sweeping arc boundaries for bead $i = 3$ are illustrated. The solid line represents $\theta_3(t)$, the dash-dot line represents $\ell_3(t)$, and the thicker solid line represents $u_3(t)$. The distance $\text{dist}_{\text{cc}}(\ell_3(t), u_3(t))$ asymptotically approaches $360/n \approx 51.42$ degrees.

For the more general case of D-unbalanced collections with $n > |D| > 1$, Theorem 53 states that $f < \frac{n/|D|}{1+n/|D|}$ is just a necessary condition for the existence of a period orbit, along which, i and $i-1$ meet always at $\ell_i + \frac{D}{|D|}\delta$, with $\delta < \frac{2\pi}{n}$. We conjecture that (i)

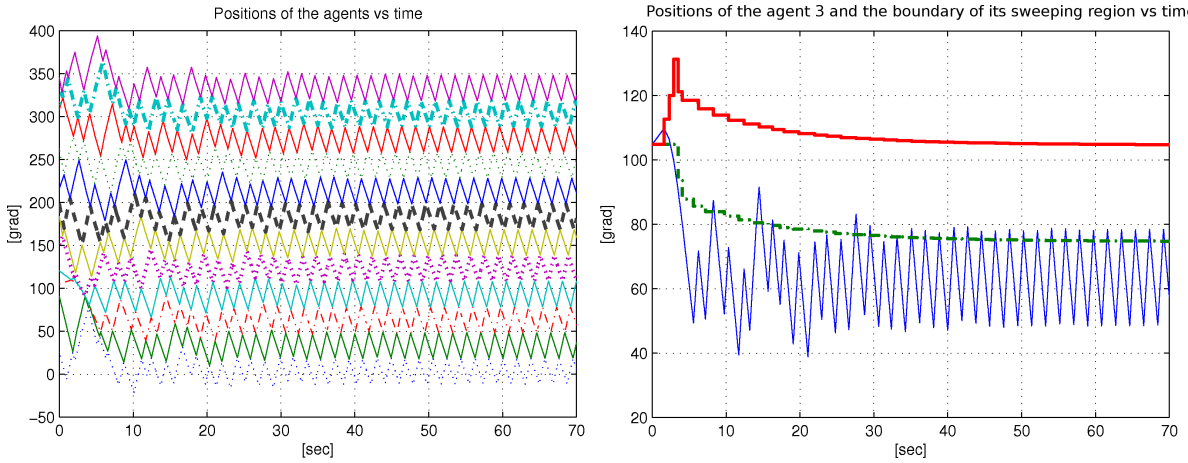

Figure 5.7: The SIS ALGORITHM is implemented for $n = 12$ beads. The beads are randomly positioned on $\mathbb{S}^1$, $v_i(0)$ is uniformly distributed in $]0, 1]$, $d_1(0) = d_2(0) = d_4(0) = d_6(0) = d_7(0) = d_9(0) = d_{12}(0) = -1$, and $f = 0.84$. (a) shows positions of the beads vs time. Beads $2, 4, 6, 8, 10, 12$ are represented by solid lines, while the dash line, dash-dot line, point line, and thicker dash line represent the positions of beads $1, 3, 5, 7, 9, 11$. (b) shows $\theta_3(t)$ (solid line), $u_3(t)$ (thicker solid line), and $\ell_3(t)$ (dash-dot line).

$f < \frac{n/|D|}{1+n/|D|}$ is also sufficient for the existence of a periodic orbit in the most general case of $|D| > 1$, and (ii) the SIS ALGORITHM steers the collection of D-unbalanced beads to synchrony.

Conjecture 3 (D-unbalanced collection: existence of periodic orbit) *Assume* $\{(\theta_i(0), x_i(0))\}_{i\in\{1,\dots,n\}} \in \mathcal{A}_{D-\text{unbal}}$, $v_i(t) = \overline{v}$, $\text{dist}_{\text{cc}}(\ell_i(0), \ell_{i+1}(0)) = \frac{2\pi}{n}$ *for all* $i \in \{1, \dots, n\}$. *The following two statements are equivalent:*

(i. $\frac{1}{2} < f < \frac{n}{1+n}$,

(ii. there exists a periodic orbit along which each bead i *impacts with its previous bead* $i-1$ *always at position* $\ell_i(0) + D\delta$, *where* $\delta = \frac{2\pi}{n^2}\frac{f}{1-f} < \frac{2\pi}{n}$.

Conjecture 4 (D-unbalanced collection: global basin of attraction) *Assume* $\{(\theta_i(0), x_i(0))\}_{i\in\{1,\dots,n\}} \in \mathcal{A}_{D-\text{unbal}}$ *with* $n > |D| > 1$, $\delta = \frac{2\pi}{n^2}\frac{f}{1-f} < \frac{2\pi}{n}$, *and let* $\tilde{C}_i(t)$ *be the center of the counterclockwise arc* $\text{Arc}(\ell_i(t)+D\delta, u_i(t)+D\delta)$ *for all* $i \in \{1, \dots, N\}$. *Let* P_i^k

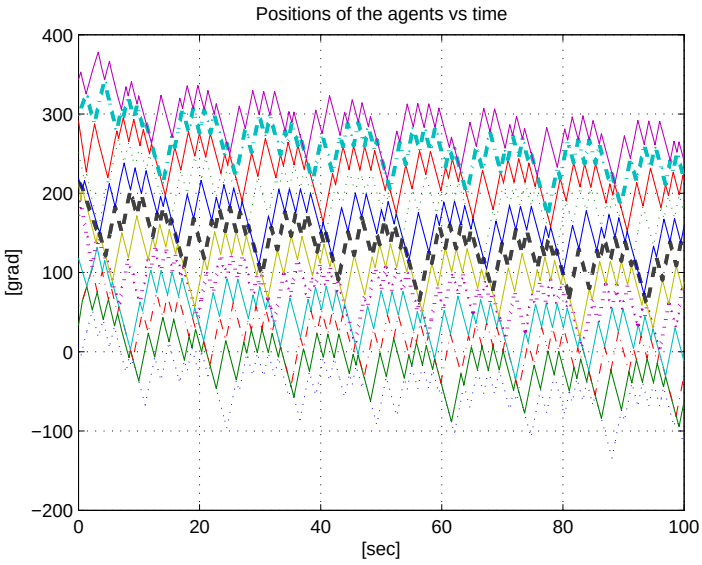

Figure 5.8: This figure shows θ_i vs time, obtained by implementing the SIS ALGORITHM with $n = 12$ beads, the beads are randomly positioned on $\mathbb{S}^1$, $v_i(0)$ uniformly distributed in $]0, 1]$, $d_1(0) = d_4(0) = d_6(0) = d_7(0) = d_8(0) = d_9(0) = d_{10}(0) = -1$, and $f = 0.87$. The positions of the beads $2, 4, 6, 8, 10, 12$ are represented by solid lines, while the dash line, dash-dot line, point line, and thicker dash line represent the positions of beads $1, 3, 5, 7, 9, 11$.

be the instant at which bead i passed by $\tilde{C}_i$ for the k^{th} time and let $P^k = [P_1^k, \dots, P_n^k]^T \in \mathbb{R}^n$. Then, along the trajectories of the SIS ALGORITHM*:*

$$\lim_{k \to +\infty} P^{2k} - P^{2(k-1)} = \mathbb{1}_n \frac{2}{\overline{v}} \frac{2\pi}{n},$$

that is, the collection of beads asymptotically reaches unbalanced synchrony.

In what follows we present the results of two simulations (Figures 5.7 and 5.8) obtained by implementing the SIS ALGORITHM with a collection of $N = 12$ beads which are D-unbalanced with $D = -2$, the beads are randomly positioned on $\mathbb{S}^1$, $v_i(0)$ uniformly distributed in $]0, 1]$. Note that according to our conjectures $f < \frac{n/|D|}{1+n/|D|} = \frac{6}{7} \approx 0.857$ has to hold in order to reach unbalanced synchrony. In the first simulation $f = 0.84$, while in the second simulation $f = 0.87$, therefore we expect to the collection of beads to be

in sync asymptotically in the first simulation but not in the second one.

Figure 5.7(a) shows the positions of the 12 beads vs time with $f = 0.84$. Clearly, asymptotically each bead meets its neighbor at the same location on the circle, reaching synchrony. In Figure 5.7(b), the positions and the desired sweeping arc boundaries for bead $i = 3$ are illustrated. The solid line represents $\theta_3(t)$, the dash-dot line represents $\ell_3(t)$, and the thicker solid line represents $u_3(t)$. The distance $\text{dist}_{\mathsf{cc}}(\ell_3(t), u_3(t))$ asymptotically approaches $360/n = 30$ degrees. Figure 5.8 shows the positions of the 12 beads vs time when $f = 0.87$. Clearly synchrony is not reached as expected.

Appendix

Proof of Lemma 44: We first prove (1). Let $\sum_{i=1}^{n} d_i(0) = D$. The only instants at which $\sum_{i=1}^{n} d_i(t)$ can change is when an impact occurs, as in equation (5.2). If the impact is of head-to-tail type, then the directions of both the beads involved do not change. On the other hand, if the impact is of head-to-head type, then the directions of the beads involved are just swapped, therefore $\sum_{i=1}^{n} d_i(t) = D$ for any $t \geq 0$.

We now prove 44(2). To initialize the algorithm, $\mathcal{D}_i(0) = \ell_i(0) = u_i(0) = \theta_i(0)$, and $\theta_i(0)$ are ordered along the counterclockwise direction. The desired sweeping arc $\mathcal{D}_i$ is updated only when the bead i is involved in an impact according to equations (5.3) and (5.4). It is elementary to show that the update equations for ℓ_i and u_i force $u_i(t^+) = \ell_{i+1}(t^+)$ and $\ell_i(t^+) = u_{i-1}(t^+)$. This clearly implies that the order of the desired sweeping arcs is never changed and that any two desired sweeping arcs can at most share a boundary.

We finally prove 44(3). The order of the beads can change only as a consequence of an impact. However, we show next that even after an impact the order of the beads is preserved. If beads i and $i+1$ are involved in an impact of head-to-head type, then after the impacts both beads change their direction so clearly $\text{dist}_{\mathsf{cc}}(\theta_{i-1}(t+s), \theta_i(t+s)) \leq$

$\text{dist}_{\text{cc}}(\theta_{i-1}(t+s), \theta_{i+1}(t+s))$, with $0 \leq s < \overline{s}$ and $t+\overline{s}$ is the time at which i impacts again. If the impact is of head-to-tail type, then the directions of the two beads does not change, but their nominal velocities $v_i(t^+)$ and $v_{i+1}(t^+)$ are equal because of equation (5.1). The impact can occur in $\mathcal{D}_i(t)$, or in $\mathcal{D}_{i+1}(t)$ or in neither, see Figure 5.2. If the impact occurs in $\mathcal{D}_i(t)$ and $d_i(t) = d_{i+1}(t) = +1$, then after the impact $\dot{\theta}_i(t^+) = v_i(t^+)$ while $\dot{\theta}_{i+1}(t^+) = hv_{i+1}(t^+)$. In fact, because of part (2), $i+1$ is moving towards its desired sweeping arc. If the impact occurs in $\mathcal{D}_i(t)$ and $d_i(t) = d_{i+1}(t) = -1$, then after the impact $\dot{\theta}_i(t^+) = -v_i(t^+)$ and $\dot{\theta}_{i+1}(t^+) = -fv_{i+1}(t^+)$ because $i+1$ is moving away from its desired sweeping arc, again because of part (2). Recalling that $f < 1$ and $h > 1$ we have that, in both cases, $\text{dist}_{\text{cc}}(\theta_{i-1}(t+s), \theta_i(t+s)) \leq \text{dist}_{\text{cc}}(\theta_{i-1}(t+s), \theta_{i+1}(t+s))$ for any time $0 \leq s < \overline{s}$. An analogous reasoning leads to the conclusion that this property holds also if the impact occurs in $\mathcal{D}_{i+1}(t)$. Now, if the impact occurs in neither $\mathcal{D}_i(t)$ nor $\mathcal{D}_{i+1}(t)$, then the beads are both moving either towards or away their desired sweeping arcs. Therefore, $\dot{\theta}_i(t^+) = \dot{\theta}_{i+1}(t^+) = hv_i(t^+)$ or $\dot{\theta}_i(t^+) = \dot{\theta}_{i+1}(t^+) = fv_i(t^+)$. Again $\text{dist}_{\text{cc}}(\theta_{i-1}(t+s), \theta_i(t+s)) \leq \text{dist}_{\text{cc}}(\theta_{i-1}(t+s), \theta_{i+1}(t+s))$ for any $0 \leq s < \overline{s}$.

Proof of Lemma 45: Note that $\min_{i\in\{1,\dots,n\}} v_i(t) \geq \min_{i\in\{1,\dots,n\}} v_i(0) = v_{\min}$ because of equation (5.1). Therefore for any $t > 0$ the lowest possible speed at which a bead can travel is $fv_{\min}$. We first show that at most after $\frac{\pi}{fv_{\min}}$ any bead has a head-to-head type impact with one of its neighbors. First, any bead i can only impact neighbors $i+1$ and $i-1$ because of Lemma 44, part (3). The necessary time for two beads i and $i+1$ to impact depends on their positions, the directions of motion and the speeds they are traveling with.

In the worst possible case at a time $t = t_0$ all the beads are clustered in a small arc of $\mathbb{S}^1$ of length ϵ, with i and $i+1$ at the opposite ends of the arc (i.e., $\text{dist}_{\text{cc}}(\theta_{i+1}(t_0), \theta_i(t_0)) = \epsilon$), $d_i(t_0) = d_{i+1}(t_0)$, and the speeds have the smallest possible value $|\dot{\theta}_i(t_0)| = |\dot{\theta}_{i+1}(t_0)| =$

$fv_{\min}$. Let us suppose $d_i(t_0) = d_{i+1}(t_0) = +1$. That is, $i+1$ is moving towards the cluster of beads and i is moving away from it. Because of Lemma 44, part (1), we have that $\sum_{i=1}^{n} d_i(t_0)| = D < n$ and this implies that $i+1$ can travel at most for $\frac{\epsilon}{2fv_{\min}}$ before having a head-to-head type impact. So at $t_1 \leq t_0 + \frac{\epsilon}{2fv_{\min}}$, $d_{i+1}(t_1) = -1$, and $\text{dist}_{\mathsf{cc}}(\theta_{i+1}(t_1), \theta_i(t_1)) \geq \epsilon$. Now, suppose that even after the impact $|\dot{\theta}_{i+1}(t_1)| = fv_{\min}$, then beads i and $i+1$ are moving towards each other and $\text{dist}_{\mathsf{cc}}(\theta_i(t_1), \theta_{i+1}(t_1)) \leq 2\pi - \epsilon$. They then meet at time $t_2 \leq t_1 + \frac{2\pi-\epsilon}{2fv_{\min}} \leq t_0 + \frac{\epsilon}{2fv_{\min}} + \frac{2\pi-\epsilon}{2fv_{\min}} = t_0 + \frac{\pi}{fv_{\min}}$.

After the impact with $i+1$, $d_i(t_2) = -1$ and, therefore, in its next head-to-head type impact bead i meets $i-1$. Following the same reasoning, we have that at most after $\frac{\pi}{fv_{\min}}$ the two beads i and $i-1$ meet. Hence across the interval $[t_0, t_0 + \frac{2\pi}{fv_{\min}}]$ any bead impacts at least once with both its neighbors.

Proof of Lemma 49: Let us suppose that at time t the beads i and $i+1$, with directions $d_i(t) = -d_{i+1}(t) = +1$, are about to collide after their k^{th} impact. According to Assumption (A4), they also have sweeping arcs that have converged and same nominal velocity $\bar{v}$. Let us assume, without any loss of generality, that the impact between beads i and $i+1$ occurs in $\mathcal{D}_{i+1}$ and precisely at $u_i + \Delta_1$ as shown in Figure 5.9.

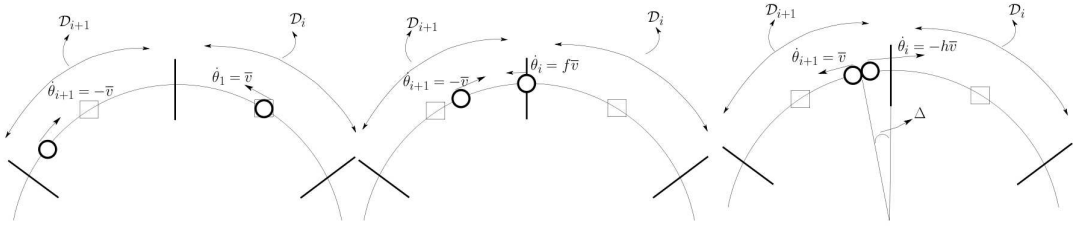

Figure 5.9: This figure shows how the speeds of bead i and $i+1$ change while they are traveling towards each other. Note that bead i is early with respect to bead $i+1$.

The time instant at which beads i and $i+1$ reach the point $u_i + \Delta_1$ simultaneously

is:

$$P_i^k + \frac{\pi}{n}\frac{1}{\overline{v}} + \frac{\Delta_1}{f\overline{v}} = P_{i+1}^k + \frac{\pi}{n}\frac{1}{\overline{v}} - \frac{\Delta_1}{\overline{v}}\,. \tag{5.5}$$

Solving (5.5) for Δ_1 we have:

$$\Delta_1 = \overline{v}\frac{f}{1+f}(P_{i+1}^k - P_i^k)\,. \tag{5.6}$$

According to Assumption (A5), beads $i-2$ and $i-1$ are also either going to or have already collided with each other. Let us assume that the impact between them occurs in $\mathcal{D}_{i-1}$ and precisely at $u_{i-2} + \Delta_2$. Following a similar analysis as done for obtaining Δ_1, one can conclude that

$$\Delta_2 = \overline{v}\frac{f}{1+f}(P_{i-1}^k - P_{i-2}^k)\,. \tag{5.7}$$

After the impact between beads i and $i+1$, the directions of both beads change because the impact is of head-to-head type, and they both head towards C_i and C_{i+1}, which they would reach at time P_i^{k+1} and P_{i+1}^{k+1} respectively. In order for the variable P_i^{k+1} to be defined, the bead i should reach the center of its sweeping arc C_i before bead $i-1$ does, after its own k^{th} impact. For this to hold true, the time taken for the former event should be smaller than or equal to the time take for the later event:

$$P_i^k + \frac{2\pi}{n}\frac{1}{\overline{v}} + \frac{\Delta_1}{\overline{v}}\left(\frac{1}{f} + \frac{1}{h}\right) \leq P_{i-1}^k + \frac{2}{\bar{v}}\left(\frac{\pi}{n} - \Delta_2\right) + \frac{\pi}{n}\frac{1}{\bar{v}}\left(\frac{1+f}{f}\right) \tag{5.8}$$

Using (5.6) and (5.7) and simplifying,

$$\left(\frac{1-f}{1+f}\right)\left(P_i^k - P_{i-1}^k\right) + \left(\frac{2f}{1+f}\right)\left(P_{i+1}^k - P_{i-2}^k\right) \leq \frac{\pi}{n\bar{v}}\left(\frac{f+1}{f}\right)$$

should hold for P_i^{k+1} to be defined. This is the case, based on Assumption (A6) and the fact that the dynamics of the passage times is average consensus, as will be proved later.

The same analysis can be carried out to prove that P_{i+1}^{k+1} is also well-defined. The choice of the impact locations $u_i + \Delta_1$ and $u_{i-2} + \Delta_2$ also accounts for the worst case scenario. Calculating P_i^{k+1} and P_{i+1}^{k+1}:

$$\begin{aligned} P_i^{k+1} &= P_i^k + \frac{2\pi}{n}\frac{1}{\overline{v}} + \frac{\Delta_1}{\overline{v}}\left(\frac{1}{f} + \frac{1}{h}\right), \\ P_{i+1}^{k+1} &= P_{i+1}^k + 2\left(\frac{\pi}{n} - \Delta_1\right)\frac{1}{\overline{v}}. \end{aligned}$$

Simplifying:

$$\begin{aligned} P_i^{k+1} &= \frac{1-f}{1+f}P_i^k + \frac{2f}{1+f}P_{i+1}^k + \frac{2\pi}{n\overline{v}}, \\ P_{i+1}^{k+1} &= \frac{2f}{1+f}P_i^k + \frac{1-f}{1+f}P_{i+1}^k + \frac{2\pi}{n\overline{v}}. \end{aligned}$$

Note that $0 < \frac{1-f}{1+f} < 1/3$ and $2/3 < \frac{2f}{1+f} < 1$ since $f \in]0.5, 1[$. Now, let us define the matrices C_{even} and $C_{\text{odd}} \in \mathbb{R}^{n\times n}$ by:

$$[C_{\text{even}}]_{lm} = \begin{cases} \frac{1-f}{1+f}, & \text{if } (l,m) \in \{(i, i+1)\}, \\ \frac{2f}{1+f}, & \text{if } (l,m) \in \{(i, i+2), (j,j)\},\ i \text{ odd}, j \text{ even} \end{cases}$$

$$[C_{\text{odd}}]_{lm} = \begin{cases} \frac{1-f}{1+f}, & \text{if } l = m, \\ \frac{2f}{1+f}, & \text{if } (l,m) \in \{(i, i+1), (i+1, i)\},\ i \text{ odd}. \end{cases}$$

Once again, we use the identification $n+1 \equiv 1$ while working with indices i and j. If the first impact after $t = 0$ is between i and $i+1$, and i is even, then the vector P^k evolves as follows:

$$P^{k+1} = \begin{cases} C_{\text{odd}}P^k + \frac{2\pi}{n\overline{v}}\mathbb{1}_n, & \text{if } k \text{ odd}, \\ C_{\text{even}}P^k + \frac{2\pi}{n\overline{v}}\mathbb{1}_n, & \text{if } k \text{ even}. \end{cases} \tag{5.9}$$

If the first impact is between i and $i+1$, and i is odd, then equation (5.9) is still valid as long as the definitions of C_{odd} and C_{even} are exchanged. In any case, the dynamics of the passage times is just the average consensus dynamics with matrices C_{odd} and C_{even}. Therefore, it can be easily proved that $\lim_{k\to+\infty} P^k = \frac{\mathbb{1}_n^T P^k}{n}\mathbb{1}_n$. Further, $\|P^k - \frac{\mathbb{1}_n^T P^k}{n}\mathbb{1}_n\|_2 \geq \|P^{k+1} - \frac{\mathbb{1}_n^T P^{k+1}}{n}\mathbb{1}_n\|_2$ and $\delta \geq \max_{i\{1,\dots,n\}} |P_i^k - P_j^k| \geq \max_{i\{1,\dots,n\}} |P_i^{k+1} - P_j^{k+1}|$. In other words, if the initial conditions of the collection of beads are close to the periodic orbit, i.e., satisfy Assumption (A6), then the resulting trajectory remains close to the periodic orbit. Furthermore, because of Proposition 1 and Theorem 46, the balanced synchrony, i.e., the consensus, is asymptotically reached.

Proof of Lemma 54: *Case (i)* Let us suppose $\delta < \frac{\pi}{n}$, and $\sum_{i=1}^n d_i(0) = -1$. According to Assumption (A7), the beads have sweeping arcs which have converged and same nominal velocity $\bar{v}$. Let us suppose that bead $i-1$ and bead i are moving towards each other and let P_{i-1}^k and P_i^k be the last time they passed by $\tilde{C}_{i-1}$ and $\tilde{C}_i$ with directions $d_{i-1} = +1$ and $d_i = -1$. If the two beads are not in unbalanced sync, they will not meet at $u_{i-1} - \delta$ but at $u_{i-1} - \delta - \Delta$, as shown in Figure 5.10.

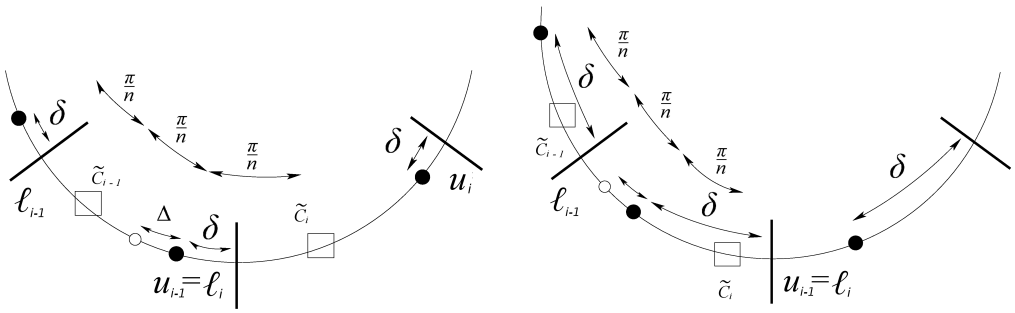

Figure 5.10: From top to bottom, the figure illustrates the position of $\tilde{C}_{i-1}$, $\tilde{C}_i$, and of $u_{i-1} - \delta - \Delta$ for $\delta < \frac{\pi}{n}$ and $\delta > \frac{\pi}{n}$.

In order to calculate where and when the beads impact we need to impose that i and

$i-1$ reach simultaneously $u_{i-1}-\delta-\Delta$:

$$P_{i-1}^k+(\frac{\pi}{n}-\Delta)\frac{1}{\overline{v}}=P_i^k+(\frac{\pi}{n}-\delta)\frac{1}{\overline{v}}+\frac{(\delta+\Delta)}{f\overline{v}}\,.$$

Note that the speeds of the beads are decided based on their location with respect to the sweeping arcs. According to Assumption (A8), these are shifted by an amount δ from the desired sweeping arcs $\mathcal{D}_i$ defined earlier. The direction of shift is determined by the sign of D. Solving for Δ we have:

$$\Delta=\frac{-f}{f+1}\overline{v}(P_i^k-P_{i-1}^k)+\frac{f-1}{f+1}\delta. \tag{5.10}$$

Note that requiring i and $i-1$ to be in unbalanced sync is equivalent to imposing $\Delta=0$ which implies $P_i^k-P_{i-1}^k=\frac{f-1}{f}\frac{\delta}{\overline{v}}$. After impacting at $u_{i-1}-\delta-\Delta$, beads $i-1$ and i change directions and head back towards $\tilde{C}_{i-1}$ and $\tilde{C}_i$, that they will reach at time P_{i-1}^{k+1} and P_i^{k+1}:

$$\begin{aligned}P_{i-1}^{k+1}&=P_{i-1}^k+2(\frac{\pi}{n}-\Delta)\frac{1}{\overline{v}}\,,\\ P_i^{k+1}&=P_i^k+2(\frac{\pi}{n}+\Delta)\frac{1}{\overline{v}}\,.\end{aligned}$$

Recalling equation (5.10) and rearranging the terms we have:

$$\begin{bmatrix}P_{i-1}^{k+1}\\P_i^{k+1}\end{bmatrix}=M\begin{bmatrix}P_{i-1}^k\\P_i^k\end{bmatrix}+\frac{2\delta}{\overline{v}}\frac{1-f}{f}\begin{bmatrix}1\\-1\end{bmatrix}+\frac{1}{\overline{v}}\frac{2\pi}{n}\begin{bmatrix}1\\1\end{bmatrix},$$

where

$$M=\begin{bmatrix}1-\frac{2f}{f+1} & \frac{2f}{f+1}\\ \frac{2f}{f+1} & 1-\frac{2f}{f+1}\end{bmatrix}. \tag{5.11}$$

Note that the dynamics matrix M is doubly stochastic since $f \in]0.5, 1[$.

Before proceeding, we note that for P_i^{k+1} to be defined, we must impose that bead i reaches $\tilde{C}_i$ before bead $i+1$ does, i.e.

$$P_i^{k+1} \leq P_{i+1}^k + (\ \frac{\pi}{n} + \delta)\ \frac{1}{\bar{v}} + (\ \frac{\pi}{n} - \delta)\ \frac{1}{f\bar{v}}$$

The term on the left hand side of this inequality is the time required for $i+1$ to reach $\tilde{C}_i$ after its k^{th} impact. This inequality can be simplified further:

$$(\ \frac{2f}{f+1})\ P_{i-1}^k + (\ \frac{1-f}{1+f})\ P_i^k - P_{i+1}^k \leq \frac{1}{\bar{v}}(\ \delta + \frac{\pi}{n})\ (\ \frac{1-f}{f})$$

This is true according to Assumption (A9), and the convergence properties of the passage times which will be proved later.

Returning back to the dynamics of the passage times, any time an impact between $i-1$ and i occurs, if Assumption (A9) is satisfied, the beads pass again by the centers of their cells at:

$$\begin{bmatrix} P_1^k \\ \vdots \\ P_{i-1}^{k+1} \\ P_i^{k+1} \\ \vdots \\ P_n^k \end{bmatrix} = E_{i-1} \begin{bmatrix} P_1^k \\ \vdots \\ P_{i-1}^k \\ P_i^k \\ \vdots \\ P_n^k \end{bmatrix} + \frac{2\delta}{\overline{v}} \frac{1-f}{f} u_{i-1} + \frac{1}{\overline{v}} \frac{2\pi}{n} w_{i-1}\,,$$

where

$$E_{i-1} = \begin{bmatrix} 1 & 0 & \dots & & & 0 \\ 0 & \ddots & & & & 0 \\ \vdots & & M_{11} & M_{12} & & \vdots \\ \vdots & & M_{21} & M_{22} & & \vdots \\ & & & & \ddots & \\ 0 & 0 & \dots & & & 1 \end{bmatrix}, \quad u_{i-1} = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ -1 \\ \vdots \\ 0 \end{bmatrix}, \quad w_{i-1} = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ 1 \\ \vdots \\ 0 \end{bmatrix},$$

and M_{ij} are the entries of the matrix M defined in equation (5.11). After any bead has met both its two neighbors once, the vector P^{k+2} can be calculated in closed form:

$$P^{k+2} = \tilde{E}P^k + \frac{2\delta}{\overline{v}}\frac{1-f}{f}\tilde{U} + \frac{1}{\overline{v}}\frac{2\pi}{n}\tilde{W}, \tag{5.12}$$

where $\tilde{E} = \prod_{m=1}^{n} E_{j_m}$, $j_m \in \{1, \dots, n\}$ (the value of j_m depends on the order of the impacts), $\tilde{U} = \sum_{r=1}^{n} \left(\prod_{m=1+r}^{n} E_{j_m}\right) u_{j_r}$, and $\tilde{W} = \sum_{r=1}^{n} \left(\prod_{m=1+r}^{n} E_{j_m}\right) w_{j_r}$.

For all $k \in \natural$ the dynamics matrix $\tilde{E}$ is actually constant because by assumption the order of the impacts is just like in Figure 5.4. Since the dynamics (5.12) is time invariant we can write the trajectory in closed-form:

$$P^{2k+1} = \tilde{E}^k P^1 + \left(\sum_{j=1}^{k-1} \tilde{E}^j\right) \left(\frac{2\delta}{\overline{v}}\frac{1-f}{f}\tilde{U} + \frac{1}{\overline{v}}\frac{2\pi}{n}\tilde{W}\right).$$

We can then calculate:

$$P^{2k+1} - P^{2(k-1)+1} = (\tilde{E}^k - \tilde{E}^{k-1})P^1 + \tilde{E}^{k-1}\left(\frac{2\delta}{\overline{v}}\frac{1-f}{f}\tilde{U} + \frac{1}{\overline{v}}\frac{2\pi}{n}\tilde{W}\right).$$

Now, note that:

$$P^{2(k+1)+1} - P^{2k+1} = (\tilde{E}^{k+1} - \tilde{E}^k)P^1 + \tilde{E}^k \left(\frac{2\delta}{\overline{v}} \frac{1-f}{f} \tilde{U} + \frac{1}{\overline{v}} \frac{2\pi}{n} \tilde{W} \right),$$

therefore we can write:

$$P^{2(k+1)+1} - P^{2k+1} = \tilde{E}(P^{2k+1} - P^{2(k-1)+1}).$$

Since $\tilde{E}$ is doubly stochastic, $\tilde{E}\mathbb{1}_n = \mathbb{1}_n$ and therefore:

$$P^{2(k+1)+1} - P^{2k+1} - \mathbb{1}_n \frac{2}{\overline{v}} \frac{2\pi}{n} = \tilde{E}\Big(P^{2k+1} - P^{2(k-1)+1} - \mathbb{1}_n \frac{2}{\overline{v}} \frac{2\pi}{n} \Big).$$

This implies that $\|(P^{2k+1} - P^{2(k-1)+1}) - \mathbb{1}_n \frac{2}{\overline{v}} \frac{2\pi}{n}\|_2 \geq \|(P^{2(k+1)+1} - P^{2k+1}) - \mathbb{1}_n \frac{2}{\overline{v}} \frac{2\pi}{n}\|_2$ and that $\max_{i\{1,\dots,n\}} |(P_i^{2k+1} - P_i^{2(k-1)+1}) - \mathbb{1}_n \frac{2}{\overline{v}} \frac{2\pi}{n}| \geq \max_{i\{1,\dots,n\}} |(P^{2(k+1)+1} - P^{2k+1}) - \mathbb{1}_n \frac{2}{\overline{v}} \frac{2\pi}{n}|$. Therefore, if the initial conditions of the collection of beads are close to the periodic orbit, then the resulting trajectory remains close to the periodic orbit. We now prove that the collection of beads asymptotically reaches unbalanced synchrony. Since $\tilde{E}$ is doubly stochastic and its associated graph is connected, $\lim_{k\to+\infty} \tilde{E}^k = \frac{\mathbb{1}_n \mathbb{1}_n^T}{n}$ (see [72]),

and therefore:

$$\begin{aligned}
\lim_{k\to+\infty} P^{2k+1} - P^{2(k-1)+1} &= \left(\frac{\mathbb{1}_n\mathbb{1}_n^T}{n} - \frac{\mathbb{1}_n\mathbb{1}_n^T}{n}\right)P^1 + \frac{\mathbb{1}_n\mathbb{1}_n^T}{n}\left(\frac{2\delta}{\overline{v}}\frac{1-f}{f}\tilde{U} + \frac{1}{\overline{v}}\frac{2\pi}{n}\tilde{W}\right) \\
&= \frac{\mathbb{1}_n\mathbb{1}_n^T}{n}\sum_{r=1}^{n}\left(\frac{2\delta}{\overline{v}}\frac{1-f}{f}\prod_{m=1+r}^{n}E_{j_m}u_{j_r} + \frac{1}{\overline{v}}\frac{2\pi}{n}\prod_{m=1+r}^{n}E_{j_m}w_{j_r}\right) \\
&= \frac{2\delta}{\overline{v}}\frac{1-f}{f}\sum_{r=1}^{n}\left(\frac{\mathbb{1}_n\mathbb{1}_n^T}{n}u_{j_r}\right) + \frac{1}{\overline{v}}\frac{2\pi}{n}\sum_{r=1}^{n}\left(\frac{\mathbb{1}_n\mathbb{1}_n^T}{n}w_{j_r}\right) \\
&= 0 + \frac{1}{\overline{v}}\frac{2\pi}{n}\sum_{r=1}^{n}2\frac{\mathbb{1}_n}{n} \\
&= \frac{2}{\overline{v}}\frac{2\pi}{n}\mathbb{1}_n\,.
\end{aligned}$$

The third equality holds because $\mathbb{1}_n^T E_{j_m} = \mathbb{1}_n^T$ for all $j_m \in \{1,\dots,n\}$ since E_{j_m} is doubly stochastic, while the fourth equality holds because $\mathbb{1}_n^T u_{j_r} = 0$ and $\mathbb{1}_n^T w_{j_r} = 2$ for all $j_r \in \{1,\dots,n\}$.

Case (ii) Let us now suppose $\delta \geq \frac{\pi}{n}$. To calculate where beads $i-1$ and i will impact we need to solve (see Figure 5.10):

$$P_{i-1}^k + (\delta - \frac{\pi}{n})\frac{1}{h\overline{v}} + (\frac{2\pi}{n} - \delta - \Delta)\frac{1}{\overline{v}} = P_i^k + (\frac{\pi}{n} + \Delta)\frac{1}{f\overline{v}},$$

solving for Δ we have:

$$\Delta = \frac{-f}{f+1}\overline{v}(P_i^k - P_{i-1}^k) + \frac{f-1}{f+1}\delta, \tag{5.13}$$

just like for case (i). After impacting at $u_{i-1} - \delta - \Delta$ beads $i-1$ and i change directions

and head back towards $\tilde{C}_{i-1}$ and $\tilde{C}_i$. We can now calculate P_{i-1}^{k+1} and P_i^{k+1}:

$$P_{i-1}^{k+1} = P_i^k + 2(\frac{\pi}{n} + \Delta)\frac{1}{\overline{v}},$$
$$P_i^{k+1} = P_i^k + 2(\frac{\pi}{n} - \Delta)\frac{1}{\overline{v}}\,.$$

The dynamics of P_{i-1} and P_i are just like in case (i), therefore the analysis and conclusion of case (i) are valid also for case (ii).

Proof of Lemma 50: We prove the theorem by constructing the periodic orbit. Without loss of generality let us suppose that $\sum_{i=1}^{n} d_i(0) = -1$. Let I_i^1 be the time at which bead i and bead $i+1$ impact at $u_i(0)-\delta \equiv \ell_{i+1}(0)-\delta$. Let us suppose that $\theta_{i-1}(I_i^1) = \ell_{i-1}(0)-\alpha$ and that $\theta_{i-2}(I_i^1)$ is such that:

$$I_{i-2}^1 = I_{i-1}^1 + \frac{\delta - \alpha}{f\overline{v}}\,, \tag{5.14}$$

with $\delta < \frac{2\pi}{n}$ and $\alpha < \delta$ (see Figure 5.3). Recalling (5.14) and by symmetry we have:

$$I_2^1 = I_1^1 + \frac{n-1}{2}\frac{\delta - \alpha}{f\overline{v}}, \tag{5.15}$$
$$I_n^1 = I_1^1 + \frac{n+1}{2}\frac{\delta - \alpha}{f\overline{v}}. \tag{5.16}$$

For beads 1 and 2 to meet again at $u_1(0) - \delta \equiv \ell_2(0) - \delta$, the following must hold:

$$I_2^1 + \left(\frac{2\pi}{n} - \delta\right)\frac{1}{\overline{v}} + \frac{\delta}{f\overline{v}} = I_n^1 + \frac{\delta}{h\overline{v}} + \left(\frac{2\pi}{n} - \delta\right)\frac{1}{\overline{v}}. \tag{5.17}$$

In fact, after impacting with bead 3, bead 2 travels along the arc $\mathrm{Arc}(\ell_2(0), u_2(0) - \delta)$ with velocity $-\overline{v}$ since it is in its desired sweeping arc. After crossing $\ell_2(0)$, the speed of bead 2 becomes $-f\overline{v}$ because it is moving away from its arc. For bead 1 the dual is

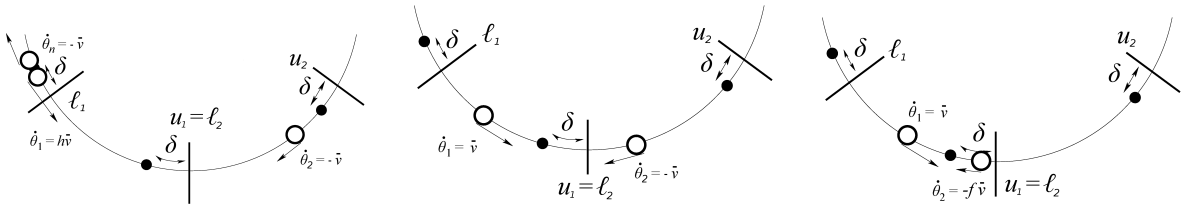

Figure 5.11: This figure shows how the speeds of bead 1 and 2 change as they are traveling towards each other, shortly after bead 1 meets bead n.

true. After impacting with bead n, bead 1 travels along the arc $\mathrm{Arc}(\ell_1(0)-\delta, \ell_1(0))$ with speed $h\overline{v}$ since it is moving towards its desired sweeping arc. After crossing $\ell_1(0)$, the speed of bead 1 becomes $\overline{v}$ because it is in its arc (see Figure 5.11).

Recalling (5.15) and (5.16), we have:

$$I_1^1 + \frac{n-1}{2}\frac{\delta-\alpha}{f\overline{v}} + \left(\frac{2\pi}{n}-\delta\right)\frac{1}{\overline{v}} + \frac{\delta}{f\overline{v}} = I_1^1 + \frac{n+1}{2}\frac{\delta-\alpha}{f\overline{v}} + \frac{\delta}{h\overline{v}} + \left(\frac{2\pi}{n}-\delta\right)\frac{1}{\overline{v}}.$$

Rearranging all the terms and solving for α:

$$\alpha = \delta(2f-1). \tag{5.18}$$

In order to be a periodic orbit we need to impose that beads 1 and 2 meet again after a period:

$$I_1^1 + \frac{n-1}{2}\frac{\delta-\alpha}{f\overline{v}} + \left(\frac{2\pi}{n}-\delta\right)\frac{1}{\overline{v}} + \frac{\delta}{f\overline{v}} = I_1^1 + 2\frac{2\pi}{n}\frac{1}{\overline{v}}. \tag{5.19}$$

Substituting (5.18) in (5.19) and solving for δ, we have:

$$\delta = \frac{2\pi}{n^2}\frac{f}{1-f}.$$

Recalling the assumption of f we have:

$$f < \frac{n}{1+n} \quad \Longrightarrow \quad \delta = \frac{2\pi}{n^2}\frac{f}{1-f} < \frac{2\pi}{n}.$$

Proof of Lemma 52: Let us assume, with no loss of generality, that $\sum_{i=1}^{n} d_i(0) = -1$. Let t^+ be the time spent by each bead traveling along the positive direction, and t^- be the time spent by each bead traveling along the negative direction in a period of the periodic orbit. In other words, if $\delta < \frac{2\pi}{n}$, then $t^- = (\frac{2\pi}{n} - \delta)\frac{1}{\overline{v}} + \frac{\delta}{f\overline{v}}$, and $t^+ = \frac{\delta}{h\overline{v}} + (\frac{2\pi}{n} - \delta)\frac{1}{\overline{v}}$, as in (5.17). Clearly $t^- + t^+ = 2\frac{2\pi}{n}\frac{1}{\overline{v}}$, which is the period of the orbit, and $t^- > t^+$, that is each bead spends more time traveling along the negative direction than along the positive. At every instant of time only one bead is unbalanced and $t^- - t^+$ is the time each bead is unbalanced during a period. By symmetry we can then conclude that $n(t^- - t^+)$ must be equal to a period:

$$2\frac{2\pi}{N}\frac{1}{\overline{v}} = n(t^- - t^+)\,. \tag{5.20}$$

Recalling the expressions for t^- and t^+, we have:

$$2\frac{2\pi}{n}\frac{1}{\overline{v}} = n2\frac{\delta}{\overline{v}}\frac{f}{1-f}\,,$$

and solving for δ

$$\delta = \frac{2\pi}{n^2}\frac{f}{1-f}\,.$$

By assumption $\delta < \frac{2\pi}{n}$, therefore:

$$\delta = \frac{2\pi}{n^2}\frac{f}{1-f} < \frac{2\pi}{n} \quad \Longrightarrow \quad f < \frac{n}{1+n}.$$

Proof of Lemma 53: The proof parallels the one of Theorem 52. Without loss of generality let us assume $\sum_{i=1}^{n} d_i(t) = D < -1$. At every instant of time $|D|$ beads are unbalanced and $t^- - t^+$ is the time each bead is unbalanced during a periodic orbit. By symmetry we can then conclude that $n\frac{(t^- - t^+)}{|D|}$ must be equal to a period, therefore equation (5.20) becomes:

$$2\frac{2\pi}{n}\frac{1}{\overline{v}} = n\frac{(t^- - t^+)}{|D|}\,,$$

where $t^- - t^+ = 2\frac{\delta}{\overline{v}}\frac{f}{1-f}$. Solving for δ we have:

$$\delta = |D|\frac{2\pi}{n^2}\frac{f}{1-f}\,.$$

Imposing the constraint $\delta < \frac{2\pi}{n}$ we can calculate the necessary condition for the existence of the periodic orbit in a D-unbalanced collection of beads:

$$f < \frac{n/|D|}{1 + n/|D|}\,.$$

Note that the higher the ratio $|D|/n$ is, the smaller f needs to be so that each bead spends enough time outside of its desired sweeping arc $\mathrm{Arc}(\ell_i(t), u_i(t))$ but it does not get too far from it.

Chapter 6

Conclusions and Future Directions

6.1 Radially Escaping Targets Problem

In Chapter 2 we introduced a novel vehicle routing problem termed the RET problem in which targets move radially outward in a disk with the intention of escaping it quickly. We established two policy independent upper bounds on the performance of any algorithm for the RET problem. We also proposed three policies for different parameter regimes of the RET problem. In Table 2.1, we also summarized the lower bounds on the capture fraction achieved by these policies as well as their factor of optimality. The SAC policy is optimal for $\lambda \to 0^+$ while for moderate arrival rates, for a fixed target speed, the SW policy is within a constant factor of the optimal. The SNB policy is within a constant factor of the optimal for $\lambda \to +\infty$. When the disk radius is greater than or equal to one, this factor is equal to 2.52.

The current problem setup can be extended in various ways. We assume that the vehicle needs to intercept the target exactly in order to capture it. An interesting and realistically motivated modification of the problem is when the vehicle has a small capture radius. The SAC policy may be extended and applied relatively easily to that setup.

On the other hand, the other policies would require extensive computation due to the history dependence which would be introduced because of the capture radius model. In the current setup, the targets move radially outward with the intention of escaping the environment in minimum time. Another modification of the setup is the case in which the targets modify their trajectories in order to evade the pursuing vehicle.

A variation of the RET problem is also the scenario in which the targets are moving radially inward towards an inner boundary instead of moving radially outward and the vehicle has to stop the targets from reaching the inner boundary. Generalizations of the RET problem, like for instance, when the distribution of targets in the environment is not uniform, or when the environment is an arbitrary closed curve and the targets have arbitrary velocities are also open to exploration.

6.2 Quickest Detection of Intruder Location

In Chapter 3 we studied the problem of how to optimally design a Markov chain which minimizes the mean first passage time to go from one region to any other region in a connected environment. We presented the first formulation of the mean first passage time for a doubly-weighted graph, which we refer to as the weighted Kemeny constant, and also provided a provably correct convex formulation for the minimization of both the Kemeny constant and the weighted Kemeny constant. Finally, we showed that both problems can be written as SDPs and, moreover, demonstrated the effectiveness of using a Markov chain with minimal mean first passage time as a surveillance policy as compared to other well-known Markov chain policies.

This work leaves open various directions for further research. First, we designed surveillance policy only for single agent systems and it would be of practical interest to consider the case where there are multiple agents: [94, 24, 5, 25] are examples of work

in this direction. Second, it would be useful to understand bounds on the design of of the mean first passage time for general graph topologies. Finally, we treat only the optimization of the transition matrix of the graph. It would be of interest to study how we can optimize the weight matrix W in conjunction with the transition matrix. This can have the interpretation of optimizing the "capacity" or "resistance" of the graph, a topic in optimization which is of independent interest [41].

6.3 Quickest Detection of Anomalies

In Chapter 4 we studied the problem of quickest detection of anomalies based on sensor observations in environments with arbitrary graph topologies. We analyzed the Ensemble CUSUM Algorithm for this surveillance task and provided guarantees on its performance. We framed an optimization problem to compute the optimal policy for the Ensemble CUSUM Algorithm. We also proposed an efficient policy which can be computed by solving a convex optimization problem. Through numerical simulations, we compared the performance of the optimal policy to the efficient policy. The detection delays guaranteed by the efficient policy were much smaller compared to alternative policies considered, especially for higher levels of uncertainties in sensor observations.

There are several possible extensions of the ideas considered here. First, the current method assume known distributions in presence and absence of anomalies in the regions under surveillance. An interesting direction is to design anomalydetection strategies that are robust to the uncertainties in these distributions. As mentioned earlier, in this case, the CUSUM algorithm can be replaced by the minimax quickest detection change algorithm [103]. Second, in the current setup, the anomalies on appearing one region, are always contained in that region. It would be of interest to consider anomalies that can move from region to region. The case of multivehicle surveillance for this setup can also

be considered. In particular, the extent of information that the vehicles can share with eachother will influence their individual routing policies. Lastly, in the Markov chain based proposed policies we relied on time-homogeneous Markov chains. A time varying Markov chain may potentially display shorter anomaly detection delays. This is also an interesting direction to be pursued.

6.4 Synchronization of beads on a ring

In Chapter 5 presented and analyzed the SIS Algorithm that synchronizes a collection of n agents or beads, moving on a ring, so that each bead patrols only a sector of the ring. The algorithm is distributed and requires that two agents exchange information only when they meet. We have established that the proposed algorithm renders locally attractive the periodic modes corresponding to balanced and unbalanced synchrony. Simulations indicate that convergence to the desired periodic modes takes places for a large set of initial conditions.

Without providing a formal analysis, we mention here a few properties of the proposed algorithm. The SIS Algorithm (1) adapts smoothly to arrival and departures of agents throughout execution time, including adapting to switches between odd and even numbers of agents, (2) handles smoothly measurement noise and control disturbances, (3) has memory requirements and message sizes independent of n, (4) is truly distributed and does not require agents to have unique identifiers, and (5) is invariant under rotations and reflections.

Furthermore, our algorithm may be implemented even on robotic agents that do not have access to their position with respect to a global reference frame on the ring, i.e., even if they do not agree upon the position of the absolute 0 angle. To be specific, assume that each agent can only measure the angular distances that it travels and that,

at communication impacts, the agent transmits its travel distance from its arc center to the impact position. Then, it is easy to see that this "relative angle" information suffices to implement the update rules of the feedback law.

One may design alternative approaches to the basic problem of steering a group of agents to balanced synchronization. An example alternative solution is described as follows: all agents could rendezvous at a common location, thereby forming a connected communication network; then they could elect a leader, agree upon an open-loop plan, and implement it without any further communication. This approach is philosophically and practically very different from our proposed algorithm. We leave a detailed comparison to future works.

Bibliography

[1] M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions: With Formulas, Graphs, and Mathematical Tables*. Courier Dover Publications, 1972.

[2] N. Agmon, S. Kraus, and G. A. Kaminka. Multi-robot perimeter patrol in adversarial settings. In *IEEE Int. Conf. on Robotics and Automation*, pages 2339–2345, Pasadena, CA, May 2008.

[3] A. Albert. Conditions for positive and nonnegative definiteness in terms of pseudoinverses. *SIAM Journal on Applied Mathematics*, 17(2):434–440, 1969.

[4] A. Almeida, G. Ramalho, H. Santana, P. Tedesco, T. Menezes, V. Corruble, and Y. Chevaleyre. Recent advances on multi-agent patrolling. In *Advances in Artificial Intelligence*, volume 3171 of *Lecture Notes in Computer Science*, pages 474–483. Springer, 2004.

[5] D. A. Anisi, P. Ögren, and X. Hu. Cooperative minimum time surveillance with multiple ground vehicles. *IEEE Transactions on Automatic Control*, 55(12):2679–2691, 2010.

[6] Y. Asahiro, E. Miyano, and S. Shimoirisa. Grasp and delivery for moving objects on broken lines. *Theory of Computing Systems*, 42(3):289–305, 2008.

[7] A. Attanasio, J.F. Cordeau, G. Ghiani, and G. Laporte. Parallel tabu search heuristics for the dynamic multi-vehicle dial-a-ride problem. *Parallel Computing*, 30(3):377–387, 2004.

[8] E. Bakolas and P. Tsiotras. Optimal pursuit of moving targets using dynamic Voronoi diagrams. In *IEEE Conf. on Decision and Control*, pages 7431–7436, December 2010.

[9] E. Bakshy, I. Rosenn, C. Marlow, and L. Adamic. The role of social networks in information diffusion. In *Int. Conference on World Wide Web*, pages 519–528, Lyon, France, 2012.

[10] A. Beaudry, G. Laporte, T. Melo, and S. Nickel. Dynamic transportation of patients in hospitals. *OR spectrum*, 32(1):77–107, 2010.

[11] D. J. Bertsimas and G. J. van Ryzin. A stochastic and dynamic vehicle routing problem in the Euclidean plane. *Operations Research*, 39(4):601–615, 1991.

[12] D. J. Bertsimas and G. J. van Ryzin. Stochastic and dynamic vehicle routing in the Euclidean plane with multiple capacitated vehicles. *Operations Research*, 41(1):60–76, 1993.

[13] D. J. Bertsimas and G. J. van Ryzin. Stochastic and dynamic vehicle routing with general interarrival and service time distributions. *Advances in Applied Probability*, 25:947–978, 1993.

[14] T. H. Blackwell and J. S. Kaufman. Response time effectiveness: Comparison of response time and survival in an urban emergency medical services system. *Academic Emergency Medicine*, 9(4):288–295, 2002.

[15] S. D. Bopardikar, S. L. Smith, and F. Bullo. On dynamic vehicle routing with time constraints. *IEEE Transactions on Robotics*, 30(6):1524–1532, 2014.

[16] S. D. Bopardikar, S. L. Smith, F. Bullo, and J. P. Hespanha. Dynamic vehicle routing for translating demands: Stability analysis and receding-horizon policies. *IEEE Transactions on Automatic Control*, 55(11):2554–2569, 2010.

[17] B. Bošanský, V. Lisý, M. Jakob, and M. Pěchouček. Computing time-dependent policies for patrolling games with mobile targets. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 989–996, 2011.

[18] S. Boyd, P. Diaconis, and L. Xiao. Fastest mixing Markov chain on a graph. *SIAM Review*, 46(4):667–689, 2004.

[19] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[20] Olli Bräysy and Michel Gendreau. Vehicle routing problem with time windows, part i: Route construction and local search algorithms. *Transportation science*, 39(1):104–118, 2005.

[21] L. Breiman. *Probability*, volume 7 of *Classics in Applied Mathematics*. SIAM, 1992. Corrected reprint of the 1968 original.

[22] F. Bullo, E. Frazzoli, M. Pavone, K. Savla, and S. L. Smith. Dynamic vehicle routing for robotic systems. *Proceedings of the IEEE*, 99(9):1482–1504, 2011.

[23] W. Burgard, A. B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. Experiences with an interactive museum tour-guide robot. *Artificial intelligence*, 114(1):3–55, 1999.

[24] G. Cannata and A. Sgorbissa. A minimalist algorithm for multirobot continuous coverage. *IEEE Transactions on Robotics*, 27(2):297–312, 2011.

[25] D. W. Casbeer, D. B. Kingston, R. W. Beard, T. W. Mclain, S.-M. Li, and R. Mehra. Cooperative forest fire surveillance using a team of small unmanned air vehicles. *International Journal of Systems Sciences*, 37(6):351–360, 2006.

[26] P. Chalasani and R. Motwani. Approximating capacitated routing and delivery problems. *SIAM Journal on Computing*, 28(6):2133–2149, 1999.

[27] L. Chen and J. Leneutre. A game theoretical framework on intrusion detection in heterogeneous networks. *IEEE Transactions on Information Forensics and Security*, 4(2):165–178, June 2009.

[28] Y. Chevaleyre. Theoretical analysis of the multi-agent patrolling problem. In *IEEE/WIC/ACM Int. Conf. on Intelligent Agent Technology*, pages 302–308, Beijing, China, September 2004.

[29] J. Clark and R. Fierro. Mobile robotic sensors for perimeter detection and tracking. *ISA Transactions*, 46(1):3–13, 2007.

[30] B. Cooley and P. K. Newton. Iterated impact dynamics of N-beads on a ring. *SIAM Review*, 47(2):273–300, 2005.

[31] G.B. Dantzig and J.H. Ramser. The truck dispatching problem. *Management science*, 6(1):80–91, 1959.

[32] M. H. DeGroot. Reaching a consensus. *Journal of the American Statistical Association*, 69(345):118–121, 1974.

[33] M. Desrochers, J. Desrochers, and M. Solomon. A new optimization algorithm for the vehicle routing problem with time windows. *Operations Research*, 40(2):342–354, 1992.

[34] Jacques Desrosiers, Yvan Dumas, Marius M Solomon, and François Soumis. Time constrained routing and scheduling. *Handbooks in operations research and management science*, 8:35–139, 1995.

[35] P. Diaconis and D. Stroock. Geometric bounds for eigenvalues of Markov chains. *Annals of Applied Probability*, 1(1):36–61, 1991.

[36] D. V. Dimarogonas, E. Frazzoli, and K. H. Johansson. Distributed event-triggered control for multi-agent systems. *IEEE Transactions on Automatic Control*, 57(5):1291–1297, 2012.

[37] P. G. Doyle and J. L. Snell. *Random Walks and Electric Networks*. Mathematical Association of America, 1984.

[38] W. Ellens, F. M. Spieksma, P. Van Mieghem, A. Jamakovic, and R. E. Kooij. Effective graph resistance. *Linear Algebra and its Applications*, 435(10):2491–2506, 2011.

[39] Y. Elmaliach, A. Shiloni, and G. A. Kaminka. A realistic model of frequency-based multi-robot polyline patrolling. In *International Conference on Autonomous Agents*, pages 63–70, Estoril, Portugal, May 2008.

[40] F. Garin and L. Schenato. A survey on distributed estimation and control applications using linear consensus algorithms. In A. Bemporad, M. Heemels, and M. Johansson, editors, *Networked Control Systems*, LNCIS, pages 75–107. Springer, 2010.

[41] A. Ghosh, S. Boyd, and A. Saberi. Minimizing effective resistance of a graph. *SIAM Review*, 50(1):37–66, 2008.

[42] A. Goel and V. Gruhn. A general vehicle routing problem. *European Journal of Operational Research*, 191(3):650–660, 2008.

[43] B. Golden, S. Raghavan, and E. Wasil. *The Vehicle Routing Problem: Latest Advances and New Challenges*, volume 43 of *Operations Research/Computer Science Interfaces*. Springer, 2008.

[44] J. Grace and J. Baillieul. Stochastic strategies for autonomous robotic surveillance. In *IEEE Conf. on Decision and Control and European Control Conference*, pages 2200–2205, Seville, Spain, December 2005.

[45] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. `http://cvxr.com/cvx`, October 2014.

[46] M. Hammar and B. J. Nilsson. Approximation results for kinetic variants of TSP. *Discrete and Computational Geometry*, 27(4):635–651, 2002.

[47] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[48] C. S. Helvig, G. Robins, and A. Zelikovsky. The moving-target traveling salesman problem. *Journal of Algorithms*, 49(1):153–174, 2003.

[49] J. J. Hunter. Generalized inverses and their application to applied probability problems. *Linear Algebra and its Applications*, 45:157–198, 1982.

[50] J. J. Hunter. *Mathematical Techniques of Applied Probability*, volume 1 of *Discrete Time Models: Basic Theory*. Academic Press, 1983.

[51] J. J. Hunter. *Mathematical Techniques of Applied Probability*, volume 2 of *Discrete Time Models: Techniques and Applications*. Academic Press, 1983.

[52] J. J. Hunter. The role of Kemeny's constant in properties of Markov chains. *Communications in Statistics - Theory and Methods*, 43(7):1309–1321, 2014.

[53] R. Isaacs. *Differential Games*. John Wiley & Sons, 1965.

[54] A. Jadbabaie, J. Lin, and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988–1001, 2003.

[55] J. G. Kemeny and J. L. Snell. *Finite Markov Chains*. Springer, 1976.

[56] S. J. King and C. Weiman. Helpmate autonomous mobile robot navigation system. In *Fibers' 91, Boston, MA*, pages 190–198. International Society for Optics and Photonics, 1991.

[57] D. B. Kingston, R. W. Beard, and R. S. Holt. Decentralized perimeter surveillance using a team of UAVs. *IEEE Transactions on Robotics*, 24(6):1394–1404, 2008.

[58] S. Kirkland. Fastest expected time to mixing for a Markov chain on a directed graph. *Linear Algebra and its Applications*, 433(11-12):1988–1996, 2010.

[59] D. J. Klein and M. Randić. Resistance distance. *Journal of Mathematical Chemistry*, 12(1):81–95, 1993.

[60] L. Kleinrock. *Queueing Systems. Volume I: Theory*. John Wiley & Sons, 1975.

[61] Y. Lan. Multiple mobile robot cooperative target intercept with local coordination. In *IEEE Conf. on Decision and Control and Chinese Control Conference*, pages 145–151, December 2012.

[62] G. Laporte. The vehicle routing problem: An overview of exact and approximate algorithms. *ejor*, 59(3):345–358, 1992.

[63] J.K. Lenstra. Vehicle routing with time windows: optimization and approximation. In B. Golden and A. Assad, editors, *Vehicle Routing: Methods and Studies*, pages 65–84. Elsevier, 1988.

[64] M. Levene and G. Loizou. Kemeny's constant and the random surfer. *The American Mathematical Monthly*, 109(8):741–745, 2002.

[65] D. A. Levin, Y. Peres, and E. L. Wilmer. *Markov Chains and Mixing Times*. American Mathematical Society, 2009.

[66] W. Li and C. G. Cassandras. A cooperative receding horizon controller for multivehicle uncertain environments. *IEEE Transactions on Automatic Control*, 51(2):242–257, 2006.

[67] Y. Y. Li and L. E. Parker. Intruder detection using a wireless sensor network with an intelligent mobile robot response. In *IEEE SoutheastCon*, pages 37–42, Huntsville, AL, USA, April 2008.

[68] J. López, D. Pérez, E. Paz, and A. Santana. WatchBot: A building maintenance and surveillance system based on autonomous robots. *Robotics and Autonomous Systems*, 61(12):1559–1571, 2013.

[69] A. Marino, L. Parker, G. Antonelli, and F. Caccavale. Behavioral control for multi-robot perimeter patrol: A finite state automata approach. In *IEEE Int. Conf. on Robotics and Automation*, pages 3350–3355, Kobe, Japan, May 2009.

[70] A. Mauroy, P. Sacré, and R. J. Sepulchre. Kick synchronization versus diffusive synchronization. In *IEEE Conf. on Decision and Control*, pages 7171–7183, Maui, HI, USA, December 2012.

[71] C. D. Meyer. *Matrix Analysis and Applied Linear Algebra*. SIAM, 2001.

[72] L. Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Transactions on Automatic Control*, 50(2):169–182, 2005.

[73] C. Nowzari and J. Cortés. Self-triggered coordination of robotic networks for optimal deployment. *Automatica*, 48(6):1077–1087, 2012.

[74] R. Olfati-Saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.

[75] J. L. Palacios. Closed-form formulas for Kirchhoff index. *International Journal of Quantum Chemistry*, 81(2):135–140, 2001.

[76] F. Pasqualetti, A. Franchi, and F. Bullo. On cooperative patrolling: Optimal trajectories, complexity analysis and approximation algorithms. *IEEE Transactions on Robotics*, 28(3):592–606, 2012.

[77] A. Patcha and J. Park. A game theoretic approach to modeling intrusion detection in mobile ad hoc networks. In *Information Assurance Workshop. Proceedings from the Fifth Annual IEEE SMC*, pages 280–284, June 2004.

[78] R. Patel, P. Agharkar, and F. Bullo. Robotic surveillance and Markov chains with minimal first passage time. *IEEE Transactions on Automatic Control*, May 2014. To appear.

[79] M. Pavone, N. Bisnik, E. Frazzoli, and V. Isler. A stochastic and dynamic vehicle routing problem with time windows and customer impatience. *ACM/Springer Journal of Mobile Networks and Applications*, 14(3):350–364, 2009.

[80] M. Pavone, E. Frazzoli, and F. Bullo. Adaptive and distributed algorithms for vehicle routing in a stochastic and dynamic environment. *IEEE Transactions on Automatic Control*, 56(6):1259–1274, 2011.

[81] A. G. Percus and O. C. Martin. Finite size and dimensional dependence of the Euclidean traveling salesman problem. *Physical Review Letters*, 76(8):1188–1191, 1996.

[82] S. Phillips, R. G. Sanfelice, and R. S. Erwin. On the synchronization of two impulsive oscillators under communication constraints. In *American Control Conference*, pages 2443–2448, Montréal, Canada, June 2012.

[83] V. Pillac, M. Gendreau, C. Guéret, and A. L. Medaglia. A review of dynamic vehicle routing problems. *European Journal of Operational Research*, 225(1):1–11, 2012.

[84] H. N. Psaraftis. Dynamic vehicle routing problems. In B. Golden and A. Assad, editors, *Vehicle Routing: Methods and Studies*, pages 223–248. Elsevier (North-Holland), 1988.

[85] W. Ren, R. W. Beard, and E. M. Atkins. Information consensus in multivehicle cooperative control: Collective group behavior through local interaction. *IEEE Control Systems Magazine*, 27(2):71–82, 2007.

[86] S. M. Ross. *Applied Probability Models with Optimization Applications*. Dover Publications, 1992.

[87] T. Sak, J. Wainer, and S. Goldenstein. Probabilistic multiagent patrolling. In *Brazilian Symposium on Artificial Intelligence*, Advances in Artificial Intelligence, pages 124–133, Salvador, Brazil, 2008. Springer.

[88] D. Siegmund. *Sequential Analysis: Tests and Confidence Intervals*. Springer, 1985.

[89] R. G. Simmons, R. H. Goodwin, K. Z. Haigh, S. Koenig, J. O'Sullivan, and M. M. Veloso. Xavier: Experience with a layered robot architecture. *ACM Sigart Bulletin*, 8(1-4):22–33, 1997.

[90] S. L. Smith, S. D. Bopardikar, and F. Bullo. A dynamic boundary guarding problem with translating demands. In *IEEE Conf. on Decision and Control and Chinese Control Conference*, pages 8543–8548, Shanghai, China, December 2009.

[91] S. L. Smith and D. Rus. Multi-robot monitoring in dynamic environments with guaranteed currency of observations. In *IEEE Conf. on Decision and Control*, pages 514–521, Atlanta, GA, USA, December 2010.

[92] S. L. Smith, M. Schwager, and D. Rus. Persistent robotic tasks: Monitoring and sweeping in changing environments. *IEEE Transactions on Robotics*, 28(2):410–426, 2012.

[93] M. M. Solomon. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35(2):254–265, 1987.

[94] K. Srivastava, D. M. Stipanovič, and M. W. Spong. On a stochastic robotic surveillance problem. In *IEEE Conf. on Decision and Control*, pages 8567–8574, Shanghai, China, December 2009.

[95] V. Srivastava, F. Pasqualetti, and F. Bullo. Stochastic surveillance strategies for spatial quickest detection. *International Journal of Robotics Research*, 32(12):1438–1458, 2013.

[96] J. M. Steele. Probabilistic and worst case analyses of classical problems of combinatorial optimization in Euclidean space. *Mathematics of Operations Research*, 15(4):749–770, 1990.

[97] S. Susca, S. Martínez, and F. Bullo. Monitoring environmental boundaries with a robotic sensor network. *IEEE Transactions on Control Systems Technology*, 16(2):288–296, 2008.

[98] B.W. Thomas. Waiting strategies for anticipating service requests from known customer locations. *Transportation Science*, 41(3):319–331, 2007.

[99] S. Thrun et al. Robotic mapping: A survey, 2002.

[100] P. Toth and D. Vigo, editors. *The Vehicle Routing Problem*. Monographs on Discrete Mathematics and Applications. SIAM, 2001.

[101] I. Triandaf and I. B. Schwartz. A collective motion algorithm for tracking time-dependent boundaries. *Mathematics and Computers in Simulation*, 70(4):187–202, 2005.

[102] J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE Transactions on Automatic Control*, 31(9):803–812, 1986.

[103] J. Unnikrishnan, V. V. Veeravalli, and S. P. Meyn. Minimax robust quickest change detection. *IEEE Transactions on Information Theory*, 57(3):1604–1614, 2011.

[104] W. Wang and J.-J. E. Slotine. On partial contraction analysis for coupled nonlinear oscillators. *Biological Cybernetics*, 92(1):38–53, 2005.

[105] X. Wu and Z. Liu. How community structure influences epidemic spread in social networks. *Physica A: Statistical Mechanics and its Applications*, 387(2):623–630, 2008.

[106] P. R. Wurman, R. D'Andrea, and M. Mountz. Coordinating hundreds of cooperative, autonomous vehicles in warehouses. *AI Magazine*, 29(1):9–20, 2008.

[107] F. Zhang and N. E. Leonard. Coordinated patterns of unit speed particles on a closed curve. *Systems & Control Letters*, 56(6):397–407, 2007.