Dynamic Vehicle Routing for Robotic Networks

by

Marco Pavone

Laurea in Ingegneria Informatica, Università degli Studi di Catania (2004)
Diploma, Scuola Superiore di Catania (2005)

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

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2010

Author .....
Department of Aeronautics and Astronautics
May 26, 2010

Certified by .....
Emilio Frazzoli
Associate Professor of Aeronautics and Astronautics
Thesis Supervisor

Certified by .....
Jonathan P. How
Professor of Aeronautics and Astronautics
Thesis Committee Member

Certified by .....
John N. Tsitsiklis
Professor of Electrical Engineering
Thesis Committee Member

Accepted by .....
Eytan H. Modiano
Associate Professor of Aeronautics and Astronautics
Chair, Committee on Graduate Students

Dynamic Vehicle Routing for Robotic Networks

by

Marco Pavone

Submitted to the Department of Aeronautics and Astronautics
on May 26, 2010, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Recent years have witnessed great advancements in the sciences and technology of autonomy, robotics and networking. This dissertation develops concepts and algorithms for dynamic vehicle routing (DVR), that is, for the automatic planning of optimal multi-vehicle routes to provide service to demands (or more generally to perform tasks) that are generated over time by an exogenous process. We consider a rich variety of scenarios relevant for robotic applications. We begin by reviewing some of the approaches available to tackle DVR problems. Next, we study different multi-vehicle scenarios based on different models for demands (in particular, demands with time constraints, demands with different priority levels, and demands that must be transported from a pick-up to a delivery location). The performance criterion used in these scenarios is either the expected waiting time of the demands or the fraction of demands serviced successfully. In each specific DVR scenario we adopt a rigorous technical approach, which we call algorithmic queueing theory and which relies upon methods from queueing theory, combinatorial optimization, and stochastic geometry. Algorithmic queueing theory consists of three basics steps: 1) queueing model of the DVR problem and analysis of its structure; 2) establishment of fundamental limitations on performance, independent of algorithms; and 3) design of algorithms that are either optimal or constant-factor away from optimal.

In the second part of the dissertation, we address problems concerning the implementation of routing policies in large-scale robotic networks, such as adaptivity and decentralized computation. We first present distributed algorithms for environment partitioning, and then we apply them to devise routing policies for DVR problems that (i) are spatially distributed, scalable to large networks, and adaptive to network changes, and (ii) have remarkably good performance guarantees.

The technical approach developed in this dissertation is applicable to a wide variety of DVR problems: several possible extensions are discussed throughout the thesis.

Thesis Supervisor: Emilio Frazzoli
Title: Associate Professor of Aeronautics and Astronautics

Thesis Committee Member: Jonathan P. How
Title: Professor of Aeronautics and Astronautics

Thesis Committee Member: John N. Tsitsiklis
Title: Professor of Electrical Engineering

To my grandmother, Matilde

Acknowledgments

First and foremost, I would like to thank my advisor, Prof. Emilio Frazzoli. Emilio has been an excellent teacher and mentor, and a trustworthy friend at the same time. His dedication to research, his openness to new ideas, and his intellectual integrity will certainly have a lasting impact on my future career.

I would like to thank Prof. Francesco Bullo at University of California at Santa Barbara for his collaboration on several topics addressed in this dissertation. Francesco has been an "unofficial", yet dedicated co-advisor, who has instilled in me, among other things, appreciation for mathematical rigor. I will never forget our long discussions on Skype on the finest mathematical details of our problems. I would also like to thank Prof. Volkan Isler at University of Minnesota for many stimulating discussions and for several career tips.

My research has significantly benefited from the collaboration with Dr. Stephen L. Smith, Prof. Alessandro Arsie (now at University of Toledo), and Dr. Jaime L. Ramirez. By sharing with me their expertise in control theory, mathematics, and spacecraft dynamics, respectively, they have significantly contributed to development of my research. It was a pleasure working with them and I hope that our collaboration will continue in the future.

I would like to acknowledge Prof. Jonathan P. How and Prof. John N. Tsitsiklis for their willingness to serve on my thesis committee and for their constructive and insightful feedback. I am grateful to Prof. Dimitri P. Bertsekas for his precious advices on my work and his mentoring during my first experience as a teacher.

Among the many friends I had the pleasure to meet at MIT, Amirali Ahmadi deserves a special comment. With his unconventional thinking, bizarre business ideas, and profound sense of humor he has contributed to make these years very enjoyable; his friendship is a precious legacy of my experience at MIT. Special thanks go to Sertac Karaman, for countless discussions on every possible topic, ranging from distributed algorithms to the Turkish invasion of Sicily ("Mamma li Turchi!"), and to John Enright, who eased my introduction into the American lifestyle. I would like to thank all other friends and colleagues at MIT who made these years memorable, including: Luca F. Bertuccelli, Francesco D'Eramo, Stephanie Gil, Georgia-Evangelia Katsargyri, Megumi Matsutani, Paul K. Njoroge, Mesrob I. Ohannessian, Mitra Osqui, Sameera Ponda, Michael Rinehart, Mardavij Roozbehani, Parikshit Shah, Tom Temple and the entire ARES Group.

Special thanks go to the staff members of the Department of Aeronautics and Astronautics and of LIDS, in particular to Marie Stuppard, Doris Inslee, and Jennifer Donovan, for their constant help.

I acknowledge the National Science Foundation for financial support for my graduate work through grants #0705451 and #0705453.

I am grateful to my mother Lella and to my father Piero for their dedication and unconditional love; it has been hard to stay apart from each other. To them and their smile I owe everything I have attained.

Finally, I thank Manuela for her love and support in these rewarding, but stressful years. Thanks to her I never felt alone. This is just our first step together.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis presents a joint algorithmic and queueing approach to the design of cooperative control and task allocation strategies for networks of uninhabited vehicles and robots. The approach enables groups of robots to complete tasks in uncertain and dynamically changing environments, where new task requests are generated in real-time. Applications include surveillance and monitoring missions, as well as transportation networks and automated material handling.

As a motivating example, consider the following scenario: a sensor network is deployed in order to detect suspicious activity in a region of interest. (Alternatively, the sensor network is replaced by a a high-altitude sensor-rich aircraft loitering over the region.) In addition to the sensor network, a team of unmanned aerial vehicles (UAVs) is available and each UAV is equipped with close-range high-resolution on-board sensors. Whenever a sensor detects a potential event, a request for close-range observation by one of the UAVs is generated. In response to this request, a UAV visits the location to gather close-range information and investigate the cause of the alarm. Each request for close-range observation might include priority levels or time windows during which the inspection must occur and it might require an on-site service time. In summary, from a control algorithmic viewpoint, each time a new request arises, the UAVs need to decide which vehicle will inspect that location and along which route. Thus, the problem is to design algorithms that enable real-time task allocation and vehicle routing.

Accordingly, this thesis presents allocation and routing algorithms that typically blend ideas from receding-horizon resource allocation, distributed optimization, combinatorics and control. The key novelty in our approach is the simultaneous introduction of stochastic, combinatorial and queueing aspects in the distributed coordination of robotic networks.

1.1 Static and Dynamic Vehicle Routing

In the recent past, considerable efforts have been devoted to the problem of how to cooperatively assign and schedule demands for service that are defined over an extended geographical area [69, 89, 2, 11, 6]. In these papers, the main focus is in developing distributed algorithms that operate with knowledge about the demand locations and with limited communication between robots. However, the underlying mathematical model is *static*, in that no new demands arrive over time, and fits within the framework of the static vehicle routing problem (see [99] for a thorough introduction to this problem), whereby: (i) a team of m vehicles is required to service a set of n demands in a 2-dimensional space; (ii) each demand

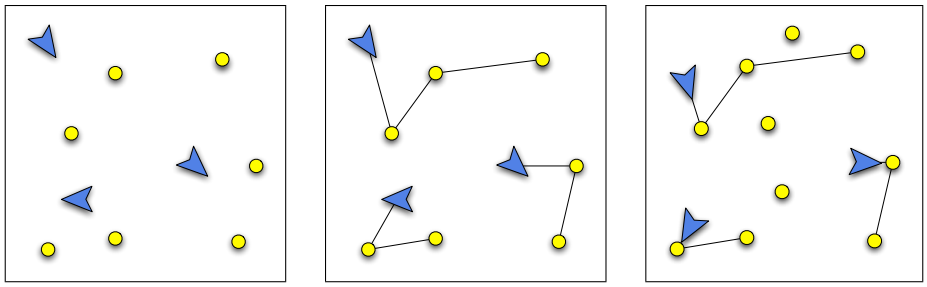


Figure 1-1: An illustration of *dynamic routing problems for a robotic system*. Panel #1: demands are generated. Panel #2: vehicles are assigned to demands and select routes. Panel #3: the DVR problem is how to recompute partitions and routes when new demands appear.

requires a certain amount of on-site service; (iii) the goal is to compute a set of routes that optimizes the cost of servicing (according to some quality of service metric) the demands. In general, most of the available literature on routing for robotic networks focuses on static environments and does not properly account for scenarios in which dynamic, stochastic and adversarial events take place.

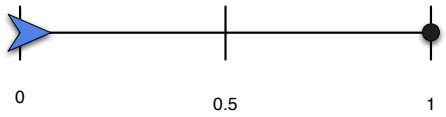
The problem of planning routes through service demands that arrive *during* a mission execution is known as the "dynamic vehicle routing problem" (abbreviated as the DVR problem in the operations research literature). There are two key differences between static and dynamic vehicle routing problems. First, planning algorithms should actually provide *policies* (in contrast to pre-planned routes) that prescribe how the routes should evolve as a function of those inputs that evolve in real-time. Second, dynamic demands (i.e., demands that arrive and vary over time) add *queueing phenomena* to the combinatorial nature of vehicle routing. In such a dynamic setting, it is natural to focus on steady-state performance instead of optimizing the performance for a single task. Additionally, system stability in terms of the number of waiting demands is an issue to be addressed.

1.2 Algorithmic Approaches to DVR Problems

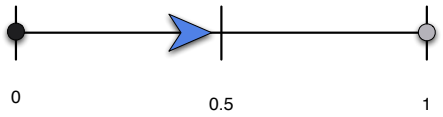
Broadly speaking, there are three main approaches available in the literature to tackle DVR problems. The first approach is to simply re-optimize every time a new event takes place; we call this approach "one-step sequential optimization". In the second approach, called "online algorithms", routing policies are designed to minimize the worst-case ratio between their performance and the performance of an optimal offline algorithm which has a priori knowledge of the entire input sequence. In the third approach, which we call "algorithmic queueing theory", the routing problem is embedded within the framework of queueing theory and routing policies are designed to minimize typical queueing-theoretical cost functions such as the expected waiting time in the system for the demands. In this section we review the three aforementioned approaches and we motivate our choice to use a queueing-theoretical framework to study DVR problems for robotic networks.

1.2.1 One-Step Sequential Optimization

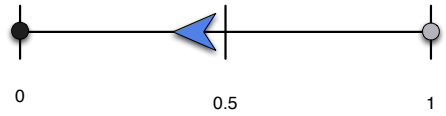
A naive approach to DVR is to re-optimize every time a new demand arrives, by using an algorithm that is optimal for the corresponding static vehicle routing problem. However,



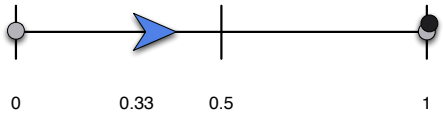
(a) A new demand arrives at $x = 1$.



(b) A new demand arrives at $x = 0$ just before the vehicle reaches $x = 0.5$.



(c) The vehicle re-optimizes its route and reverses its motion.



(d) A new demand arrives at $x = 1$ and the vehicle after re-optimizing reverses its motion

Figure 1-2: Example where re-optimization causes a vehicle to travel forever without providing service to any demand. The vehicle is represented by a blue chevron object, a newly arrived demand is represented by a black circle, and old demands are represented by grey circles.

this approach can lead to highly undesirable behaviors as the following example shows. Assume that a unit-velocity vehicle provides service along a line segment of unit length (see Figure 1-2(a)). New demands arrive either at endpoint $x = 0$ or at endpoint $x = 1$. Assume that the objective is to minimize the average waiting time of the demands (as it is common in the DVR literature); hence a re-optimization algorithm provides a route that minimizes $\sum_{j=1}^{n} W_j$, where n is the number of *outstanding* demands at that time, and W_j is the waiting time for the jth demand. Assume that at time 0 the vehicle is at $x = 0$ and a new demand arrives at $x = 1$. Hence, the vehicle travels immediately toward that demand. Assume that just before reaching $x = 1/2$ a new demand arrives at $x = 0$. It is easy to show that the optimal strategy is to reverse motion and provide service first to the demand at $x = 0$. However, assume that just before reaching $x = 1/3$ a new demand arrives at $x = 1$. It is easy to show that the optimal strategy is to reverse motion and provide service first to the demands at $x = 1$. In general, let k, n be positive integers and let $\varepsilon_k = k/(2k+1)$. Assume that just before time $t_{2n-1} = 1/2 + \sum_{k=1}^{n-1}(1 - 2\varepsilon_k)$ a new demand arrives at $x = 0$, and that just before time $t_{2n} = t_{2n-1} + 1/2 - \varepsilon_n$ a new demand arrives at $x = 1$. (Assume that at time $t_0 = 0$ the vehicle is at $x = 0$ and a new demand arrives at $x = 1$.) It is possible to show that at each new arrival the optimal strategy ensuing from a re-optimization algorithm is to reverse motion before one of the two endpoints is reached. Note that $\lim_{n\to+\infty} t_n = +\infty$, hence the vehicle will travel forever without servicing any demand!

This example therefore illustrates the pitfalls of the straightforward application of static routing and sequential re-optimization algorithms to dynamic problems. Broadly speaking, we argue that DVR problems require tailored routing algorithms with provable performance guarantees. There are currently two main algorithmic approaches that allow both a rigorous synthesis and an analysis of routing algorithms for DVR problems; we review these two approaches next.

1.2.2 Online Algorithms

An online algorithm is one that operates based on input information given up to the current time. Thus, these algorithms are designed to operate in scenarios where the entire input is not known at the outset, and new pieces of the input should be incorporated as they become available. The distinctive feature of the online algorithm approach is the method that is used to evaluate the performance of online algorithms, which is called *competitive analysis* [88]. In competitive analysis, the performance of an online algorithm is compared to the performance of a corresponding offline algorithm (i.e., an algorithm that has *a priori* knowledge of the entire input) in the worst case scenario. Specifically, an online algorithm is c-competitive if its cost on *any* problem instance is at most c times the cost of an optimal offline algorithm:

$$\text{Cost}_{\text{online}}(I) \leq c\, \text{Cost}_{\text{optimal offline}}(I), \quad \forall \text{ problem instances } I.$$

In the recent past, dynamic vehicle routing problems have been studied in this framework, under the name of the online traveling repairman problem [56, 49, 52].

While the online algorithm approach applied to DVR has led to numerous results and interesting insights, it leaves some questions unanswered, especially in the context of robotic networks. First, competitive analysis is a *worst-case* analysis, hence, the results are often overly pessimistic for normal problem instances. Moreover, in many applications there is some probabilistic problem structure (e.g., spatial distribution of future demands), that can be advantageously exploited by the vehicles. In online algorithms, this additional information is not taken into account. Second, competitive analysis is used to bound the performance relative to the optimal offline algorithm, and thus it does not give an absolute measure of performance. In other words, an optimal online algorithm is an algorithm with minimum "cost of causality" in the worst-case scenario, but not necessarily with the minimum worst-case cost. Finally, many important real-world constraints for DVR, such as time windows, priorities, and pick-up/delivery locations "have so far proved to be too complex to be considered in the online framework" [45, page 206]. Some of these drawbacks have been recently addressed by [100], where a combined stochastic and online approach is proposed for a general class of combinatorial optimization problems and is analyzed under some technical assumptions.

This discussion motivates an alternative approach for DVR in the context of robotic networks, based on probabilistic modeling, and average-case analysis.

1.2.3 Algorithmic Queueing Theory

Algorithmic queueing theory embeds the dynamic vehicle routing problem within the framework of queueing theory and overcomes most of the limitations of the online algorithm approach; in particular, it allows to take into account several real-world constraints, such as time constraints and priorities. We call this approach *algorithmic queueing theory* since its objective is to *synthesize* an efficient control policy, whereas in traditional queueing theory the objective is usually to *analyze* the performance of a specific policy. Here, an efficient policy is one whose *expected* performance is either optimal or optimal within a constant factor.[1] Algorithmic queueing theory consists of the following steps:

[1] The expected performance of a policy is the expected value of the performance over all possible inputs (i.e., demand arrival sequences). A policy performs within a constant factor κ of the optimal if the ratio between the policy's expected performance and the optimal expected performance is upper bounded by κ.

1. queueing model of the robotic system and analysis of its structure;

2. establishment of fundamental limitations on performance, independent of algorithms; and

3. design of algorithms that are either optimal or constant-factor away from optimal, possibly in specific asymptotic regimes.

Finally, the proposed algorithms are evaluated via numerical, statistical, and experimental studies, including Monte Carlo comparisons with alternative approaches.

In order to make the model tractable, demands are usually considered "statistically independent" and their arrival process is assumed stationary (with possibly unknown parameters). Because these assumptions can be unrealistic in some scenarios, this approach has its own limitations. Pioneering work in this context is that of Bertsimas and Van Ryzin [14, 15, 16], who introduced queueing methods to solve the simplest DVR problem (a vehicle moves along straight lines and visits demands whose time of arrival, location and on-site service are stochastic; information about demand location is communicated to the vehicle upon demand arrival); see also the earlier related work [84].

One of the fundamental contributions of this thesis is to show that algorithmic queueing theory, despite the aforementioned disadvantages, is a very useful framework for the design of routing algorithms for robotic networks and a valuable complement to the online algorithm approach.

1.3 Contributions of the Thesis

The objective of this thesis is to develop a joint algorithmic and queueing approach to the design of cooperative control and task allocation strategies for networks of uninhabited vehicles required to operate in dynamic and uncertain environments. By leveraging on the algorithmic queueing theory approach introduced in [14, 15, 16] and integrating ideas from dynamics, combinatorial optimization, probability theory, and distributed algorithms, we develop a systematic approach to tackle complex dynamic routing problems for robotic networks. The power of algorithmic queueing theory stems from the wide spectrum of aspects, critical to the routing of robotic networks, for which it enables a rigorous study; specific examples taken from this thesis include complex models for the demands such as time constraints, service priorities, and pick-up/delivery locations, and problems concerning robotic implementation such as adaptivity and decentralized computation.

It is important to emphasize that many of the routing policies proposed in this thesis can not be analyzed by using standard techniques in queueing theory (this is due to the fact the the travel times introduce correlations among the service times for the demands); hence in this dissertation we also introduce novel analysis techniques that merge ideas from control theory and combinatorics and that are interesting in their own right.

This thesis is divided into two parts. The first part, which includes chapters 3, 4, and 5, deals with the application of algorithmic queueing theory to DVR problems with complex demand models. The second part includes chapters 6 and 7, and deals with the study of vehicle routing policies that are specifically tailored to large-scale robotic networks. The contributions of each chapter can be summarized as follows.

Chapter 2: Preliminaries. In this chapter we first introduce some notation. Then, we review some basic results in probability theory, computational geometry, locational

optimization and combinatorics on which we will rely throughout the thesis. Finally, we review the m-vehicle Dynamic Traveling Repairman Problem, which paved the way for the algorithmic queueing theory approach.

Chapter 3: DVR with Stochastic Time Constraints. In this chapter we study time-constrained DVR problems where demands have deadlines on their waiting times. Surprisingly, little is known about time-constrained versions of DVR problems, despite their practical relevance. The purpose of this chapter is to fill this gap. Specifically, we study the following problem: m vehicles operating in a bounded environment and traveling with bounded velocity must service demands whose time of arrival, location and on-site service are stochastic; moreover, once a demand arrives, it remains active for a (possibly stochastic) amount of time, and then expires. An active demand is successfully serviced when one of the vehicles visits its location before its deadline and provides the required on-site service. The aim is to find the minimum number of vehicles needed to ensure that the long-time fraction of demands that are successfully serviced is larger than a desired value $\phi^d \in (0, 1)$, and to determine the policy the vehicles should execute to ensure that such objective is attained. Our contributions are threefold. First, we carefully formulate this problem by also taking into account the possible types of available information (e.g., the deadlines). In setting up the problem, we prove some ergodicity results that are interesting in their own right. Second, by using a variety of techniques from geometric probability, we establish a lower bound on the optimal number of vehicles for a given level of service quality (i.e., ϕ^d). In deriving the lower bound, we introduce a novel type of facility location problem, which we call the m-Location Problem with Impatient Customers (m-LPIC), and for which we provide some analysis and algorithms. Third, we analyze two service policies: we (i) show that one of the proposed policies is optimal in light load (i.e., when the arrival rate is small); (ii) derive an analytical upper bound on the number of vehicles needed by one of the two policies to achieve a given service quality; (iii) find that if the on-site service requirement is "negligible", the minimum number of vehicles is $O(\sqrt{\lambda})$, where λ is the arrival rate for the demands; (iv) prove that one of the proposed policies is within a small factor of the optimal when ϕ^d is close to one, the system is in heavy load (i.e., the arrival rate is large), and the deadlines are deterministic.

Chapter 4: DVR with Priority Classes. In this chapter we study a DVR problem in which there are multiple priority classes of service demands. Demands belonging to multiple priority classes arrive in the environment randomly over time and require a random amount of on-site service that is characteristic of the class. To service a demand, one of m vehicles must travel to the demand location and remain there for the required on-site service time. The quality of service provided to each class is given by the expected delay between the arrival of a demand in the class, and that demand's service completion. The goal is to design a routing policy for the service vehicles which minimizes a convex combination of the delays for each class. This problem has important applications in areas such as UAV surveillance, where targets are given different priority levels based on their urgency or potential importance [11]. First, we derive a lower bound on the achievable values of the convex combination of delays. Second, we propose a novel policy, which we call SQ policy, in which each class of demands is served separately from the others. We show that in heavy load

the policy performs within a constant factor $2n^2$ of the lower bound, where n is the number of classes. Thus, the constant factor is independent of the number of vehicles, the arrival rates of demands, the on-site service times, and the convex combination coefficients. Finally, we present an improvement on the SQ policy in which classes of similar priority are merged together. We also perform extensive simulations and introduce an effective heuristic improvement called the tube heuristic.

Chapter 5: DVR in Transportation Systems. Transportation on demand (TOD) systems, where users generate requests for transportation from a pick-up point to a delivery point, are already very popular and are expected to increase in usage dramatically as the inconvenience of privately-owned cars in metropolitan areas becomes excessive. Routing service vehicles through customers is usually accomplished with heuristic algorithms. In this chapter we study TOD systems in the form of a unit-capacity, multiple-vehicle dynamic pick-up and delivery problem, whereby pick-up requests arrive according to a Poisson process and are randomly located according to a general probability density. Corresponding delivery locations are also randomly distributed according to a general probability density, and a number of unit-capacity vehicles must transport demands from their pick-up locations to their delivery locations. First, we derive insightful fundamental bounds on the steady-state waiting times for the demands, and then we devise constant-factor optimal dynamic routing policies that rely on the repeated solution of traveling salesman and bipartite matching problems.

Chapter 6: Spatially Distributed Algorithms for Environment Partitioning. The best previously known control policies for DVR problems rely on centralized task assignment and are not robust against changes in the environment, in particular changes in load conditions; therefore, they are of limited applicability in scenarios involving ad hoc networks of autonomous vehicles operating in a time-varying environment. In this chapter, by blending ideas from algebraic topology and control theory, we devise spatially distributed algorithms for environment partitioning that will be pivotal to design distributed routing policies for DVR problems. The application of the algorithms developed in this chapter to DVR problems is discussed in chapter 7.

The distributed partitioning algorithms we present in this chapter are indeed useful beyond their application to DVR problems, since they allow a mobile robotic network to equitably share the workload among its members in a wide variety of scenarios.

Chapter 7: Adaptive and Distributed Algorithms for DVR. In this chapter we leverage on the spatially distributed algorithms developed in chapter 6 to obtain adaptive and distributed algorithms for DVR problems, in particular for the m-vehicle Dynamic Traveling Repairman Problem (m-DTRP).

Specifically, the contributions of this chapter are as follows. First, we present a new class of unbiased policies for the 1-DTRP. In particular, we propose the Divide & Conquer (DC) policy, whose performance depends on a design parameter $r \in \mathbb{N}$. If $r \to +\infty$, the policy is (i) provably optimal both in light- and in heavy-load conditions, and (ii) adaptive with respect to changes in the load conditions and in the statistics of the on-site service requirement; if, instead, $r = 1$, the policy is (i) provably optimal in light-load conditions and within a factor 2 of the optimal in heavy-load conditions, and (ii) adaptive with respect to all problem data, in particular, and perhaps surprisingly, it does not require *any* knowledge about the demand generation process. Moreover,

by applying ideas of receding-horizon control to dynamic vehicle routing problems, we introduce the Receding Horizon (RH) policy, that also does not require any knowledge about the demand generation process; we show that the RH policy is optimal in light-load and stable in any load condition, and we heuristically argue that its performance is close to optimal in heavy-load conditions (in particular, we heuristically argue that the RH policy is the best available unbiased and adaptive policy for the 1-DTRP). Second, we show that specific partitioning policies, whereby the environment is partitioned among the vehicles and each vehicle follows a certain set of rules in its own region, are optimal in heavy-load conditions. Finally, by combining the DC policy with the spatially distributed algorithms for environment partitioning developed in chapter 6, we design a routing policy for the m-DTRP (called m-DC policy) that (i) is spatially distributed, scalable to large networks, and adaptive to network changes, (ii) is within a constant-factor of the optimal performance in heavy-load conditions (in particular, it is optimal when demands are uniformly dispersed over the environment or when the average on-site service time requirement is negligible) and stabilizes the system in *any* load condition. Here, by network changes we mean changes in the number of vehicles, in the arrival rate of demands, and in the characterization of the on-site service requirement.

Chapter 8: Conclusion In this final chapter we draw our conclusions, and present some ideas for future research.

Chapter 2

Preliminaries

In this chapter we first introduce some notation. Then, we review some basic results in probability theory, computational geometry, locational optimization and combinatorics on which we will rely throughout this thesis. Concepts that, instead, are specific to single chapters will be presented at the beginning of those chapters. Finally, we review the m-vehicle Dynamic Traveling Repairman Problem, which paved the way for the algorithmic queueing theory approach.

2.1 Notation

We let $\mathbb{N}_0$, $\mathbb{N}$, $\mathbb{R}$, $\mathbb{R}_{\geq 0}$, and $\mathbb{R}_{>0}$ denote the set of nonnegative integers, the set of positive integers, the set of real numbers, the set of nonnegative reals number, and the set of positive reals numbers, respectively. Let $\|\cdot\|$ denote the Euclidean norm. Let $\mathcal{E}$ be a compact, convex subset of $\mathbb{R}^d$, $d \in \mathbb{N}$. We denote the boundary of $\mathcal{E}$ as $\partial\mathcal{E}$ and the Lebesgue measure of $\mathcal{E}$ as $|\mathcal{E}|$. We define the diameter of $\mathcal{E}$ as: $\operatorname{diam}(\mathcal{E}) \doteq \sup\{||p-q|| \,|\, p, q \in \mathcal{E}\}$. The distance from a point x to a set $\mathcal{S}$ is defined as $\operatorname{dist}(x, \mathcal{S}) \doteq \inf_{p\in\mathcal{S}} \|x-p\|$. We define $I_m \doteq \{1, 2, \ldots, m\}$. Let $G = (g_1, \ldots, g_m) \subset \mathcal{E}^m$ denote the location of m points.

For $h, g : \mathbb{N} \to \mathbb{R}_{\geq 0}$, we say that $h \in O(g)$ (resp., $h \in \Omega(g)$) if there exist $n_0 \in \mathbb{N}$ and $K \in \mathbb{R}_{>0}$ (resp., $k \in \mathbb{R}_{>0}$) such that $h(n) \leq Kg(n)$ for all $n \geq n_0$ (resp., $h(n) \geq kg(n)$ for all $n \geq n_0$). If $h \in O(g)$ and $h \in \Omega(g)$, then we use the notation $h \in \Theta(g)$.

2.2 Some Basic Definitions and Facts in Probability Theory

If X and Y are two random variables defined on the same probability space, then X is almost surely larger than Y if and only if $\mathbb{P}[X \geq Y] = 1$; X is surely larger than Y if and only if $X(\omega) \geq Y(\omega)$ for all samples $\omega \in \Omega$, with Ω being the sample space. A sequence of random variables $\{Y_j;\, j \in \mathbb{N}_0\}$ converges almost surely to a random variable Y if and only if the event $\{\omega \in \Omega : \lim_{j\to+\infty} Y_j(\omega) = Y(\omega)\}$, where Ω is the sample space, has probability 1. For any nonnegative, real-valued random variable Y, one can show that $\mathbb{E}[Y] = \int_0^{+\infty} \mathbb{P}[Y > y]\, dy$. Suppose $h(\cdot)$ is a *convex* function and Y is a random variable, then *Jensen's inequality* states that $\mathbb{E}[h(Y)] \geqslant h(\mathbb{E}[Y])$, provided both expectations exist. Finally, if X is an integrable random variable (i.e., a random variable satisfying $\mathbb{E}[|X|] < +\infty$) and Y is any random variable, not necessarily integrable, on the same probability space, then $\mathbb{E}[X] = \mathbb{E}[\mathbb{E}[X|Y]]$.

2.3 Computational Geometry

2.3.1 Partitions

Let $\mathcal{E} \subset \mathbb{R}^d$ be a bounded, convex set. An m-partition of $\mathcal{E}$ (where $m \in \mathbb{N}$) is a collection of m closed subsets $\{\mathcal{E}_k\}_{k=1}^m$ with disjoint interiors, whose union is $\mathcal{E}$. A partition $\{\mathcal{E}_k\}_{k=1}^m$ is *convex* if each $\mathcal{E}_k$ is convex.

2.3.2 Equitable partitions

Given a measurable function $f : \mathcal{E} \to \mathbb{R}_{>0}$, an m-partition $\{\mathcal{E}_k\}_{k=1}^m$ is *equitable* with respect to f if $\int_{\mathcal{E}_k} f(x)\,dx = \int_{\mathcal{E}} f(x)\,dx/m$ for all $k \in \{1,\dots,m\}$. Similarly, given two measurable functions $f_j : \mathcal{E} \to \mathbb{R}_{>0}$, $j \in \{1,2\}$, an m-partition $\{\mathcal{E}_k\}_{k=1}^m$ is *simultaneously equitable* with respect to f_1 and f_2 if $\int_{\mathcal{E}_k} f_j(x)\,dx = \int_{\mathcal{E}} f_j(x)\,dx/m$ for all $k \in \{1,\dots,m\}$ and $j \in \{1,2\}$. Theorem 12 in [17] and Corollary 3 in [85] show that, given two measurable functions $f_j : \mathcal{E} \to \mathbb{R}_{>0}$, $j \in \{1,2\}$, there *always* exists an m-partition of $\mathcal{E}$ that is simultaneously equitable with respect to f_1 and f_2 and where the subsets $\mathcal{E}_k$ are convex.

2.3.3 Voronoi diagrams and power diagrams

We refer the reader to [72] and [48] for comprehensive treatments, respectively, of Voronoi diagrams and power diagrams, which are special types of partitions. Assume, first, that G is an ordered set of *distinct* points in $\mathcal{E}$. The *Voronoi diagram* $\mathcal{V}(G) = (V_1(G),\dots,V_m(G))$ of $\mathcal{E}$ generated by points $G = (g_1,\dots,g_m)$ is defined by

$$V_i(G) = \{x \in \mathcal{E} |\ \|x - g_i\| \le \|x - g_j\|,\ \forall j \ne i,\ j \in I_m\}. \tag{2.1}$$

We refer to G as the set of *generators* of $\mathcal{V}(G)$, and to $V_i(G)$ as the Voronoi cell or region of dominance of the ith generator. For $g_i, g_j \in G$, $i \ne j$, we define the *bisector* between g_i and g_j as $b(g_i,g_j) = \{x \in \mathcal{E} |\ \|x-g_i\| = \|x-g_j\|\}$. The face $b(g_i,g_j)$ bisects the line segment joining g_i and g_j, and this line segment is orthogonal to the face (*Perpendicular Bisector Property*). The bisector divides $\mathcal{E}$ into two convex subsets, and leads to the definition of the set $D(g_i,g_j) = \{x \in \mathcal{E} |\ \|x-g_i\| \le \|x-g_j\|\}$; we refer to $D(g_i,g_j)$ as the *dominance region* of g_i *over* g_j. Then, the Voronoi partition $\mathcal{V}(G)$ can be equivalently defined as $V_i(G) = \bigcap_{j \in I_m \setminus \{i\}} D(g_i,g_j)$. This second definition clearly shows that each Voronoi cell is a convex set. Indeed, a Voronoi diagram of $\mathcal{E}$ is a convex partition of $\mathcal{E}$ (see Figure 2-1(a)). The Voronoi diagram of an ordered set of possibly *coincident* points is not well-defined. We define

$$\Gamma_{\text{coinc}} = \{(g_1,\dots,g_m) \in \mathcal{E}^m \,|\, g_i = g_j \text{ for some } i \ne j \in \{1,\dots,m\}\}. \tag{2.2}$$

Assume, now, that each point $g_i \in G$ has assigned an individual weight $w_i \in \mathbb{R}$, $i \in I_m$; let $W = (w_1,\dots,w_m)$. We define the power distance

$$d_P(x,g_i;w_i) \doteq \|x-g_i\|^2 - w_i. \tag{2.3}$$

We refer to the pair (g_i,w_i) as a *power point*. We define

$$G_W \doteq \Big((g_1,w_1),\dots,(g_m,w_m)\Big).$$

Two power points (g_i,w_i) and (g_j,w_j) are *coincident* if $g_i = g_j$ *and* $w_i = w_j$. Assume, first,

that G_W is an ordered set of *distinct* power points. Similarly as before, the *power diagram* $\mathcal{V}(G_W) = (V_1(G_W), \ldots, V_m(G_W))$ of $\mathcal{E}$ generated by power points $\Big((g_1, w_1), \ldots, (g_m, w_m)\Big)$ is defined by

$$V_i(G_W) = \{x \in \mathcal{E} |\ \|x - g_i\|^2 - w_i \leq \|x - g_j\|^2 - w_j,\ \forall j \neq i,\ j \in I_m\}. \tag{2.4}$$

We refer to G_W as the set of *power generators* of $\mathcal{V}(G_W)$, and to $V_i(G_W)$ as the power cell or region of dominance of the ith power generator; moreover we call g_i and w_i, respectively, the position and the weight of the power generator (g_i, w_i). Notice that, when all weights are the same, the power diagram of $\mathcal{E}$ coincides with the Voronoi diagram of $\mathcal{E}$. As before, power diagrams can be defined as intersection of convex sets; thus, a power diagram is, as well, a convex partition of $\mathcal{E}$. Indeed, power diagrams are the generalized Voronoi diagrams that have the strongest similarities to the original diagrams [8]. There are some differences, though. First, a power cell might be empty. Second, g_i might not be in its power cell (see Figure 2-1(b)). Finally, the bisector of (g_i, w_i) and (g_j, w_j), $i \neq j$, is

$$b\Big((g_i, w_i), (g_j, w_j)\Big) = \{x \in \mathcal{E} |\ (g_j - g_i)^{\mathrm{T}} x = \frac{1}{2}(\|g_j\|^2 - \|g_i\|^2 + w_i - w_j)\}. \tag{2.5}$$

Hence, $b\Big((g_i, w_i), (g_j, w_j)\Big)$ is a face orthogonal to the line segment $\overline{g_i\, g_j}$ and passing through the point g_{ij}^* given by

$$g_{ij}^* = \frac{\|g_j\|^2 - \|g_i\|^2 + w_i - w_j}{2\|g_j - g_i\|^2}(g_j - g_i);$$

this last property means that, by changing weights, it is possible to arbitrarily move the bisector between the positions of two power generators, while still preserving the orthogonality constraint.

The power diagram of an ordered set of possibly *coincident* power points is not well-defined. We define

$$\Gamma_{\text{coinc}} = \Big\{G_W \in (\mathcal{E} \times \mathbb{R})^m \,|\, g_i = g_j \text{ and } w_i = w_j \text{ for some } i \neq j \in \{1, \ldots, m\}\Big\}. \tag{2.6}$$

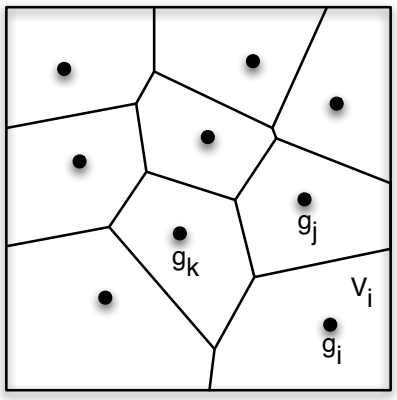
Notice that we used the same symbol as in equation (2.2): the meaning will be clear from the context.

For simplicity, we will refer to $V_i(G)$ ($V_i(G_W)$) as V_i. When the two Voronoi (power) cells V_i and V_j are adjacent (i.e., they share a face), g_i ((g_i, w_i)) is called a *Voronoi* (*power*) *neighbor* of g_j ((g_j, w_j)), and vice-versa. The set of indices of the Voronoi (power) neighbors of g_i ((g_i, w_i)) is denoted by N_i. We also define the (i, j)-face as $\Delta_{ij} \doteq V_i \cap V_j$.

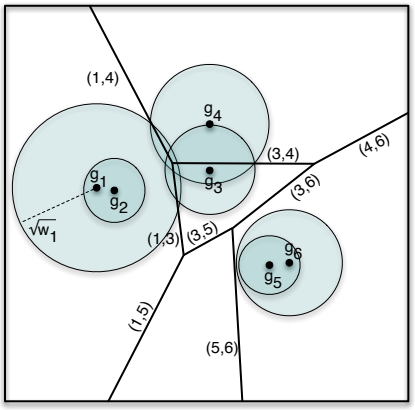
2.3.4 The continuous multi-median problem

Given a set $\mathcal{E} \subset \mathbb{R}^d$ and a vector $G = (g_1, \ldots, g_m)$ of m distinct points in $\mathcal{E}$, the expected distance between a random point x, generated according to a probability density function f, and the closest point in G is given by

$$H_m(G, \mathcal{E}) \doteq \mathbb{E}\left[\min_{k \in \{1,\ldots,m\}} \|g_k - x\|\right] = \sum_{k=1}^{m} \int_{V_k(G)} \|g_k - x\| f(x)\, dx,$$



(a) A Voronoi Diagram.



(b) A power diagram. The weights w_i are assumed positive. The radii of circles represent the magnitudes of weights. Power generator (g_2, w_2) has an empty cell. Power generator (g_5, w_5) is outside its region of dominance.

Figure 2-1: Voronoi diagrams and power diagrams.

where $\mathcal{V}(G) = (V_1(G), \ldots, V_m(G))$ is the Voronoi partition of the set $\mathcal{E}$ generated by the points G. The function H_m is known in the locational optimization literature as the continuous Weber function or the continuous multi-median function; see [1, 39] and references therein.

The m-median of the set $\mathcal{E}$, with respect to the measure induced by f, is the global minimizer

$$G_m^*(\mathcal{E}) = \arg\min_{G \in \mathcal{E}^m} H_m(G, \mathcal{E}).$$

We let $H_m^*(\mathcal{E}) = H_m(G_m^*(\mathcal{E}), \mathcal{E})$ be the global minimum of H_m. It is straightforward to show that the map $G \mapsto H_1(G, \mathcal{E})$ is differentiable and strictly convex on $\mathcal{E}$. Therefore, it is a simple computational task to compute $G_1^*(\mathcal{E})$. It is convenient to refer to $G_1^*(\mathcal{E})$ as the median of $\mathcal{E}$. On the other hand, when $m > 1$, the map $G \mapsto H_m(G, \mathcal{E})$ is differentiable (whenever $(g_1, \ldots, g_m)$ are distinct) but not convex, thus making the solution of the continuous m-median problem hard in the general case. It is known [1, 65] that the discrete version of the m-median problem is NP-hard for $d \geq 2$. The set of critical points of H_m contains all configurations $(g_1, \ldots, g_m)$ with the property that each point g_k is the generator of the Voronoi cell $V_k(G)$ as well as the median of $V_k(G)$ (we refer to such Voronoi diagrams as *median* Voronoi diagrams); in particular, the global minimum of H_m must be one of these configurations.

2.4 Combinatorics

2.4.1 The traveling salesman problem in the Euclidean plane

The Euclidean Traveling Salesman Problem (for short, TSP) is formulated as follows: given a set Q of n points in $\mathbb{R}^d$, find a minimum-length tour (i.e., a cycle that visits all nodes exactly once) of Q; the length of a tour is the sum of all Euclidean distances on the tour. Let

$\mathrm{TSP}(Q)$ denote the minimum length of a tour through all the points in Q; by convention, $\mathrm{TSP}(\emptyset) = 0$. Assume that the locations of the n points are random variables independently and identically distributed in a compact set $\mathcal{E}$; in [96] it is shown that there exists a constant $\beta_{\mathrm{TSP},d}$ such that

$$\lim_{n\to+\infty} \frac{\mathbb{E}\left[\mathrm{TSP}(Q)\right]}{n^{1-1/d}} = \beta_{\mathrm{TSP},d} \int_{\mathcal{E}} \bar{f}^{1-1/d}(x)\, dx, \tag{2.7}$$

where $\bar{f}$ is the density of the absolutely continuous part of the point distribution. For the case $d = 2$, the constant $\beta_{\mathrm{TSP},2}$ has been estimated numerically as $\beta_{\mathrm{TSP},2} \simeq 0.7120 \pm 0.0002$ [83]. The constant $\beta_{\mathrm{TSP},3}$ has been estimated numerically as $\beta_{\mathrm{TSP},3} \approx 0.6979 \pm 0.0002$, [83]. In this thesis we will refer to $\beta_{\mathrm{TSP},2}$ and $\beta_{\mathrm{TSP},3}$ simply as β_{TSP}: the meaning will be clear from the context. Notice that the limit in equation (2.7) holds for all compact sets: the shape of the set only affects the convergence rate. According to [58], if $\mathcal{E}$ is a "fairly compact and fairly convex" set in the plane, then equation (2.7) provides a "good" estimate of the expected TSP tour length for values of n as low as 15.

Remarkably, the asymptotic average cost of the stochastic TSP for uniform point distributions is an upper bound on the asymptotic average cost for general point distributions: i.e.,

$$\lim_{n\to+\infty} \frac{\mathbb{E}\left[\mathrm{TSP}(Q)\right]}{n^{1-1/d}} \leq \beta_{\mathrm{TSP},d} |\mathcal{E}|^{1/d},$$

where $|\mathcal{E}|$ is the hypervolume of $\mathcal{E}$; this follows directly from an application of Jensen's inequality to the concave function $g(x) = x^{1-\frac{1}{d}}$, $x \in \mathbb{R}_{\geq 0}$ and $d \in \mathbb{N}$, in the right hand side of (2.7)

$$\int_{\mathcal{E}} \bar{f}^{1-1/d}(x)\, dx \leq |\mathcal{E}|^{1/d} \left(\int_{\mathcal{E}} \bar{f}(x)\, dx \right)^{1-\frac{1}{d}} \leq |\mathcal{E}|^{1/d}.$$

Finally, for any bounded environment $\mathcal{E}$, the following (deterministic) bound holds on the length of the TSP tour, uniformly on n [96]:

$$\mathrm{TSP}(Q) \leq \beta_{\mathcal{E},d}\, |\mathcal{E}|^{1/d}\, n^{1-1/d}, \tag{2.8}$$

where $|\mathcal{E}|$ is the hypervolume of $\mathcal{E}$ and $\beta_{\mathcal{E},d}$ is a constant generally larger than $\beta_{\mathrm{TSP},d}$. We will refer to $\beta_{\mathcal{E},d}$ as the characteristic constant of $\mathcal{E}$.

2.4.2 The bipartite matching problem

Let Q be a set of points $X_1, \ldots, X_n$, $Y_1, \ldots, Y_n$ that are i.i.d. in a compact set $\mathcal{E} \subset \mathbb{R}^d$, $d \geq 3$, and distributed according to a density f. Let $\mathrm{BM}(Q) = \min_\sigma \sum_{i=1}^n \|X_i - Y_{\sigma(i)}\|$ denote the optimal bipartite matching of the X and Y points, where σ ranges over all permutations of the integers $1, 2, \ldots, n$. In [38] it is shown that there exists a constant $\beta_{\mathrm{M},d}$ such that

$$\lim_{n\to+\infty} \frac{\mathbb{E}\left[\mathrm{BM}(Q)\right]}{n^{1-1/d}} = \beta_{\mathrm{M},d} \int_{\mathcal{E}} \bar{f}^{1-1/d}(x)\, dx. \tag{2.9}$$

The constant $\beta_{\mathrm{M},3}$ has been estimated numerically as $\beta_{\mathrm{M},3} \approx 0.7080 \pm 0.0002$, [47]. In this thesis we will refer to $\beta_{\mathrm{M},3}$ simply as β_{M}.

2.4.3 Tools for solving TSPs

The decision version of the TSP belongs to the class of NP-complete problems, which suggests that there is no general algorithm capable of finding the optimal tour in an amount of time polynomial in the size of the input. Even though the exact optimal solution of a large TSP can be very hard to compute, several exact and heuristic algorithms and software tools are available for the numerical solution of TSPs.

The most advanced TSP solver to date is arguably `concorde` [3]. Polynomial-time algorithms are available for constant-factor approximations of TSP solutions, among which we mention Christofides' [29]. On a more theoretical side, Arora [4] proved the existence of polynomial-time approximation schemes for the TSP, providing a $(1+\varepsilon)$ constant-factor approximation for any $\varepsilon > 0$.

A modified version of the Lin-Kernighan heuristic [61] is implemented in `linkern`; this powerful solver yields approximations in the order of 5% of the optimal tour cost very quickly for many instances. For example, in our numerical experiments on a machine with a 2GHz Intel Core Duo processor, approximations of random TSPs with 10,000 points typically required about twenty seconds of CPU time.[1]

In this thesis, we will present algorithms that require on-line solutions of possibly large TSPs. Practical implementations of the algorithms will rely on heuristics, such as Lin-Kernighan's or Christofides'. If a constant-factor approximation algorithm is used, the effect on the asymptotic performance guarantees of our algorithms can be simply modeled as a scaling of the constant $\beta_{\mathrm{TSP},d}$.

2.5 Algorithmic Queueing Theory for DVR

Algorithmic queueing theory embeds DVR problems within the framework of queueing theory and consists of three main steps: (i) queueing model of the DVR problem, (ii) establishment of fundamental limitations on performance; and (iii) design of algorithms that enjoy performance guarantees. In this section we review the pioneering work of Bertsimas and Van Ryzin [14, 15, 16], who introduced queueing methods to solve the simplest DVR problem: a vehicle moves along straight lines and visits demands whose time of arrival, location and on-site service are stochastic; information about demand location is communicated to the vehicle upon demand arrival. This problem is called the m-vehicle Dynamic Traveling Repairman Problem, and its analysis paved the way for the development of algorithmic queueing theory.

2.5.1 A basic queueing model for DVR

Consider m vehicles free to move, traveling at a constant velocity v, within a compact, convex environment $\mathcal{E} \subset \mathbb{R}^2$ (in chapter 5 we will also consider the case $\mathcal{E} \subset \mathbb{R}^3$). The vehicles are identical, and have unlimited range and demand servicing capacity. Demands are generated according to a homogeneous (i.e., time-invariant) spatio-temporal Poisson process[2], with time intensity $\lambda \in \mathbb{R}_{>0}$, and spatial density $f : \mathcal{E} \to \mathbb{R}_{>0}$. In other words,

[1] Both `concorde` and `linkern` are freely available for academic research use at `www.tsp.gatech.edu/concorde/index.html`.

[2] There are three main reasons behind the choice of modeling the demand arrival process with a Poisson process: First, the system we deal with is a spatial queueing system, and the arrival process to a queueing system is often well approximated by a Poisson process (see, e.g., [58]). Second, the Poisson process leads

demands arrive to $\mathcal{E}$ according to a Poisson process with intensity λ, and their locations $\{X_j; j \in \mathbb{N}_0\}$ are i.i.d. (i.e., independent and identically distributed) and distributed according to a density f whose support is $\mathcal{E}$. A demand's location becomes known (is realized) at its arrival epoch; thus, at time t we know with *certainty* the locations of demands that arrived prior to time t, but future demand locations form an i.i.d. sequence. The density f satisfies:

$$\mathbb{P}[X_j \in \mathcal{S}] = \int_{\mathcal{S}} f(x)\,dx \quad \forall \mathcal{S} \subseteq \mathcal{E}, \quad \text{and} \quad \int_{\mathcal{E}} f(x)\,dx = 1.$$

We assume that the number of initial outstanding demands is a random variable with finite first and second moments.

Each demand j (demands are labeled in an increasing order with respect to time of arrival) requires an on-site service time $s_j \geq 0$ that is i.i.d. and generally distributed with finite first and second moments denoted by $\bar{s}$ and $\overline{s^2}$ (we also assume $\bar{s} > 0$). A realized demand is removed from the system after one of the vehicles has completed its on-site service. We define the *load factor* $\varrho \doteq \lambda \bar{s}/m$.

The system time of demand j, denoted by T_j, is defined as the elapsed time between the arrival of demand j and the time one of the vehicles completes its service. The waiting time of demand j, W_j, is defined by $W_j = T_j - s_j$. The steady-state system time is defined by $\overline{T} \doteq \limsup_{j\to\infty} \mathbb{E}[T_j]$. A policy for routing the vehicles is said to be *stable* if the expected number of demands in the system is uniformly bounded at all times. A necessary condition for the existence of a stable policy is that $\varrho < 1$. When we refer to *light-load* conditions, we consider the case $\varrho \to 0^+$, in the sense that $\lambda \to 0^+$; when we refer to *heavy-load* conditions, we consider the case $\varrho \to 1^-$, in the sense that $\lambda \to (m/\bar{s})^-$.

Let $\mathcal{P}$ be the set of all causal, stable, and stationary routing policies and $\overline{T}_\pi$ be the system time of a particular policy $\pi \in \mathcal{P}$. The problem is to find a policy $\pi^* \in \mathcal{P}$ (if one exists) such that

$$\overline{T}_{\pi^*} = \inf_{\pi\in\mathcal{P}} \overline{T}_\pi \doteq \overline{T}^*.$$

This problem is known as the m-vehicle Dynamic Traveling Repairman Problem (m-DTRP), and has been introduced and studied by Bertsimas and Van Ryzin in [14, 15, 16].

In general, it is difficult to characterize the optimal achievable performance $\overline{T}^*$ and to compute the optimal policy π^* for arbitrary values of the problem parameters λ, m, etc. It is instead possible and useful to consider particular ranges of parameter values. Specifically, in [14, 15, 16, 108], lower bounds for the optimal steady-state system time are derived for the light-load case (i.e., $\varrho \to 0^+$), and for the heavy-load case (i.e., $\varrho \to 1^-$). Subsequently, policies are designed for these two limiting regimes, and their performance is compared to the lower bounds.

2.5.2 Lower bounds on the optimal system time

For the light-load case, a tight lower bound on the system time is derived in [15]. In the light-load case, the lower bound on the system time is strongly related to the solution of the m-median problem:

$$\overline{T}^* \geq \frac{1}{v} H_m^*(\mathcal{E}) + \bar{s}, \qquad \text{as } \varrho \to 0^+. \tag{2.10}$$

to a model that is analytically tractable. Third, the Poisson process, being "the most random" way to distribute events in time, leads to useful "worst-case" scenario evaluations (worst-case with respect to the possible types of arrival processes).

The bound is tight: there exist policies whose system times, in the limit $\varrho \to 0^+$, attain this bound; we present such asymptotically optimal policies for the light-load case in section 2.5.3.

Two lower bounds exist for the heavy-load case [16, 108]. To present them, we need two definitions.

Definition 2.5.0.1. *Let X be the location of a randomly chosen demand and W be its waiting time. A policy π is said to be*

1. spatially unbiased *if for every pair of sets $\mathcal{S}_1$, $\mathcal{S}_2 \subseteq \mathcal{E}$*

$$\mathbb{E}\left[W \,|X \in \mathcal{S}_1\right] = \mathbb{E}\left[W \,|X \in \mathcal{S}_2\right]; \quad \textit{and}$$

2. spatially biased *if there exists sets $\mathcal{S}_1$, $\mathcal{S}_2 \subseteq \mathcal{E}$ such that*

$$\mathbb{E}\left[W \,|X \in \mathcal{S}_1\right] > \mathbb{E}\left[W \,|X \in \mathcal{S}_2\right].$$

Within the class in $\mathcal{P}$ of *spatially-unbiased* policies the optimal system time $\overline{T}^*_{\mathrm{U}}$ is lower bounded by

$$\overline{T}^*_{\mathrm{U}} \geq \frac{\beta^2_{\mathrm{TSP}}}{2} \frac{\lambda \left[\int_{\mathcal{E}} f^{1/2}(x)dx\right]^2}{m^2\, v^2\, (1-\varrho)^2} \qquad \text{as } \varrho \to 1^-. \tag{2.11}$$

Within the class in $\mathcal{P}$ of *spatially-biased* policies the optimal system time $\overline{T}^*_{\mathrm{B}}$ is lower bounded by

$$\overline{T}^*_{\mathrm{B}} \geq \frac{\beta^2_{\mathrm{TSP}}}{2} \frac{\lambda \left[\int_{\mathcal{E}} f^{2/3}(x)dx\right]^3}{m^2\, v^2\, (1-\varrho)^2} \qquad \text{as } \varrho \to 1^-. \tag{2.12}$$

Both bounds (2.11) and (2.12) are tight: there exist policies whose system times, in the limit $\varrho \to 1^-$, attain these bounds; therefore the inequalities in (2.11) and (2.12) should indeed be replaced by equalities. We present asymptotically optimal policies for the heavy-load case in section 2.5.3. It is shown in [16] that the lower bound in equation (2.12) is always lower than or equal to the lower bound in equation (2.11) for all densities f.

We conclude with some remarks.

Remark 2.5.1. *In equations* (2.11) *and* (2.12)*, the right-hand side approaches $+\infty$ as $\varrho \to 1^-$. Thus, one should more formally write the inequalities with $\overline{T}^*(1-\varrho)^2$ on the left-hand side, so that the right-hand side is finite. However, this makes the presentation less readable, and thus in this thesis we adhere to the less formal but more transparent style of equations* (2.11) *and* (2.12)*. (This is the style generally used in the literature on the DTRP, see, e.g., [16] Section 4.)*

Remark 2.5.2. *It is possible to show (see [16], Proposition 1) that a* uniform *spatial density function leads to the* worst possible performance *and that any deviation from uniformity in the demand distribution will strictly lower the optimal mean system time in both the unbiased and biased case. Additionally, allowing biased service results in a strict reduction of the optimal expected system time for any non-uniform density f. Finally, when the density is uniform there is nothing to be gained by providing biased service.*

Remark 2.5.3. *In this thesis we assume that the vehicles have full knowledge about the spatial demand distribution f. In practice, the vehicles might use an empirical distribution*

function for the demand locations (which might be computed either through a centralized algorithm or through a distributed algorithm). Since the demand locations are i.i.d., the empirical distribution will converge to the true distribution, and therefore after a transient (whose length depends on the arrival rate λ) the vehicles will provide a system time that agrees with the bounds derived under the assumption that f is known by the vehicles.

2.5.3 Centralized and ad hoc policies

In this section, we present centralized, ad hoc policies that are *either* optimal in light load *or* optimal in heavy load. We say that a policy is ad hoc if it performs "well" only for a limited range of values of ϱ. In light load, the SQM policy, described below, provides optimal performance (i.e., $\lim_{\varrho\to 0^+} \overline{T}_{\mathrm{SQM}}/\overline{T}^* = 1$):

> **The m Stochastic Queue Median (SQM) Policy [15]** — Locate one vehicle at each of the m median locations for the environment $\mathcal{E}$. When demands arrive, assign them to the vehicle corresponding to the nearest median location. Have each vehicle service its respective demands in First-Come, First-Served (FCFS) order returning to its median after each service is completed.

This policy, although optimal in light load, has two characteristics that limit its application to robotic networks: First, it quickly becomes unstable as the load increases, i.e., there exists $\varrho_c < 1$ such that for all $\varrho > \varrho_c$ the system time $\overline{T}_{\mathrm{SQM}}$ is infinite (hence, this policy is *ad hoc*). Second, a *central entity* needs to compute the m-median locations and assign them to the vehicles (hence, it is *centralized*).

In heavy load, the UTSP policy, described below, provides optimal unbiased performance (i.e., $\lim_{\varrho\to 1^-} \overline{T}_{\mathrm{UTSP}}/\overline{T}^*_{\mathrm{U}} = 1$):

> **The Unbiased TSP (UTSP) Policy [16]** — Let r be a fixed positive, large integer. From a central point in the interior of $\mathcal{E}$, subdivide the service region into r wedges $\mathcal{E}_1, \ldots, \mathcal{E}_r$ such that $\int_{\mathcal{E}_k} f(x)dx = 1/r$, $k \in \{1, \ldots, r\}$. Within each subregion, form sets of size n/r (n is a design parameter). As sets are formed, deposit them in a queue and service them FCFS with the first available vehicle by forming a TSP on the set and following it in an arbitrary direction. Optimize over n (see [16] for details).

It is possible to show that, as $\varrho \to 1^-$,

$$\overline{T}_{\mathrm{UTSP}} \leq \left(1 + \frac{m}{r}\right) \frac{\beta^2_{\mathrm{TSP}}}{2} \frac{\lambda \left[\int_{\mathcal{E}} f^{1/2}(x)dx\right]^2}{m^2\, v^2\, (1-\varrho)^2}; \tag{2.13}$$

thus, letting $r \to \infty$, the lower bound in equation (2.11) is achieved.

In [16] the authors present an optimal biased policy. This policy, called Biased TSP (BTSP) Policy, relies on an even finer partition of the environment, requires f to be piecewise constant, and also depends on a parameter n.

Although both the UTSP and the BTSP policies are optimal within their respective classes, they have two characteristics that limit their application to robotic networks: First, in the UTSP policy, to ensure stability, n should be chosen so that (see [16], page 961)

$$n > \frac{\lambda^2 \beta^2_{\mathrm{TSP}} \left[\int_{\mathcal{E}} f^{1/2}(x)\, dx\right]^2}{m^2\, v^2\, (1-\varrho)^2};$$

therefore, to ensure stability over a wide range of values of ϱ, the system designer is forced to select a *large* value for n. However, if during the execution of the policy the load factor turns out to be only moderate, demands have to wait for an excessively large set to be formed, and the overall system performance deteriorates significantly. Similar considerations hold for the BTSP policy. Hence, these two policies are *ad hoc*. Second, both policies require a centralized data structure (the demands' queue is shared by the vehicles); hence, both policies are centralized.

Part I
DVR with complex demand models

Chapter 3

DVR with Stochastic Time Constraints

The routing of robotic vehicles is often dynamic *and* time-constrained, in the sense that demands have (possibly stochastic) deadlines on their waiting times. Surveillance missions by teams of unmanned aerial vehicles are a first clear example; in this case, demands are targets whose potential hazard has to be assessed. Should the vehicles take too long to reach the location of the demand, the latter may have escaped and be hard to track. Automatic delivery of time-critical payloads by mobile robotic networks provides a second example. Other examples include transportation-on-demand systems, automated refuse collection trucks, etc.

Routing problems with both a dynamic and a stochastic component have been extensively studied in the last 20 years both in the literature on online algorithms [56, 49] and in the literature on queueing systems (under the name of Dynamic Traveling Repairman Problem, see section 2.5); however, little is known about time-constrained versions of DVR problems, despite their practical relevance. The purpose of this chapter is to fill this gap. Specifically, by using an algorithmic queueing theory approach, we study the following problem, which we call the Dynamic Vehicle Routing Problem with Stochastic Time Constraints (DVRPSTC): m vehicles operating in a bounded environment and traveling with bounded velocity must service demands whose time of arrival, location and on-site service are stochastic; moreover, once a demand arrives, it remains active for a possibly stochastic and demand-dependant amount of time, and then expires. An active demand is successfully serviced when one of the vehicles visits its location before its deadline and provide the required on-site service. The aim is to find the minimum number of vehicles needed to ensure that the long-time fraction of demands that are successfully serviced is larger than a desired value $\phi^{\mathrm{d}} \in (0, 1)$, and to determine the policy the vehicles should execute to ensure that such objective is attained.

Some of the characteristics of the DVRPSTC have been studied in isolation in the literature. When there is *no dynamic* component, and all problem data are known with certainty, the DVRPSTC is closely related to the well-known Vehicle Routing Problem with Time Windows (VPRTW). The VRPTW has been the subject of intensive research efforts for both heuristic and exact optimization approaches (see, for example, [94, 36, 99, 20, 21]). Indeed, the VRPTW is NP-hard; even finding a feasible solution to the VRPTW when the number of vehicles is fixed is itself an NP-complete problem [86]. Chapter 7 in [99] provides a comprehensive survey on exact (exponential-time) solution techniques. Because

of the difficulty of the VRPTW and its wide applicability to real-life situations, many heuristic solution techniques capable of producing high-quality solutions in limited time have been proposed; a recent thorough survey on heuristics for the VRPTW can be found in [20, 21]. The static and deterministic version of the DVRPSTC is also related to the multiple traveling salesman problem with time windows, which deals with finding optimal routes for a fleet of vehicles in order to serve a set of locations; lower and upper bounds for the minimum number of vehicles needed to serve all the locations are found in [68].

When there is *no spatial* component, i.e., all demands arrive at a specific facility, the DVRPSTC becomes a queueing problem with impatient (or reneging) customers. Many papers have studied such queuing problems, especially for their connection with the multi-million call center industry. Pioneering works include [54] and [34], while more recent analysis can be found in [95, 9, 19, 60]. In the last few years, there has been a growing interest in asymptotic analysis [102, 109, 103, 104]. Finally, the DVRPSTC is also related to coverage problems of both static [107, 50] and mobile [33, 18] sensor networks.

The contributions of this chapter are the following. First, we carefully formulate the problem by also taking into account the possible types of available information (e.g., the deadlines). In setting up the problem, we prove some ergodicity results that are interesting in their own right. Second, by using a variety of techniques from geometric probability, we establish a lower bound on the optimal number of vehicles for a given level of service quality (i.e., ϕ^d). In deriving the lower bound, we introduce a novel type of facility location problem, which we call the m-Location Problem with Impatient Customers (m-LPIC), and for which we provide some analysis and algorithms. Third, we analyze two service policies: we (i) show that one of the proposed policies is optimal in light load; (ii) derive an analytical upper bound on the number of vehicles needed by one of the two policies to achieve a given service quality; (iii) find that if the on-site service requirement is "negligible", the minimum number of vehicles is $O(\sqrt{\lambda})$, where λ is the arrival rate for the demands; (iv) prove that one of the proposed policies is within a small factor of the optimal when ϕ^d is close to one, the system is in heavy load, and the deadlines are deterministic. The significance of our results stems from two facts: First, the structural insights into the DVRPSTC provide the system designer with essential information to build business and strategic planning models regarding, e.g., fleet sizing and depot locations. Second, our analysis provides directions and guidelines to route the robotic vehicles once the system is deployed.

The work in this chapter is based on the conference paper [78] and the journal article [77].

3.1 Regenerative Processes and Stopping Times

In this section, we describe some known concepts about regenerative processes and stopping times, on which we will rely extensively later in this chapter.

3.1.1 Regenerative processes

Intuitively, a stochastic process $\{X(t);\, t \in \mathbb{T}\}$, where $\mathbb{T} = \mathbb{N}_0$ or $\mathbb{T} = \mathbb{R}_{\geq 0}$, is said to be *regenerative* if it can be split into independent and identically distributed (i.i.d.) *cycles*, or, in other words, if an imbedded renewal process can be found.

More formally, define a renewal sequence to be a sequence $\{Y_j;\, j \in \mathbb{N}\}$ of i.i.d. positive random variables, and let $S_n \doteq Y_1 + \ldots + Y_n$, $n \in \mathbb{N}$, and $S_0 = 0$. A stochastic process

$\{X(t);\, t \in \mathbb{T}\}$ is said to be regenerative (see [7], page 169) if there exists a $\mathbb{T}$-valued renewal sequence $\{Y_j;\, j \in \mathbb{N}\}$ with the following property: for each $n \in \mathbb{N}_0$, the post-S_n process

$$\theta_{S_n} X \doteq (Y_{n+1}, Y_{n+2}, \ldots, \{X(S_n + t);\, t \in \mathbb{T}\})$$

is independent of $Y_1, \ldots, Y_n$ (the cycles) and its distribution does not depend upon n. We call $\{S_n;\, n \in \mathbb{N}_0\}$ the imbedded renewal process and refer to the S_n's as the regeneration points.

The power of the concept of regenerative processes lies in the existence of a limiting distribution under conditions that are very mild and easy to verify. We next provide some results about discrete-time (i.e. $\mathbb{T} = \mathbb{N}_0$) regenerative processes, and about a particular class of continuous-time (i.e. $\mathbb{T} = \mathbb{R}_{\geq 0}$) regenerative processes, namely the class of *cumulative* processes.

Limit theorems for discrete-time regenerative processes

In this case we consider $\mathbb{T} = \mathbb{N}_0$; in particular, the cycles are integer-valued. Before proceeding, we need the following definition. A discrete probability distribution f_k, $k \in \mathbb{N}_0$, is said to be periodic (see [97]) if there exists an integer $p > 1$ such that all f_k's, except, perhaps, $f_p, f_{2p}, f_{3p}, \ldots$, vanish. Often, it is easy to check if a distribution is non-periodic:

Lemma 3.1.1 (Lemma 2 in [97])**.** *A discrete probability distribution f_k, $k \in \mathbb{N}_0$, is non-periodic if $f_1 > 0$.*

Consider a regenerative process $\{X_j;\, j \in \mathbb{N}_0\}$, with renewal sequence $\{Y_j; j \in \mathbb{N}\}$, and let $Y = Y_1$ be the first regeneration cycle. The stationary version $\{X_j^*;\, j \in \mathbb{N}_0\}$ of the regenerative process $\{X_j;\, j \in \mathbb{N}_0\}$ is defined by (see [97])

$$\mathbb{P}\left[X_j^* \in A\right] = \frac{1}{\mathbb{E}\,[Y]} \sum_{k=0}^{+\infty} \mathbb{P}\,[X_{j+k} \in A |\, Y > k] \cdot \mathbb{P}\,[Y > k], \tag{3.1}$$

for every $j \in \mathbb{N}_0$ and every Borel set A.

Theorem 3.1.2 (Adapted from theorem 2 in [97])**.** *Let $\{X_j;\, j \in \mathbb{N}_0\}$ be a nonnegative regenerative process with first regeneration cycle Y, with $\mathbb{E}\,[Y] < +\infty$. Then, the stationary process given by* (3.1) *is well defined and has a proper distribution function, which is independent of n. Moreover,*

(i) $\mathbb{E}\,[X_0^*] = \mathbb{E}\left[\sum_{j=0}^{Y-1} X_j\right] / \mathbb{E}\,[Y]$*;*

(ii) if $f_k = \mathbb{P}\,[Y = k]$, $k \in \mathbb{N}_0$, is non-periodic, then for every Borel set A,

$$\lim_{j \to +\infty} \mathbb{P}\,[X_j \in A] = \mathbb{P}\,[X_0^* \in A].$$

Limit theorems for cumulative processes

Let $h : [a, b] \to \mathbb{R}$ be a function and Ξ be the set of partitions ξ: $a = x_0 < x_1 < \ldots < x_n = b$ of the interval $[a, b]$. Letting

$$V_h(\xi) = \sum_{k=1}^{n} |h(x_i) - h(x_{i-1})|, \quad \xi \in \Xi,$$

then h is of *bounded variation* if

$$V_h(a,b) \doteq \sup_{\xi \in \Xi} V_h(\xi) < +\infty.$$

We refer to $V_h(a,b)$ as the total variation of h in the interval $[a,b]$. It is immediate to prove the following.

Lemma 3.1.3. *If $h : [a, b] \to \mathbb{R}$ is monotonic, then h is of bounded variation and*

$$V_h(a,b) = |h(b) - h(a)|.$$

A continuous-time stochastic process $\{X(t);\, t \in \mathbb{R}_{\geq 0}\}$ is of bounded variation on $[0,t]$, $t \in \mathbb{R}_{>0}$, if almost all (i.e., except for a set of measure zero) paths are of bounded variation on $[0,t]$. An immediate consequence of lemma 3.1.3 is that every process with increasing paths is of bounded variation on every interval $[0,\, t]$. The total variation process $\{\tilde{X}(t);\, t \in \mathbb{R}_{\geq 0}\}$ associated to the process $\{X(t);\, t \in \mathbb{R}_{\geq 0}\}$ is defined as follows (see [93, page 262]): $\tilde{X}(t)$ maps every path of process $\{X(t);\, t \in \mathbb{R}_{\geq 0}\}$ to its total variation in $[0,t]$ ($\tilde{X}(t)$ is defined as identically zero when a path in $[0,t]$ is of unbounded variation).

A particular class of continuous-time regenerative processes is represented by the class of cumulative processes. A regenerative process $\{X(t);\, t \in \mathbb{R}_{\geq 0}\}$ is said to be cumulative relative to the imbedded renewal process $\{S_n;\, n \in \mathbb{N}_0\}$ if (see [93, page 262])

(C1) $\{X(S_{n+1}) - X(S_n);\, n \in \mathbb{N}_0\}$ is a sequence of i.i.d. random variables,

(C2) $\{X(t);\, t \in \mathbb{R}_{\geq 0}\}$ is of bounded variation on every interval $[0,t]$,

(C3) $\{\tilde{X}(S_{n+1}) - \tilde{X}(S_n);\, n \in \mathbb{N}_0\}$, is a sequence of i.i.d. random variables ($\tilde{X}(S_0)$ is taken equal to zero).

Finally, define

$$\tilde{\kappa}_1 \doteq \mathbb{E}\left[\tilde{X}(S_1)\right]. \tag{3.2}$$

Theorem 3.1.4 (adapted from [93], page 262, and [92], theorem 7)**.** *If $\{X(t);\, t \in \mathbb{R}_{\geq 0}\}$ is a cumulative process relative to the imbedded renewal process $\{S_n;\, n \in \mathbb{N}_0\}$, if $X(0)$ is finite with probability one, if $\mathbb{E}[S_1] < +\infty$, and if $\tilde{\kappa}_1 < +\infty$, then*

$$\lim_{t \to +\infty} \frac{X(t)}{t} = \frac{\mathbb{E}[X(S_1) - X(S_0)]}{\mathbb{E}[S_1]},$$

almost surely.

3.1.2 Stopping times and Wald's lemma

Let $\{\mathcal{F}_n;\, n \in \mathbb{N}_0\}$ be a filtration, i.e., a nondecreasing family of σ-fields. A random variable X is a stopping time with respect to a filtration $\{\mathcal{F}_n;\, n \in \mathbb{N}_0\}$ if

$$\{X = n\} \in \mathcal{F}_n, \quad \text{for all } n \in \mathbb{N}_0.$$

In other words, for X to be a stopping time, it should be possible to decide whether or not $\{X = n\}$ has occurred on the basis of the knowledge of $\mathcal{F}_n$. An important result about stopping times is represented by the following lemma.

Lemma 3.1.5 (Wald's lemma (adapted from proposition A10.2 in [7])). *Let X be an almost surely finite stopping time with respect to $\{\mathcal{F}_n;\, n \in \mathbb{N}_0\}$. Further, let A_1, A_2, ... be i.i.d. random variables such that, for any n, A_n is $\mathcal{F}_n$-measurable and A_{n+1}, A_{n+2}, ... are independent of $\mathcal{F}_n$, and let $\mu = \mathbb{E}[A_1]$. Then, if $A_1 \geq 0$, it holds:*

$$\mathbb{E}\left[\sum_{k=1}^{X} A_k\right] = \mu \mathbb{E}[X].$$

3.2 Problem Setup

In this section, we set up the problem we study in this chapter.

3.2.1 The model

Let the workspace $\mathcal{E} \subset \mathbb{R}^2$ be a compact, convex set. A total of m holonomic vehicles operate in $\mathcal{E}$; the vehicles are free to move, traveling at a maximum velocity v, within $\mathcal{E}$. The vehicles are identical, have unlimited range and demand servicing capacity; moreover, each vehicle is associated with a depot whose location is at $g_k \in \mathcal{E}$, $k \in \{1, \ldots, m\}$. Demands are generated according to a homogeneous (i.e., time-invariant) spatio-temporal Poisson process, with time intensity $\lambda \in \mathbb{R}_{>0}$, and uniform spatial density over $\mathcal{E}$. In other words, demands arrive to $\mathcal{E}$ according to a Poisson process with intensity λ, and their locations $\{X_j;\, j \in \mathbb{N}_0\}$ are i.i.d. and distributed according to a uniform density whose support is $\mathcal{E}$; moreover, the locations are independent of demands' arrival times and of vehicles' positions. Let $\{t_j;\, j \in \mathbb{N}_0\}$ denote the sequence of arrival times of demands; we assume that $t_0 = 0$, and that the first arrival finds the system empty. Let $N(t)$, $t \in \mathbb{R}_{\geq 0}$, denote the number of arrivals in $[0,\, t]$, i.e., $N(t) = \max\{j \in \mathbb{N}_0 | t_j \leq t\}$. Each demand j requires a stochastic amount of on-site service time s_j. A vehicle provides on-site service by staying at the demand's location for the entire on-site service time. On-site service is not-interruptible: once a vehicle starts the service, neither the vehicle can interrupt the service nor the demand can leave the system before service completion. We assume that the nonnegative on-site service times $\{s_j;\, j \in \mathbb{N}_0\}$ are i.i.d. and generally distributed according to a distribution function $F_S(s)$ with first moment $\bar{s}$ and maximum value $s_{\max} \geq 0$.

Each demand j waits for the beginning of its service no longer than a stochastic *patience time* P_j. We assume that the nonnegative patience times $\{P_j; j \in \mathbb{N}_0\}$ are i.i.d and generally distributed according to a distribution function $F_P(p)$ with first moment $\bar{P}$ and maximum value $p_{\max} > 0$; moreover, we assume that $\mathbb{P}[P_j = 0] = 0$, and that the patience times are independent of demands' arrival times, demands' locations, and vehicles' positions. A vehicle can start the on-site service for the jth demand only within the stochastic time window $[t_j,\, t_j + P_j)$. If the on-site service for the jth demand is not started before the time instant $t_j + P_j$, then the jth demand is considered lost; in other words, such demand leaves the system and never returns. If, instead, the on-site service for the jth demand is started before the time instant $t_j + P_j$, then the demand is considered successfully serviced (recall our assumption that on-site service is not interruptible); we call such demand a *successful* demand. The waiting time of demand j, denoted by W_j, is the elapsed time between the arrival of demand j and the time either one of the vehicles starts its service or such demand departs from the system due to impatience, whichever happens first. Hence, the jth demand is considered serviced if and only if $W_j < P_j$. Finally, let $\{t_j^s;\, j \in \mathbb{N}_0\}$ denote the sequence of arrival times of successful demands; note that the sequence $\{t_j^s;\, j \in \mathbb{N}_0\}$ is a thinning of

$\{t_j;\, j \in \mathbb{N}_0\}$. Let $N^s(t)$, $t \in \mathbb{R}_{\geq 0}$, denote the number of arrivals in $[0,\, t]$ that will eventually be serviced, i.e., $N^s(t) = \max\{j \in \mathbb{N}_0 | t_j^s \leq t\}$.

An instance of the problem is represented by the data vector $\mathcal{I} = \{\mathcal{E}, v, \lambda, F_S(s), F_P(p)\}$; the number m of vehicles and their routing policy are, instead, decision variables.

3.2.2 Information structure and control policies

We first describe the information on which a control policy can rely. We identify four types of information.

Arrival time and Location : we assume that the information on arrivals and locations of demands is immediately available to control policies.

On-site service : the on-site service requirement of demands may either (i) be available, (ii) or may be available only through prior statistics, or (iii) may not be available to control policies.

Patience time : the patience time of demands may either (i) be available, (ii) or may be available only through prior statistics, or (iii) may not be available to control policies; however, we assume that $p_{\max}$ (or at least an upper bound on it) is always known.

Departure notification : the information that a demand leaves the system due to impatience may or may not be available to control policies (if the patience time is available, such information is clearly available).

Hence, several information structures are relevant. The *least informative* case is when on-site service requirements, patience times and departure notifications are not available; the *most informative* case is when on-site service requirements and patience times are available.

We next define the notion of *outstanding demand* for different information structures. If departure notifications are available, a demand is considered outstanding if (i) no vehicle has yet reached its location, (ii) the demand is still in the system. When departure notifications are *not* available, a demand is considered outstanding if (i) the elapsed time from its arrival is less than $p_{\max}$ (note that $p_{\max}$ is always known by the vehicles), and (ii) no vehicle has yet reached its location. Note that, in absence of departure notifications, a vehicle will sometimes reach locations of demands that are no longer in the system.

Given an instance $\mathcal{I}$ and a particular information structure, let $\mathcal{P}$ be the set of all causal, stationary, and *work-conserving* policies. A policy is said to be work-conserving if (i) when a vehicle has no outstanding (in the above sense) demands to service, it moves rectilinearly to (or remains at) its depot location, (ii) when there are outstanding demands, there is at least one vehicle providing service to at least one of them (either on-site or by traveling). Property (i) is a technical condition needed to ensure that the underlying stochastic processes are regenerative, while property (ii) is a standard assumption needed to avoid pathological situations. The system is said to be *idling* if all vehicles are at their depot locations and there are no outstanding demands. We assume that initially all vehicles are at their depots.

We will show in section 3.3 that under any policy belonging to $\mathcal{P}$ the stochastic process $\{N^s(t)/t;\, t \in \mathbb{R}_{\geq 0}\}$ converges almost surely to a *constant*; hence we define the arrival rate of successful demands under a particular policy $\pi \in \mathcal{P}$ as

$$\lambda_\pi^s \doteq \lim_{t \to +\infty} \frac{N^s(t)}{t},$$

where the limit is an almost-sure limit. We, then, define the *success factor* of a policy $\pi \in \mathcal{P}$ as

$$\phi_\pi \doteq \frac{\lambda_\pi^s}{\lambda} = \lim_{t \to +\infty} \frac{N^s(t)}{N(t)}. \tag{3.3}$$

From its definition, ϕ_π represents the fraction of demands that are successfully serviced; in particular, $\phi_\pi \in [0, 1]$, and the larger ϕ_π, the larger the fraction of successfully serviced demands.

3.2.3 Problem definition

Given an instance $\mathcal{I}$, a particular information structure, and a desired success factor $\phi^d \in (0, 1)$, the problem is to determine a vehicle routing policy π^* that guarantees a success factor at least as large as ϕ^d with the minimum possible number of vehicles. Formally, we study the following optimization problem $\mathcal{OPT}$:

$$\begin{aligned} \mathcal{OPT} : \min_{\pi \in \mathcal{P}} \quad & |\pi| \\ \text{subject to} \quad & \phi_\pi \geq \phi^{\mathrm{d}}, \end{aligned} \tag{3.4}$$

where $|\pi|$ is the number of vehicles used by π. We let m^* denote the solution to the optimization problem $\mathcal{OPT}$.

A particular issue that we will address in this chapter is how the solution to the optimization problem $\mathcal{OPT}$ scales with the arrival rate λ. Formally, consider an information structure and a desired success factor $\phi^d \in (0, 1)$; moreover, let $\mathcal{I}(\lambda)$ be a problem instance where the arrival rate λ is a variable parameter, and let $\mathcal{P}(\lambda)$ be the corresponding set of admissible policies, parameterized by λ. The solution to the optimization problem $\mathcal{OPT}$ is said to be $\Omega(g(\lambda))$, where $g(\lambda) : \mathbb{R}_{>0} \to \mathbb{R}_{>0}$, if

$$\liminf_{\lambda \to +\infty} \frac{m^*(\lambda)}{g(\lambda)} \geq c, \qquad c \in \mathbb{R}_{>0}.$$

Similarly, the solution to the optimization problem $\mathcal{OPT}$ is said to be $O(g(\lambda))$, where $g(\lambda) : \mathbb{R}_{>0} \to \mathbb{R}_{>0}$, if

$$\limsup_{\lambda \to +\infty} \frac{m^*(\lambda)}{g(\lambda)} \leq c, \qquad c \in \mathbb{R}_{\geq 0}.$$

The solution to the optimization problem $\mathcal{OPT}$ is said to be $\Theta(g(\lambda))$, where $g(\lambda) : \mathbb{R}_{>0} \to \mathbb{R}_{>0}$, if it is both $\Omega(g(\lambda))$ and $O(g(\lambda))$.

3.3 Ergodicity, Acceptance Probabilities, and Limit Theorems

A demand that finds the system idling faces a situation *probabilistically identical* to that of the first demand. Hence, in our model, all of the relevant stochastic processes (e.g., $N^s(t)/t$) are *regenerative*, and the regeneration points are the time instants in which an arrival finds the system idling.

With the above discussion in mind, consider the following quantities.

- Let $\{C_i;\, i \in \mathbb{N}\}$ be the sequence of successive busy cycles: a busy cycle is the duration between two successive arrival epochs of demands finding the system idling. The C_i's

are i.i.d. random variables on $\mathbb{R}_{\geq 0}$.

- Let $\{B_i; i \in \mathbb{N}\}$ be the sequence of successive busy periods: the busy period is the part of the busy cycle during which at least one vehicle is providing service (either by traveling or on-site) to a demand, or it is moving to its depot. The B_i's are i.i.d. random variables on $\mathbb{R}_{\geq 0}$.

- Let $\{L_i; i \in \mathbb{N}\}$ (or $\{L_i^s; i \in \mathbb{N}\}$) be the number of demands arrived in the system (or successfully serviced) during the ith busy period, including the one initializing it. The L_i's (or L_i^s's) are i.i.d. random variables on $\mathbb{N}$.

In what follows, we use the imbedded renewal processes:

$$\begin{cases} \Gamma_0 &= 0, \\ \Gamma_i &= \Gamma_{i-1} + C_i, \qquad i \geq 1. \end{cases} \qquad (\text{on } \mathbb{R}_{\geq 0})$$

$$\begin{cases} \Lambda_0 &= 0, \\ \Lambda_i &= \Lambda_{i-1} + L_i, \qquad i \geq 1. \end{cases} \qquad (\text{on } \mathbb{N}_0)$$

In order to apply the results on regenerative processes, we first have to prove the finiteness of busy cycles.

Lemma 3.3.1 (Finiteness of busy cycles)**.** *Given an instance $\mathcal{I}$, an information structure, and a policy belonging to $\mathcal{P}$, we have*

$$\mathbb{E}[C_1] < +\infty.$$

Proof. From the definitions we have $\mathbb{E}[C_1] = \mathbb{E}[B_1] + (1/\lambda)$, where B_1 is the part of the first regeneration cycle during which the vehicles continuously work, i.e., there is at least one outstanding demand, or there is at least one vehicle moving to its depot. Note that, if there are no arrivals during a time interval of length $p_{\max} + s_{\max} + \operatorname{diam}(\mathcal{E})/v$, all vehicles will surely be at their depots after that time interval; in fact, in such a case, by our definition of work-conserving policy, the last demand is visited no later than $p_{\max}$, and therefore the last demand is serviced no later than $p_{\max} + s_{\max}$ and all vehicles will be at their depots no later than $p_{\max} + s_{\max} + \operatorname{diam}(\mathcal{E})/v$. Hence, we obtain the inequality, for $b \in \mathbb{R}_{\geq 0}$,

$$\begin{aligned} \mathbb{P}\left[B_1 \geq b + p_{\max} + s_{\max} + \tfrac{\operatorname{diam}(\mathcal{E})}{v}\right] &= \mathbb{P}\left[B_1 \geq b + p_{\max} + s_{\max} + \tfrac{\operatorname{diam}(\mathcal{E})}{v} \middle|\, B_1 \geq b\right] \mathbb{P}[B_1 \geq b] \\ &\quad + \underbrace{\mathbb{P}\left[B_1 \geq b + p_{\max} + s_{\max} + \tfrac{\operatorname{diam}(\mathcal{E})}{v} \middle|\, B_1 < b\right]}_{=0} \mathbb{P}[B_1 < b] \\ &\leq \mathbb{P}\left[\text{at least one arrival in } \left[b,\, b + p_{\max} + s_{\max} + \tfrac{\operatorname{diam}(\mathcal{E})}{v}\right)\right] \mathbb{P}[B_1 \geq b] \\ &= \left(1 - \exp\left(-\lambda\left(p_{\max} + s_{\max} + \frac{\operatorname{diam}(\mathcal{E})}{v}\right)\right)\right) \mathbb{P}[B_1 \geq b]. \end{aligned} \tag{3.5}$$

Then, we can bound $\mathbb{E}[B_1]$ according to

$$\begin{aligned}
\mathbb{E}[B_1] &= \int_0^{+\infty} \mathbb{P}[B_1 \geq b]\, db = \int_0^{p_{\max}+s_{\max}+\operatorname{diam}(\mathcal{E})/v} \mathbb{P}[B_1 \geq b]\, db \\
&\quad + \sum_{k=1}^{\infty} \int_{k(p_{\max}+s_{\max}+\operatorname{diam}(\mathcal{E})/v)}^{(k+1)(p_{\max}+s_{\max}+\operatorname{diam}(\mathcal{E})/v)} \mathbb{P}[B_1 \geq b]\, db \\
&= \int_0^{p_{\max}+s_{\max}+\operatorname{diam}(\mathcal{E})/v} \mathbb{P}[B_1 \geq b]\, db \\
&\quad + \sum_{k=1}^{\infty} \int_0^{p_{\max}+s_{\max}+\operatorname{diam}(\mathcal{E})/v} \mathbb{P}[B_1 \geq b + k\,(p_{\max}+s_{\max}+\operatorname{diam}(\mathcal{E})/v)]\, db \\
&\leq \int_0^{p_{\max}+s_{\max}+\operatorname{diam}(\mathcal{E})/v} \mathbb{P}[B_1 \geq b]\, db \\
&\quad + \sum_{k=1}^{\infty} \int_0^{p_{\max}+s_{\max}+\operatorname{diam}(\mathcal{E})/v} (1-\exp(-\lambda\,(p_{\max}+s_{\max}+\operatorname{diam}(\mathcal{E})/v)))^k\, \mathbb{P}[B_1 \geq b]\, db \\
&< +\infty.
\end{aligned}$$

Thus, we have $\mathbb{E}[C_1] < +\infty$. □

A simple relation between C_1 and L_1 is provided by the following lemma.

Lemma 3.3.2. *Given an instance $\mathcal{I}$, a particular information structure, and a policy belonging to $\mathcal{P}$, we have*

$$\mathbb{E}[L_1] = \lambda\, \mathbb{E}[C_1].$$

Proof. Let A_j, $j \in \mathbb{N}$, be the jth interarrival time (the sequence $\{A_j;\, j \in \mathbb{N}\}$ is clearly i.i.d.), and note that (recall that the demand initializing C_1 is included in L_1, see Figure 3-1)

$$C_1 = A_1 + \ldots + A_{L_1}.$$

The idea is to use Wald's lemma, with L_1 acting as a stopping time for the sequence $\{A_j;\, j \in \mathbb{N}\}$ (see Figure 3-1). In the remainder of the proof we show that the assumptions of Wald's lemma are satisfied.

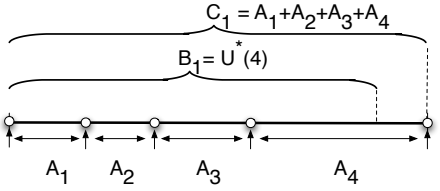


Figure 3-1: A cycle with $L_1 = 4$.

Define $\mathcal{F}_n$, $n \in \mathbb{N}$, as the filtration generated by the random variables $\{A_j\}_{j=1}^{n}$, $\{s_j\}_{j=0}^{n-1}$, $\{P_j\}_{j=0}^{n-1}$, and $\{X_j\}_{j=0}^{n-1}$. Note that A_{n+1}, A_{n+2}, ... are independent of $\mathcal{F}_n$.

Consider, n, $n \in \mathbb{N}$, arrivals to the system and let $U^*(n)$ be the time instant at which the last vehicle reaches its depot, provided that no new demand arrives during the service

of such n demands. Note that we can write (see again Figure 3-1)

$$L_1 = \inf\{j \in \mathbb{N} |\, A_1 + \ldots + A_j > U^*(j)\};$$

since the event $\{L_1 = n\}$ belongs to $\mathcal{F}_n$, L_1 can be viewed as a stopping time with respect to the filtration $\mathcal{F}_n$. L_1 is an almost surely finite stopping time. In fact, by using equation (3.5) we can write

$$\begin{aligned}\lim_{k\to+\infty} \mathbb{P}\left[B_1 \geq k(p_{\max} + s_{\max} + \operatorname{diam}(\mathcal{E})/v)\right] &\leq \lim_{k\to+\infty} (1-\exp(-\lambda(p_{\max} + s_{\max} + \operatorname{diam}(\mathcal{E})/v)))^k \\ &= 0,\end{aligned}$$

i.e. B_1 is almost surely finite; hence L_1 is almost surely finite. Hence the assumptions of Wald's lemma are satisfied and the claim follows. □

Since the busy cycles are finite, we can use the theory of regenerative processes to prove some fundamental limit theorems. We start with the following theorem about the almost sure convergence to a constant of the stochastic process $\{N^s(t)/t;\, t \in \mathbb{R}_{\geq 0}\}$.

Theorem 3.3.3 (Ergodic theorem)**.** *Given an instance $\mathcal{I}$, a particular information structure, and a policy belonging to $\mathcal{P}$, the stochastic process $\{N^s(t)/t;\, t \in \mathbb{R}_{\geq 0}\}$ converges almost surely to a* constant*, in particular*

$$\lim_{t\to+\infty} \frac{N^s(t)}{t} = \frac{\mathbb{E}\left[L_1^s\right]}{\mathbb{E}\left[C_1\right]}, \qquad \textit{almost surely.}$$

Proof. From the previous discussion, the stochastic process $\{N^s(t)/t;\, t \in \mathbb{R}_{\geq 0}\}$ is regenerative relative to the continuous-time renewal process $\{\Gamma_i;\, i \in \mathbb{N}_0\}$. To apply theorem 3.1.4, we first have to show that $\{N^s(t);\, t \in \mathbb{R}_{\geq 0}\}$ is a cumulative process with respect to $\{\Gamma_i;\, i \in \mathbb{N}_0\}$. Indeed, from the definitions, the sequence $\{N^s(\Gamma_{i+1}) - N^s(\Gamma_i);\, i \in \mathbb{N}_0\}$ is a sequence of independent, identically distributed random variables; moreover, $\{N^s(t);\, t \in \mathbb{R}_{\geq 0}\}$ is of bounded variation on every interval $[0,\, t]$, since its paths are increasing. Finally, since

$$\tilde{N}^s(\Gamma_{i+1}) - \tilde{N}^s(\Gamma_i) = N^s(\Gamma_{i+1}) - N^s(\Gamma_0) - N^s(\Gamma_i) + N^s(\Gamma_0) = N^s(\Gamma_{i+1}) - N^s(\Gamma_i),$$

for every $i \in \mathbb{N}_0$, we conclude that the sequence $\{\tilde{N}^s(\Gamma_{i+1}) - \tilde{N}^s(\Gamma_i);\, i \in \mathbb{N}\}$ is also a sequence of independent, identically distributed random variables.

Since $\{N^s(t);\, t \in \mathbb{R}_{\geq 0}\}$ has increasing paths and $\{N^s(t);\, t \in \mathbb{R}_{\geq 0}\}$ is a thinning of $\{N(t);\, t \in \mathbb{R}_{\geq 0}\}$, and by applying lemma 3.3.2, we obtain

$$\begin{aligned}\tilde{\kappa}_1 &= \mathbb{E}\left[\tilde{N}^s(\Gamma_1) - \tilde{N}^s(\Gamma_0)\right] = \mathbb{E}\left[N^s(\Gamma_1) - N^s(\Gamma_0)\right] \\ &\leq \mathbb{E}\left[N^s(\Gamma_1)\right] \leq \mathbb{E}\left[N(\Gamma_1)\right] = \mathbb{E}\left[L_1 + 1\right] = \lambda\,\mathbb{E}\left[C_1\right] + 1 < +\infty.\end{aligned}$$

Since $\{N^s(t);\, t \in \mathbb{R}_{\geq 0}\}$ is a cumulative process with respect to $\{\Gamma_i;\, i \in \mathbb{N}_0\}$, $\mathbb{E}\left[\Gamma_1\right] = \mathbb{E}\left[C_1\right] < +\infty$, and $\tilde{\kappa}_1 < +\infty$, we can apply theorem 3.1.4 and obtain

$$\lim_{t\to+\infty} \frac{N^s(t)}{t} = \frac{\mathbb{E}\left[N^s(\Gamma_1) - N^s(\Gamma_0)\right]}{\mathbb{E}\left[\Gamma_1\right]};$$

by noting that $\mathbb{E}\left[N^s(\Gamma_1) - N^s(\Gamma_0)\right] = \mathbb{E}\left[L_1^s\right]$ we obtain the claim. □

It is natural to wonder if the optimization problem $\mathcal{OPT}$ in equation (3.4) can be restated in terms of acceptance probabilities, in other words, if the equality

$$\phi_\pi \stackrel{?}{=} \lim_{j\to\infty} \mathbb{P}_\pi\left[W_j < P_j\right]$$

holds, where $\mathbb{P}_\pi\left[W_j < P_j\right]$ is the probability, under a policy π, that the jth demand is serviced. The following theorem provides an affirmative answer. The usefulness of such result stems from two facts: (i) on a theoretical level, formulating the problem in terms of time-averages or limiting probabilities is *equivalent*, (ii) on a practical level, in some cases it might be easier to characterize ϕ_π, while in other cases it might be easier to study $\lim_{j\to\infty} \mathbb{P}_\pi\left[W_j < P_j\right]$.

Theorem 3.3.4 (Well-posedness)**.** *Given an instance $\mathcal{I}$, a particular information structure, and a policy π belonging to $\mathcal{P}$, the following equality holds:*

$$\phi_\pi \;=\; \lim_{j\to\infty} \mathbb{P}_\pi\left[W_j < P_j\right], \qquad \textit{almost surely,}$$

where ϕ_π is defined in equation (3.3)*.*

Proof. In this proof, to keep the notation simple, we avoid the usage of the subscript π. Let I_j^s be the indicator random variable

$$I_j^s = \begin{cases} 1 & \text{if } W_j < P_j, \\ 0 & \text{if } W_j = P_j, \end{cases}$$

i.e., I_j^s equals one if the jth demand is successfully serviced. From the previous discussion, the stochastic process $\{I_j^s;\, j \in \mathbb{N}_0\}$ is regenerative relative to the discrete-time renewal process $\{\Lambda_i;\, i \in \mathbb{N}_0\}$, and it is nonnegative. By lemma 3.3.1 and lemma 3.3.2, the expectation of L_1, which is the first regeneration cycle, is finite. Moreover, it clearly holds $\mathbb{P}\left[L_1 = 1\right] > 0$, hence by lemma 3.1.1 the distribution of L_1 is non-periodic. Let $\{I_j^{s,*};\, j \in \mathbb{N}_0\}$ be the stationary version of $\{I_j^s;\, j \in \mathbb{N}_0\}$. Then, by applying theorem 3.1.2 part (ii), by noting that $\tilde{I}_0^{s,*}$ is an indicator random variable and thus $\mathbb{P}\left[I_0^{s,*} = 1\right] = \mathbb{E}\left[I_0^{s,*}\right]$, and by finally applying theorem 3.1.2 part (i), we obtain the series of equalities

$$\lim_{j\to+\infty} \mathbb{P}\left[I_j^s = 1\right] \;=\; \mathbb{P}\left[I_0^{s,*} = 1\right] \;=\; \mathbb{E}\left[I_0^{s,*}\right] = \frac{\mathbb{E}\left[\sum_{j=0}^{L_1-1} I_j^s\right]}{\mathbb{E}\left[L_1\right]};$$

since $\mathbb{E}\left[\sum_{j=0}^{L_1-1} I_j^s\right] = \mathbb{E}\left[L_1^s\right]$, and $\lim_{j\to+\infty} \mathbb{P}\left[I_j^s = 1\right] = \lim_{j\to+\infty} \mathbb{P}\left[W_j < P_j\right]$, we get

$$\lim_{j\to+\infty} \mathbb{P}\left[W_j < P_j\right] \;=\; \frac{\mathbb{E}\left[L_1^s\right]}{\mathbb{E}\left[L_1\right]}.$$

By lemma 3.3.2, $\mathbb{E}\left[L_1\right] = \lambda\mathbb{E}\left[C_1\right]$, hence we obtain, almost surely,

$$\lim_{j\to+\infty} \mathbb{P}\left[W_j < P_j\right] \;=\; \frac{\mathbb{E}\left[L_1^s\right]}{\lambda\mathbb{E}\left[C_1\right]} = \frac{1}{\lambda} \lim_{t\to+\infty} \frac{N^s(t)}{t} = \phi,$$

where the second equality follows from theorem 3.3.3, and the third equality follows from the definition of ϕ in equation (3.3). □

In the next three sections we provide lower bounds for the solution to the optimization problem $\mathcal{OPT}$, we analyze two service policies, and we compare their performance to the lower bounds. Our strategy is to derive lower bounds that are valid under the most informative information structures, and present and analyze policies for the least informative information structure. Such approach will allow to find scaling laws for the solution to the optimization problem $\mathcal{OPT}$, which are valid under each information structure; in particular, and perhaps surprisingly, we will find that, when the on-site service time is negligible, the optimal number of vehicles m^* is $\Theta(\sqrt{\lambda})$ under any information structure and for any problem instance.

3.4 Light Load Lower Bound

In this section, we present a lower bound for the optimization problem $\mathcal{OPT}$ that holds under any information structure. This lower bound is intimately related to a novel type of facility location problem, for which we will provide some analysis and algorithms later in this section.

3.4.1 Lower bound

Let $G = (g_1, \dots, g_m)$ and define

$$R_m(G, \mathcal{E}) \doteq \frac{1}{|\mathcal{E}|} \int_{\mathcal{E}} \left(1 - F_P \left(\min_{k \in \{1,\dots,m\}} \frac{\|x - g_k\|}{v} \right) \right) dx. \tag{3.6}$$

Theorem 3.4.1. *Given an instance $\mathcal{I}$, an information structure, and a desired success factor $\phi^{\mathrm{d}} \in (0, 1)$, the solution to the minimization problem $\mathcal{OPT}$ is lower bounded by the solution to the minimization problem*

$$\begin{aligned} \min_{m \in \mathbb{N}} \quad & m \\ \text{subject to} \quad & \sup_{G \in \mathcal{E}^m} R_m(G, \mathcal{E}) \geq \phi^{\mathrm{d}}. \end{aligned} \tag{3.7}$$

Proof. Consider a policy $\pi \in \mathcal{P}$, and assume that m vehicles execute such policy. In the remainder of the proof, to keep the notation simple, we avoid the usage of the subscript π. Consider the jth demand, and let $\tilde{g}_k$ be the position of the kth vehicle when such demand arrives. Let $\tilde{G} = (\tilde{g}_1, \dots, \tilde{g}_m)$. Obviously, the waiting time of demand j is at least as large as the minimum travel time between its position and the closest vehicle's position, i.e., $W_j \geq \min_{k \in \{1,\dots,m\}} \|X_j - \tilde{g}_k\|/v$. The vehicles are located in the workspace according to some generally unknown cumulative distribution function that depends on the policy; we denote such distribution function as $F : \mathcal{E}^m \to [0, 1]$. Then, the acceptance probability for demand j can be bounded according to (recall that X_j and P_j are, by assumption,

independent of $\tilde{G}$)

$$\begin{aligned}
\mathbb{P}[W_j < P_j] &\leq \mathbb{P}\left[\min_{k\in\{1,\dots,m\}} \tfrac{\|X_j-\tilde{g}_k\|}{v} < P_j\right] \\
&= \int_{\mathcal{E}^m} \mathbb{P}\left[\min_{k\in\{1,\dots,m\}} \tfrac{\|X_j-g_k\|}{v} < P_j \,|\, \tilde{G}=G\right] dF(G) \\
&\leq \int_{\mathcal{E}^m} \sup_{G\in\mathcal{E}^m} \mathbb{P}\left[\min_{k\in\{1,\dots,m\}} \tfrac{\|X_j-g_k\|}{v} < P_j\right] dF(G) \\
&= \sup_{G\in\mathcal{E}^m} \mathbb{P}\left[\min_{k\in\{1,\dots,m\}} \tfrac{\|X_j-g_k\|}{v} < P_j\right] \\
&= \sup_{G\in\mathcal{E}^m} \frac{1}{|\mathcal{E}|}\int_{\mathcal{E}} \mathbb{P}\left[\min_{k\in\{1,\dots,m\}} \tfrac{\|X_j-g_k\|}{v} < P_j \,|\, X_j=x_j\right] dx_j \\
&= \sup_{G\in\mathcal{E}^m} \frac{1}{|\mathcal{E}|}\int_{\mathcal{E}} \left(1 - F_P\left(\min_{k\in\{1,\dots,m\}} \frac{\|x_j-g_k\|}{v}\right)\right) dx_j.
\end{aligned}$$

Hence, we have $\phi = \lim_{j\to+\infty} \mathbb{P}[W_j < P_j] \leq \sup_{G\in\mathcal{E}^m} R_m(G,\mathcal{E})$, and a necessary condition for ϕ to be at least as large as ϕ^{d} is that $\sup_{G\in\mathcal{E}^m} R_m(G,\mathcal{E}) \geq \phi^{\mathrm{d}}$. □

In section 3.5, we will present a policy that only requires the knowledge of F_P and is optimal in light load, i.e., in the limit $\varrho \to 0^+$. As a matter of fact, in equation (3.7) we have implicitly introduced a novel type of facility location problem, which is worthy a definition.

Definition 3.4.1.1 (The m-location problem with impatient customers (m-LPIC))**.** *Given a compact, convex set $\mathcal{E} \subset \mathbb{R}^2$, a cumulative distribution function $F_P : \mathbb{R} \to [0,\ 1]$, a constant $v > 0$, and an integer $m \in \mathbb{N}$, the m-location problem with impatient customers is the optimization problem:*

$$R_m^*(\mathcal{E}) \doteq \sup_{G\in\mathcal{E}^m} R_m(G,\mathcal{E}).$$

Note that in definition 3.4.1.1, slightly generalizing our model, we did not assume $F_P(0) = 0$. Next, we provide some analysis and algorithms for the m-LPIC.

3.4.2 Analysis and algorithms for the m-LPIC

In this section we study in some detail the m-location problem with impatient customers. In particular, we study (i) conditions under which a maximizer exists, (ii) a gradient-ascent law for the optimization of the objective function R_m (defined in equation (3.6)), and (iii) the concavity of the problem for the case $m = 1$. We begin with the following theorem, which shows that a maximizer for the m-LPIC exists in most practical scenarios.

Theorem 3.4.2 (Existence of a maximizer)**.** *Assume that F_P is piecewise differentiable on $\mathbb{R}_{\geq 0}$ with a finite number of (jump) discontinuities; then, $R_m(G,\mathcal{E})$ has a global maximum.*

Proof. The claim is a direct consequence of theorem 2.2 in [32]. By using the monotonicity of F_P, we can write R_m as:

$$\begin{aligned}
R_m(G,\mathcal{E}) &= 1 - \frac{1}{|\mathcal{E}|}\int_{\mathcal{E}^m} F_P\left(\min_{k\in\{1,\dots,m\}} \frac{\|x-g_k\|}{v}\right) dx \\
&= 1 + \underbrace{\frac{1}{|\mathcal{E}|}\int_{\mathcal{E}^m} \max_{k\in\{1,\dots,m\}} -F_P\left(\frac{\|x-g_k\|}{v}\right) dx}_{\doteq L_m(G,\mathcal{E})}.
\end{aligned} \tag{3.8}$$

Since the function $-F_P$ is non-increasing and piecewise differentiable with a finite number of jump discontinuities, part (i) of theorem 2.2 in [32] shows that $L_m(G, \mathcal{E})$ (and hence $R_m(G, \mathcal{E})$) is globally Lipschitz on $\mathcal{E}^m$. Therefore, $R_m(G, \mathcal{E})$ is continuous on a compact set (since $\mathcal{E}$ is compact), and, by the extreme value theorem, it has a global maximum. □

It is also possible to state a differentiability result, which will be the basis for a gradient-ascent algorithm for the m-LPIC.

Theorem 3.4.3 (Differentiability of $R_m(G, \mathcal{E})$)**.** *Assume that F_P is differentiable on $\mathbb{R}_{\geq 0}$ with derivative equal to f_P; then $R_m(G, \mathcal{E})$ (defined in equation (3.6)) is continuously differentiable on $\mathcal{E}^m \setminus \Gamma_{coinc}$, where for each $j \in \{1, \dots, m\}$*

$$\frac{\partial R_m}{\partial g_j}(G, \mathcal{E}) = \frac{1}{v\,|\mathcal{E}|} \int_{V_j(G)} f_P(\|x - g_j\|/v)\, \frac{x - g_j}{\|x - g_j\|}\, dx,$$

where $\mathcal{V}(G) = (V_1(G), \cdots, V_m(G))$ is the Voronoi diagram generated by $G = (g_1, \dots, g_m)$.

Proof. By following the same arguments used in the proof of theorem 3.4.2, one can show that part (ii) of theorem 2.2 in [32] is applicable, and the claim is an immediate consequence. □

Remark 3.4.4. *By using the results in part (ii) of theorem 2.2 in [32] , theorem 3.4.5 can be extended to the case where F_P is piecewise differentiable on $\mathbb{R}_{\geq 0}$ with a finite number of (jump) discontinuities; however, the expression for the gradient is quite cumbersome and is omitted.*

By using theorem 3.4.3 we can readily set up a gradient-ascent law to maximize the locational optimization function R_m. Specifically, assume that F_P is differentiable on $\mathbb{R}_{\geq 0}$; then, consider the following continuous gradient-ascent law defined over the set $\mathcal{E}^m$ (a discrete version can be similarly stated), where $j \in \{1, \dots, m\}$:

$$\dot{g}_j(t) = \begin{cases} \frac{\partial R_m}{\partial g_j}(G(t), \mathcal{E}) & \text{if } G(t) \in \mathcal{E}^m \setminus \Gamma_{\text{coinc}} \text{ and } g_j(t) \in \text{int}(\mathcal{E}), \\ \text{pr}_{\mathcal{E}}\Big(\frac{\partial R_m}{\partial g_j}(G(t), \mathcal{E})\Big) & \text{if } G(t) \in \mathcal{E}^m \setminus \Gamma_{\text{coinc}} \text{ and } g_j(t) \in \partial\mathcal{E}, \\ 0 & \text{otherwise;} \end{cases} \qquad t \in \mathbb{R}_{\geq 0}, \tag{3.9}$$

where the dot represents differentiation with respect to t, $\text{int}(\mathcal{E})$ is the interior of $\mathcal{E}$, $\text{pr}_{\mathcal{E}}(\partial R_m/\partial g_j)$ is the orthogonal projection onto $\mathcal{E}$ of $\partial R_m/\partial g_j$, and we assume that the Voronoi diagram is updated continuously. The vector field is discontinuous, so we understand the solutions in the Krasovskii's sense; see [55], [42]. The convergence properties of the gradient-ascent law in equation (3.9) are summarized by the following theorem.

Theorem 3.4.5 (Convergence of gradient ascent (3.9))**.** *Assume that F_P is differentiable on $\mathbb{R}_{\geq 0}$; then, for each initial condition $G(0) \in \mathcal{E}^m \setminus \Gamma_{\text{coinc}}$, the Krasovskii solution that exactly satisfies (3.9) monotonically optimizes $R_m(G, \mathcal{E})$ and asymptotically converges to the union of Γ_{coinc} and the set of critical points of $R_m(G, \mathcal{E})$.*

Proof. The proof of this theorem is very similar to the proof of theorem 4.1 in [59]. By the definition of (3.9), the domain is strongly invariant for the closed-loop system, i.e., any

trajectory starting in $\mathcal{E}^m$ remains in the domain. Along any Krasovskii solution of the system that exactly satisfies (3.9), we have outside Γ_{coinc}

$$\frac{dR_m}{dt}(G(t), \mathcal{E}) = \sum_{j=1}^{m} \frac{\partial R_m}{\partial g_j} \dot{g}_j \geq 0;$$

in particular, dR_m/dt is zero only if the solution is at a critical point of R_m on the domain $\mathcal{E}^m \setminus \Gamma_{\text{coinc}}$. Therefore, while the solution is outside Γ_{coinc}, the function R_m is optimized. If the solution does not reach Γ_{coinc}, then the application of the LaSalle Invariance principle with the function $-R_m$ guarantees that it will reach the set of critical points of R_m. Otherwise, the solution reaches Γ_{coinc} and stays in it. This concludes the proof. □

Hence, under the assumption that F_P is differentiable on $\mathbb{R}_{\geq 0}$, theorems 3.4.2 and 3.4.5 imply that $R_m(G, \mathcal{E})$ has a global maximum, which can be computed, at least locally, with the gradient-ascent (3.9).

The gradient ascent law (3.9) is *not* guaranteed to find a global maximum. Indeed, the m-LPIC is related to the m-median problem (see section 2.3.4), where the objective is to minimize $\mathbb{E}\left[\min_{k\in\{1,\dots,m\}} \|x - g_k\|\right]$, and whose discrete version is NP-hard [1]. Since the m-LPIC can coincide with the m-median problem (for example, when the distribution F_P is uniform and $p_{\max} > \text{diam}(\mathcal{E})/v$), we conclude that the discrete version of the m-LPIC is, in general, NP-hard.

A possible variant of the gradient ascent (3.9) consists in setting $\dot{g}_j = 0$, when $G \in \Gamma_{\text{coinc}}$, only for the points that are co-located; if the points are co-located, the Voronoi diagram is computed by considering the co-located points as a single point. Such variant is still guaranteed to asymptotically converge to the union of Γ_{coinc} and the set of critical points of $R_m(G, \mathcal{E})$; moreover, it is amenable to distributed implementation, since the gradient ascent law is then distributed over the dual of the Voronoi diagram, i.e., over the Delaunay graph. This last feature is especially useful when a large network of robotic vehicles is employed.

The simultaneous application of the existence theorem 3.4.2 and the convergence theorem 3.4.5 requires that F_P is differentiable on $\mathbb{R}_{\geq 0}$. Indeed any cumulative distribution function can be approximated with arbitrary accuracy by means of a differentiable function which is also monotone, and bounded between 0 and 1 [35]. Therefore, we argue that a reasonably good solution to the m-LPIC can be computed in most practical scenarios.

Finally, we show in theorem 3.4.6 that if F_P is a convex function on $\mathbb{R}_{\geq 0}$, and m is equal to 1, the function $R_m(G, \mathcal{E})$ is concave, and therefore it is a simple computational task to compute its global maximum, provided it exists (its existence is guaranteed under the assumptions of theorem 3.4.2).

Theorem 3.4.6 (Concavity of $R_m(G, \mathcal{E})$)**.** *Assume that F_P is a convex function on $\mathbb{R}_{\geq 0}$, and that $m = 1$; then $R_m(G, \mathcal{E})$ is a concave function.*

Proof. Let y_1 and y_2 be elements of $\mathcal{E}$ and consider an arbitrary $\mu \in [0,\ 1]$; we have

$$\begin{aligned} R_1(\mu y_1 + (1-\mu)y_2, \mathcal{E}) &= \int_{\mathcal{E}} \left(1 - F_P\left(\frac{\|x - \mu y_1 - (1-\mu)y_2\|}{v}\right)\right) \frac{1}{|\mathcal{E}|}\, dx \\ &= \int_{\mathcal{E}} \left(1 - F_P\left(\frac{\|\mu(x-y_1) + (1-\mu)(x-y_2)\|}{v}\right)\right) \frac{1}{|\mathcal{E}|}\, dx \\ &\geq \int_{\mathcal{E}} \left(1 - F_P\left(\frac{\mu\,\|(x-y_1)\| + (1-\mu)\,\|(x-y_2)\|}{v}\right)\right) \frac{1}{|\mathcal{E}|}\, dx \\ &\geq \int_{\mathcal{E}} \left(1 - \left(\mu F_P\left(\frac{\|(x-y_1)\|}{v}\right) + (1-\mu) F_P\left(\frac{\|(x-y_2)\|}{v}\right)\right)\right) \frac{1}{|\mathcal{E}|}\, dx \\ &= \mu\, R_1(y_1, \mathcal{E}) + (1-\mu)\, R_1(y_2, \mathcal{E}), \end{aligned} \tag{3.10}$$

and the claim is proven. □

3.5 An Optimal Light Load Policy

In this section we propose and analyze a policy that requires the least amount of information and is optimal in light load; this result holds for any instance $\mathcal{I}$ in which F_P satisfies the (mild) assumptions of theorem 3.4.2.

3.5.1 The policy

The Nearest-Depot Assignment (NDA) policy is described in algorithm 1 (note that this policy only requires the knowledge of F_P; moreover, it is required that $R_m(G, \mathcal{E})$ has a global maximum).

Algorithm 1: **Nearest-Depot Assignment (NDA) Policy**

Assumes: m service vehicles.

1 Let $G \doteq \arg\max_{\tilde{G} \in \mathcal{E}^m} R_m(\tilde{G}, \mathcal{E})$ (if there are multiple maxima, pick one arbitrarily), and let g_k be the location of the depot for the kth vehicle, $k \in \{1, \ldots, m\}$.

2 Assign a newly arrived demand to the vehicle whose depot is the nearest to that demand's location, and let Q_k be the set of outstanding (in the sense of section 3.2.2) demands assigned to vehicle k.

3 **foreach** k **do**

4 **if** the set Q_k is empty **then**

5 Move to g_k if the vehicle is not at its depot, otherwise stop.

6 **else**

7 Visit demands in Q_k in first-come, first-served order, by taking the shortest path to a demand location.

8 Repeat.

Let m^*_{NDA} be the minimum number of vehicles required by the NDA policy to provide a success factor larger than or equal to ϕ^{d}. Next theorem shows on a theoretical level the

optimality of the NDA policy in light load, and on a practical level how to compute m^*_{NDA} for low values of λ. Recall that by light-load condition we refer to the case $\varrho \to 0^+$, in the sense that $\lambda \to 0^+$.

Theorem 3.5.1 (Optimality of NDA policy)**.** *Consider an instance $\mathcal{I}$ with F_P satisfying the assumptions of theorem 3.4.2, and any of the possible information structures. Then, for almost all values of $\phi^{\mathrm{d}} \in (0,\,1)$ (i.e., except for a set of measure zero) the NDA policy is optimal in light load, i.e.,*

$$\limsup_{\lambda \to 0^+} \frac{m^*_{\mathrm{NDA}}(\lambda)}{m^*(\lambda)} = 1.$$

Proof. Define the countable set $\mathcal{R} \doteq \{R^*_m(\mathcal{E}) \,|\, m \in \mathbb{N}\}$; by definition of $R^*_m(\mathcal{E})$, we have $\inf \mathcal{R} \geq 0$, and $\sup \mathcal{R} \leq 1$. Consider any desired success factor $\phi^{\mathrm{d}} \in (0,\,1) \setminus \mathcal{R}$; note that $\mathcal{R}$ is a countable set, hence its measure is zero (in other words, we are leaving out a zero-measure set of possible success factors).

Assume that $\underline{m}$ vehicles execute the NDA policy, where $\underline{m}$ is the solution to the minimization problem (3.7). (Note that $\underline{m}$ is independent of λ.) Consider any $\lambda \in \mathbb{R}_{>0}$, and define the event:

$$E_j \doteq \{\text{all vehicles at their depots at the arrival epoch of the } j\text{th demand}\};$$

the acceptance probability for demand j satisfies the inequality:

$$\mathbb{P}\,[W_j < P_j] \geq \mathbb{P}\,[W_j < P_j |\, E_j] \cdot \mathbb{P}\,[E_j].$$

Assume that $t_j - t_{j-1}$, $j \in \mathbb{N}$, is larger than $p_{\max} + s_{\max} + \mathrm{diam}(\mathcal{E})/v$; then, the jth demand finds, surely, all vehicles idling at their depots. Hence, we can lower bound $\mathbb{P}\,[E_j]$, $j \in \mathbb{N}$, according to:

$$\mathbb{P}\,[E_j] \geq \mathbb{P}\,[t_j - t_{j-1} > p_{\max} + s_{\max} + \mathrm{diam}(\mathcal{E})/v] = \exp(-\lambda(p_{\max} + s_{\max} + \mathrm{diam}(\mathcal{E})/v));$$

note that this bound is independent of j. Conditioning on the event E_j, all vehicles are at their depots, and therefore

$$\mathbb{P}\,[W_j < P_j |\, E_j] = \mathbb{P}\left[\min_{k \in \{1,\ldots,\underline{m}\}} \tfrac{\|X_j - g_k\|}{v} < P_j\right] = R_{\underline{m}}(G, \mathcal{E}) = R^*_{\underline{m}}(\mathcal{E}).$$

Hence, we obtain, for every $\lambda \in \mathbb{R}_{>0}$,

$$\lim_{j \to +\infty} \mathbb{P}\,[W_j < P_j] \geq R^*_{\underline{m}}(\mathcal{E})\, \exp(-\lambda(p_{\max} + s_{\max} + \mathrm{diam}(\mathcal{E})/v)).$$

From the definition of $\underline{m}$ (see equation (3.7)), and from the fact that $\phi^{\mathrm{d}} \in (0,\,1) \setminus \mathcal{R}$ (hence $R^*_{\underline{m}}(\mathcal{E}) = \phi^{\mathrm{d}}$ is impossible), it follows that $R^*_{\underline{m}}(\mathcal{E}) > \phi^{\mathrm{d}}$. Thus, we conclude that, given $\underline{m}$ vehicles, there exists $\Lambda > 0$ such that $\phi_{\mathrm{NDA}}(\lambda) \geq \phi^{\mathrm{d}}$ for all $\lambda < \Lambda$. Therefore, there exists $\Lambda > 0$ such that $\underline{m} \geq m^*_{\mathrm{NDA}}(\lambda)$ for all $\lambda < \Lambda$; hence, by applying theorem 3.4.1, we obtain $\limsup_{\lambda \to 0^+} \frac{m^*_{\mathrm{NDA}}(\lambda)}{m^*(\lambda)} \leq \limsup_{\lambda \to 0^+} \frac{\underline{m}}{\underline{m}} = 1$. This concludes the proof. □

3.5.2 Discussion and simulations

It is natural to wonder if $\phi^{\mathrm{d}} \to 1^-$ implies $m^* \to +\infty$; we now have the tools to show that, in general, this is not the case. Consider any of the possible information structures;

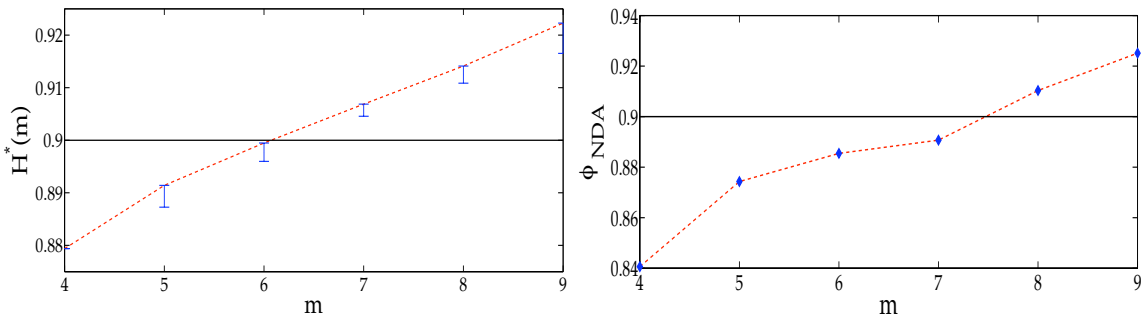


Figure 3-2: Left Figure: Approximate values for R^*_m (the bars indicate the range of values obtained by maximizing R_m). Right Figure: Experimental values of ϕ_{NDA}. The desired success factor is $\phi^{\text{d}} = 0.9$.

moreover, let $\mathcal{I}$ be an instance where F_P satisfies the assumptions of theorem 3.4.2, and the support of F_P is $[\text{diam}(\mathcal{E})/v, +\infty)$. It is easy to see that $R^*_1(\mathcal{E}) = 1$; then, by using the same arguments as those in the proof of theorem 3.5.1, one can show that there exists $\Lambda > 0$ such that for all $\lambda < \Lambda$ it holds $m^* = 1$, for any $\phi^{\text{d}} \in (0\,, 1)$. This example shows that, in general, $\phi^{\text{d}} \to 1^-$ does *not* imply that $m^* \to +\infty$.

We next provide some simulation results for the NDA policy. We consider patience times that are uniformly distributed in the interval [0, 1.6]; moreover, the arrival rate is $\lambda = 5$, the workspace is the unit square, the vehicles' velocity is $v = 1$, and $s_{\max} = 0$ (i.e., there is no on-site service requirement). Finally, we consider a desired success factor $\phi^{\text{d}} = 0.9$. To find a lower bound on the required number of vehicles, we solve the optimization problem (3.7); in particular, starting from $m = 1$, we compute $R^*_m(\mathcal{E})$ until $R^*_m(\mathcal{E}) \geq \phi^{\text{d}}$. The solution to the m-LPIC, i.e., the value $R^*_m(\mathcal{E})$, is approximately computed for each m by performing the gradient-ascent law (3.9) starting from 10 random initial conditions. In Figure 3-2, the left figure shows the range of values that are obtained by maximizing R_m, for several values of m. It can be noted that for each m the range of values is rather small, in other words the function R_m appears to have maxima whose values are close to each other. From the left figure we estimate (recall that we are using approximate values for $R^*_m(\mathcal{E})$) a lower bound on the required number of vehicles equal to 7. The right figure shows experimental values of ϕ_{NDA} as a function of the number of agents m. It can be noted that the minimum number of vehicles required by the NDA policy to ensure a success factor at least as large as ϕ^{d} is 8, in almost perfect accordance with theorem 3.5.1 (recall that theorem 3.5.1 formally holds only in the limit $\lambda \to 0^+$).

3.6 A Policy for Moderate and Heavy Loads

In this section we propose and analyze a policy that is well defined for any information structure and for any instance $\mathcal{I}$, however it is particularly tailored for the *least* informative case and is most effective in moderate and heavy loads. The Batch (B) policy is described in algorithm 2.

3.6.1 Analysis of the policy

Next theorem provides an upper bound on the minimum number of vehicles required by the Batch policy to achieve a success factor larger than or equal to ϕ^{d}, under the assumption

Algorithm 2: **Batch (B) Policy**

Assumes: m service vehicles.

Partition $\mathcal{E}$ into m equal area regions $\mathcal{E}_k$, $k \in \{1, \dots, m\}$, and assign one vehicle to each region.

Assign a newly arrived demand that falls in $\mathcal{E}_k$ to the vehicle responsible for region k, and let Q_k be the set of locations of outstanding (in the sense of section 3.2.2) demands assigned to vehicle k.

foreach vehicle-region pair k **do**

  if the set Q_k is empty **then**

    Move to the median (the "depot") of $\mathcal{E}_k$ if the vehicle is not there, otherwise stop.

  else

    Compute a TSP tour through all demands in Q_k and vehicle's current position.

    Travel along the TSP tour, by skipping demands that are no longer outstanding.

  Repeat.

$s_{\max} = 0$, and assuming the least informative information structure.

Theorem 3.6.1 (Vehicles required by Batch policy)**.** *Given an instance $\mathcal{I}$ with $s_{max} = 0$, the least informative information structure, and a desired success factor $\phi^{\mathrm{d}} \in (0, 1)$, the Batch policy guarantees a success factor at least as large as ϕ^d if the number of vehicles is equal to or larger than:*

$$\tilde{m} \doteq \min\Big\{m \Big| \sup_{\theta \in \mathbb{R}_{>0}} (1 - F_P(\theta))(1 - 2g(m)/\theta) \geq \phi^{\mathrm{d}}\Big\}, \tag{3.11}$$

where

$$g(m) \doteq \frac{1}{2}\left(\frac{\bar{\beta}^2}{v^2}\,|\mathcal{E}|\,\frac{\lambda}{m^2} + \sqrt{\frac{\bar{\beta}^4}{v^4}\,|\mathcal{E}|^2\,\frac{\lambda^2}{m^4} + 8\,\frac{\bar{\beta}^2}{v^2}\,|\mathcal{E}|\,\frac{1}{m}}\right),$$

and where $\bar{\beta}$ is a constant that depends on the shape *of the service regions.*

Proof. In the Batch policy each region has equal area, and contains a single vehicle. Thus, the m vehicle problem in a workspace of area $|\mathcal{E}|$ has been turned into m *independent* single-vehicle problems, each in a region of area $|\mathcal{E}|/m$, and with Poisson arrivals with rates λ/m. In particular, the well-posedness theorem 3.3.4 holds within each region. The strategy of the proof is then as follows: assuming that m vehicles execute the Batch policy, in part 1) we lower bound the limiting acceptance probability within each region; in other words, we lower bound $\lim_{j_k \to +\infty} \mathbb{P}\left[W_{j_k} < P_{j_k}\right]$, where j_k is the jth demand that falls in region k. Then, in part 2), we lower bound the limiting acceptance probability within the entire workspace, and we conclude the proof.

Part 1): Acceptance probability within a region. Consider a region k, $k \in \{1, \dots, m\}$. For simplicity of notation, we shall omit the label k (e.g., instead of j_k, we simply use j to denote the jth demand that falls in region k). We refer to the time instant in which the vehicle assigned to the region computes the ith, $i \in \mathbb{N}_0$, TSP tour as the epoch i of the policy; for $i \in \mathbb{N}$, we refer to the time interval between epoch $(i-1)$ and the time instant

in which the vehicle visits the last demand along the $(i-1)$th TSP tour (possibly skipping some demands) as the ith busy period, and we denote its length with B_i; similarly, we refer to the time interval between epoch $(i-1)$ and epoch i as the ith busy cycle. Let n_i, $i \in \mathbb{N}$, be the number of demands arrived during the ith busy period; we let $n_0 = 0$ (see Figure 3-3).

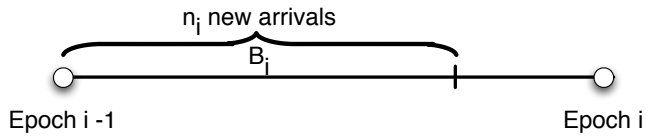


Figure 3-3: Definition of epoch and busy period for the Batch policy.

The number of demands' locations visited during the $(i+1)$th busy period, $i \in \mathbb{N}_0$, is almost surely no larger than $\max(n_i, 1)$; in particular, it may happen that during the ith busy period there are no arrivals, and thus the vehicle waits for a new demand and immediately provides service to it (recall, also, that the arrival process to each region is Poisson, and thus the probability of "bulk" arrivals is zero). Define $\bar{\beta} \doteq \max_{k\in\{1,\dots,m\}} \beta_{\mathcal{E},k}$, where $\beta_{\mathcal{E},k}$ is the characteristic constant of region k (see equation (2.8)); by the deterministic inequality for a TSP tour through n points (see equation (2.8)), we have (recall that the area of the region is $|\mathcal{E}|/m$, and that $s_{\max} = 0$)

$$B_{i+1} \leq \frac{\bar{\beta}}{v}\sqrt{\frac{|\mathcal{E}|}{m}}\sqrt{\max(n_i,\,1)+1}, \quad \text{almost surely;} \tag{3.12}$$

the $+1$ is needed to take into consideration the vehicle's starting position. By simple inductive arguments, it is immediate to show that both $\mathbb{E}[n_i]$ and $\mathbb{E}\left[\sqrt{n_i}\right]$ are finite; hence, by taking expectation in equation (3.12), and by applying Jensen's inequality for concave functions in the form $\mathbb{E}\left[\sqrt{X}\right] \leq \sqrt{\mathbb{E}[X]}$, we get

$$\mathbb{E}[B_{i+1}] \leq \frac{\bar{\beta}}{v}\sqrt{\frac{|\mathcal{E}|}{m}}\sqrt{\mathbb{E}[\max(n_i,1)]+1} \leq \frac{\bar{\beta}}{v}\sqrt{\frac{|\mathcal{E}|}{m}}\sqrt{\mathbb{E}[n_i]+2}.$$

By applying the law of iterated expectation, it is easy to show that the expected number of demands that arrive in the region during the ith busy period, i.e., $\mathbb{E}[n_i]$, is equal to $(\lambda/m)\,\mathbb{E}[B_i]$. Then, we obtain the following recurrence relation

$$\mathbb{E}[B_{i+1}] \leq \frac{\bar{\beta}}{v}\sqrt{\frac{|\mathcal{E}|}{m}}\sqrt{\frac{\lambda}{m}\,\mathbb{E}[B_i]+2}.$$

This recurrence relation allows to find an upper bound on $\limsup_{i\to+\infty}\mathbb{E}[B_i]$; indeed, it is easy to show that $\limsup_{i\to+\infty}\mathbb{E}[B_i] \leq g(m)$. We are now in a position to lower bound the limiting acceptance probability in region k. Consider, in steady state, a random tagged demand; let $\hat{W}$ be its waiting time, and $\hat{P}$ be its patience time. Moreover, let $\hat{I}$ be the epoch that immediately follows the arrival of the tagged demand. By the law of total probability, we have, for any $\theta \in \mathbb{R}_{>0}$,

$$\mathbb{P}\left[\hat{W} < \hat{P}\right] \geq \mathbb{P}\left[\hat{W} < \hat{P} |\, B_{\hat{I}} + B_{\hat{I}+1} < \theta\right]\mathbb{P}\left[B_{\hat{I}} + B_{\hat{I}+1} < \theta\right].$$

Since, from the definition of the Batch policy, $\hat{W} \leq B_{\hat{I}} + B_{\hat{I}+1}$ surely, we have

$$\mathbb{P}\left[\hat{W} < \hat{P} | \, B_{\hat{I}} + B_{\hat{I}+1} < \theta\right] \geq \mathbb{P}\left[\theta < \hat{P} | \, B_{\hat{I}} + B_{\hat{I}+1} < \theta\right] = \mathbb{P}\left[\theta < \hat{P}\right] = 1 - F_P(\theta);$$

in the previous chain of inequalities, the removal of the conditioning on the event $\{B_{\hat{I}} + B_{\hat{I}+1} < \theta\}$ is possible since, under the least informative information structure, the value of $B_{\hat{I}} + B_{\hat{I}+1}$ does not provide *any* information on the value of $\hat{P}$. Then, by collecting the previous results and applying Markov's inequality, we obtain

$$\begin{aligned} \mathbb{P}\left[\hat{W} < \hat{P}\right] &\geq (1 - F_P(\theta)) \, \mathbb{P}\left[B_{\hat{I}} + B_{\hat{I}+1} < \theta\right] \\ &\geq (1 - F_P(\theta)) \left(1 - \frac{\mathbb{E}\left[B_{\hat{I}}\right] + \mathbb{E}\left[B_{\hat{I}+1}\right]}{\theta}\right) \geq (1 - F_P(\theta)) \, (1 - 2g(m)/\theta). \end{aligned} \tag{3.13}$$

Since the previous chain of inequalities holds for all $\theta \in \mathbb{R}_{>0}$, we obtain $\mathbb{P}\left[\hat{W} < \hat{P}\right] \geq \sup_{\theta \in \mathbb{R}_{>0}} (1 - F_P(\theta)) \, (1 - 2\, g(m)/\theta)$. Hence, we conclude that within region k it holds $\lim_{j \to +\infty} \mathbb{P}\left[W_j < P_j\right] \geq \sup_{\theta \in \mathbb{R}_{>0}} (1 - F_P(\theta)) \, (1 - 2\, g(m)/\theta)$.

Part 2): Acceptance probability within the entire environment. From part 1), we have $\lim_{j_k \to +\infty} \mathbb{P}\left[W_{j_k} < P_{j_k}\right] \geq \sup_{\theta \in \mathbb{R}_{>0}} (1 - F_P(\theta)) \, (1 - 2\, g(m)/\theta)$, $k \in \{1, \ldots, m\}$. Note that this lower bound holds uniformly across the m regions. Hence, it is immediate to conclude that the same lower bound holds for the overall system. Since $\lim_{m \to +\infty} g(m) = 0$, it is clear that it is always possible to choose m so that $\phi_{\mathrm{B}} \geq \phi^{\mathrm{d}}$ (recall that $\mathbb{P}\left[P_j = 0\right] = 0$); in particular, a sufficient number of vehicles is given by the solution to the minimization problem in equation (3.11), and the theorem is proven. □

The upper bound in equation (3.11) is valid under the least informative information structure, and *a fortiori* it is valid under any information structure. Hence, theorem 3.6.1 is valid under *any* information structure.

3.6.2 On the constant $\bar{\beta}$ and the use of asymptotics

To compute $\tilde{m}$ in equation (3.11), one needs to know, at least approximately, the value of $\bar{\beta}$; it is possible to show that when each region is approximately square-shaped, the value of $\bar{\beta}$ is approximately equal to $\sqrt{2}$ [96, page 765]. Furthermore, when λ is "large", one could reasonably use the asymptotic value $\beta_{\mathrm{TSP}} \simeq 0.712$ (see Section 2.4.1) to bound B_{i+1}; it is then possible to show (the proof only requires minor modifications in the proof of theorem 3.6.1) that when λ is "large" one can replace $g(m)$ in equation (3.11) with $\tilde{g}(m) \doteq \beta_{\mathrm{TSP}}^2 \lambda |\mathcal{E}| / (v^2 m^2)$.

Using theorem 3.6.1, we next show a scaling law for the minimum number of vehicles.

3.6.3 Scaling law for the minimum number of vehicles

The scaling of the minimum number of vehicles with respect to λ is characterized by the following theorem.

Theorem 3.6.2 (Scaling law)**.** *When $s_{max} = 0$ the solution to the optimization problem $\mathcal{OPT}$ is $O(\sqrt{\lambda})$ under any information structure.*

Proof. Define $\Theta \doteq \{\theta \in \mathbb{R}_{>0} | \, 1 - F_P(\theta) > 0\}$; under the assumptions, the set Θ is not empty; moreover, we have $0 < \sup \Theta < +\infty$. Let $\bar{\theta} = (1/2) \sup \Theta$. Then, we have

$$\sup_{\theta \in \mathbb{R}_{>0}} (1 - F_P(\theta)) \, (1 - 2\, g(m;\ \lambda)/\theta) \geq (1 - F_P(\bar{\theta})) \, (1 - 2\, g(m;\ \lambda)/\bar{\theta}) \doteq h(m;\ \lambda),$$

where we have made the dependency on λ explicit. Define (note that $1 - F_P(\bar{\theta}) > 0$)

$$m(\lambda) \doteq \left\lceil \frac{\sqrt{\lambda}\, \sqrt{|\mathcal{E}|}\, \bar{\beta}}{v\, \sqrt{\delta}} \right\rceil, \quad \text{where} \quad \delta = \left(1 - \frac{\eta\, \phi^{\mathrm{d}}}{1 - F_P(\bar{\theta})}\right) \frac{\bar{\theta}}{2}, \quad \eta > 1.$$

It is straightforward to show that there exists Λ such that for all $\lambda > \Lambda$ it holds $h(m(\lambda);\ \lambda) \geq \phi^{\mathrm{d}}$. Hence we have, for $\lambda > \Lambda$, $\tilde{m}(\lambda) \leq m(\lambda)$, where $\tilde{m}(\lambda)$ is defined in theorem 3.6.1. Since $m^*(\lambda) \leq \tilde{m}(\lambda)$, we immediately obtain

$$\limsup_{\lambda \to +\infty} \frac{m^*(\lambda)}{\sqrt{\lambda}} \leq \limsup_{\lambda \to +\infty} \frac{\tilde{m}(\lambda)}{\sqrt{\lambda}} \leq \frac{\sqrt{|\mathcal{E}|}\, \bar{\beta}}{v\, \sqrt{\delta}}.$$

Since the Batch policy is well-defined for any information structure, we conclude that when $s_{\max} = 0$ the solution to the optimization problem $\mathcal{OPT}$ is $O(\sqrt{\lambda})$ under any information structure. □

3.6.4 Simulations

We consider patience times that assume either the value 0.8 with 50% probability, or the value 1.6 with the remaining 50% probability; in other words, there are two types of demands, and one type is significantly more impatient than the other one. The arrival rate is $\lambda = 200$, the workspace is the unit square, the vehicles' velocity is $v = 1$, and $s_{\max} = 0$. Finally, we consider a desired success factor $\phi^{\mathrm{d}} = 0.9$. By solving the minimization problem in equation (3.11) (with $\tilde{g}(m)$ instead of $g(m)$ since λ is "large"), we find $\tilde{m} = 36$. Figure 3-4 shows experimental values of ϕ_{B} as a function of the number of agents m. It can be noted that when $m = \tilde{m}$ the experimental success factor ϕ_{B} is larger than ϕ^{d}, in accordance with theorem 3.6.1. However, it is also possible to observe that the Batch policy is able to guarantee a success factor larger than ϕ^{d} with a number of vehicles as low as 27; this is expected, since the techniques used in the proof of theorem 3.6.1 (e.g., Markov's inequality) lead to a conservative result.

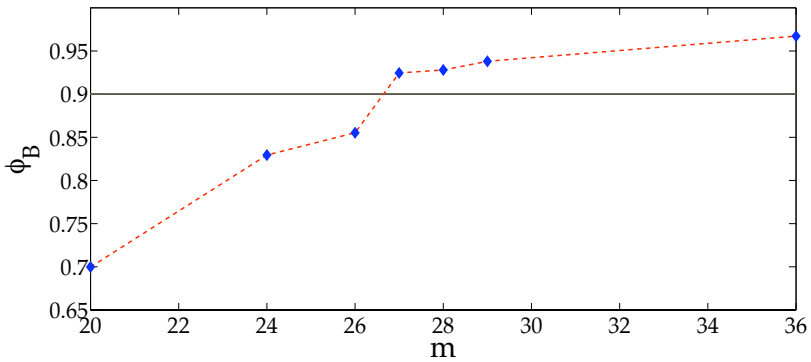


Figure 3-4: Experimental values of ϕ_{B}. The desired success factor is $\phi^{\mathrm{d}} = 0.9$.

3.7 Performance of the Batch Policy with Time Windows

In this section we consider the special case where (i) the on-site service time is negligible, i.e., $s_{\max} = 0$, (ii) almost all demands are required to be serviced before expiring, i.e., $\phi^d \to 1^-$, (iii) the system is heavily congested, i.e., $\lambda \to +\infty$, and (iv) the deadlines are deterministic, i.e., the patience times are deterministic time windows. In this case it is possible to show that the Batch policy is within a small factor of the optimum in heavy load. Specifically, we have the following result.

Theorem 3.7.1. *Assume an instance $\mathcal{I}$ with $s_{max} = 0$ and deterministic patience times (i.e., $F_P(p)$ is a step function with a single step of magnitude 1), the least informative information structure, and a desired success factor $\phi^d \approx 1$. Then the Batch policy has a constant factor guarantee:*

$$\limsup_{\lambda \to \infty} \frac{m^*_{\mathrm{B}}(\lambda)}{m^*(\lambda)} \leq 3.78, \tag{3.14}$$

where m^*_{B} is the minimum number of vehicles required by the Batch policy to provide a success factor larger than or equal to ϕ^d. In particular, for λ large enough, m^*_{B} is upper bounded by

$$\tilde{m}(\lambda) \doteq \left\lceil \sqrt{\lambda}\Big(\beta_{\mathrm{TSP}}\sqrt{\frac{2}{p_{\max}}} + \delta\Big) \right\rceil \tag{3.15}$$

where δ is an arbitrarily small constant.

The proof of theorem 3.7.1 relies on arguments similar to those presented in the proof of theorem 3.6.1. The details can be found in the journal article [77].

3.8 On the Assumptions of the Model

In light of our analysis and results, we discuss the assumptions we made in setting up the problem.

Poisson arrivals : The assumption of Poisson arrivals is exploited in section 3.3 (where the i.i.d. property of the inter-arrival intervals is necessary to establish the regenerative nature of the relevant processes), in equation (3.12) (where we need the property that the probability of bulk arrivals is zero), and whenever we compute the expected number of arrivals in a time interval t. Hence, we conclude that the light-load lower bound holds for every possible arrival process; moreover, the ergodic theorem 3.3.3 and the limit theorem 3.3.4 hold under the more general assumption that the arrival process is a renewal process. The analysis of the Batch policy carries over with minor modifications to the case of non-arithmetic, renewal arrival processes (bulk arrivals can be easily included in equation (3.12); moreover, Blackwell's theorem gives a simple expression for the expected number of arrivals in a certain time interval).

Uniform demand distribution As can be easily checked, the ergodic theorem 3.3.3 and the limit theorem 3.3.4 hold regardless of the type of demand distribution, hence they hold also when the spatial distribution for demand locations is not uniform. The light-load lower bound and its subsequent analysis carry over with minor modification to the case of non-uniform demand distribution; the only difference is that now the term $1/|\mathcal{E}|$ (which is the spatial density in the uniform case) should be replaced with the

density of the general demand distribution. The analysis of the Batch policy holds, with no modifications, also in the case of non-uniform demand distribution (however, in heavy load, the term $\bar{\beta}\sqrt{|\mathcal{E}|/m}$ should be replaced with $\beta_{\mathrm{TSP}} \int_{\mathcal{E}} \sqrt{f(x)}\, dx$, where f is the density of the demand distribution).

Extension to higher dimensions All of the results extend easily to subsets $\mathcal{E}$ of $\mathbb{R}^d$ for arbitrary dimension d; in the interest of brevity, we do not discuss the details.

Depots The assumption that a vehicle returns to its depot if it has no demands to service ensures that all the relevant processes are regenerative. If, in some applications, this assumption is not met, one should look for an equivalent condition that still ensures the existence of regeneration points.

Surely-finite patience and service times Finiteness of patience and service times is exploited heavily throughout the chapter, most importantly to establish the ergodic theorem 3.3.3 and the limit theorem 3.3.4; hence, such assumption can be hardly removed. However, surely-finite patience and service times are probably an adequate model in practice.

Other assumptions The assumption $\mathbb{P}[P_j = 0] = 0$ can be relaxed, with the understanding that, if $\mathbb{P}[P_j = 0] > 0$, the desired success factor should lie in the interval $(0,\, 1 - \mathbb{P}[P_j = 0])$.

3.9 Conclusion

In this chapter we studied a dynamic vehicle routing problem where demands have stochastic deadlines on their waiting times. There are numerous important extensions open for further research. First, in this chapter we found a lower bound for the most informative case and we characterized two service policies that require the least amount of information; it would be very interesting to find lower bounds and study policies that are specific to each particular information structure. Second, our lower bound does not capture the dependency on λ and thus it is generally highly inaccurate for large values of the arrival rate. Third, we believe that the analysis of the Batch policy is very conservative and might be improved. All these problems provide nontrivial challenges and might require techniques significantly different from those presented in this chapter.

Chapter 4

DVR with Priorities

In this chapter we explore another typical aspect of routing problems for robotic vehicles, namely prioritizing among demands. Priority queueing is indeed a classical problem in queueing theory [53]. In the simplest setup, customers arrive at a single server sequentially over time. Each customer is a member of either the high-priority, or the low-priority class. High priority customers and low priority customers form separate queues. The goal is to provide the highest possible quality of service to the high priority queue (Q_1) while maintaining stability of the low priority queue (Q_2). That is, the goal is to minimize the expected system time for high-priority customers while keeping the length of the low-priority queue finite. When both the customer inter-arrival times and the customer service times are distributed exponentially, the preemptive priority policy is known to be optimal [53]:

> When Q_1 is nonempty, serve high priority customers; when Q_1 is empty, serve low-priority customers. If a high priority customer arrives while serving Q_2, then preempt service and immediately begin serving the high-priority customer.

A more general two-class queueing problem is to minimize a convex combination of the system times for high- and low-priority customers

$$c\,\overline{T}_1 + (1-c)\,\overline{T}_2, \quad \text{where } c \in (0,1).$$

In this case an optimal policy can be created by using a mixed policy that spends fraction c of the time serving Q_1 as the high-priority queue, and fraction $(1-c)$ serving Q_2 as though it is the high-priority queue [30]. The set of achievable system times has also been studied in the more general setting of queueing networks [13].

In this chapter we consider an n-class, m-service-vehicle spatial queueing problem, called *dynamic vehicle routing with priority classes*. Demands for service arrive sequentially over time in a compact environment $\mathcal{E}$ in the plane. Each demand is a member of one of n priority classes. Upon arrival, each demand assumes a location in $\mathcal{E}$, and requires a class-dependent amount of on-site service time. To service a demand, one of the m vehicles must travel to the demand location and perform the on-site service. If we specify a policy by which the vehicles serve demands, then the expected service time for demands of class α, denoted $\overline{T}_\alpha$, is the expected amount of time between a demand's arrival and its service completion. Then, given convex combination coefficients $c_1, \ldots, c_n > 0$, the goal is to find the vehicle routing policy that minimizes $c_1\,\overline{T}_1 + \cdots + c_n\,\overline{T}_n$. By increasing the coefficients for certain classes, a higher priority level can be given to their demands. This problem has

important applications in areas such as UAV surveillance, where targets are given different priority levels based on their urgency or potential importance [11].

The main contribution of this chapter is to introduce dynamic vehicle routing with priority classes. We derive a lower bound on the achievable values of the convex combination of system times, and propose a novel policy in which each class of demands is served separately from the others. We show that in heavy load the policy performs within a constant factor $2n^2$ of the lower bound. Thus, the constant factor is independent of the number of vehicles, the arrival rates of demands, the on-site service times, and the convex combination coefficients. To establish the constant factor, we proceed in a similar manner as in section 3.6 and develop a system of nonlinear inequality-based recursive equations on the expected number of outstanding demands. We then utilize a novel proof technique to upper bound the limiting number of outstanding demands, which relies on constructing a set of linear equality-based recursive equations to bound trajectories. We present an improvement on the policy in which classes of similar priority are merged together. We also perform extensive simulations and introduce an effective heuristic improvement called the tube heuristic.

The chapter is organized as follows. In section 4.1 we formalize the problem and in section 4.2 we derive a lower bound on the achievable system time. In section 4.3 we introduce and analyze the Separate Queues policy, and present the improvements given by *queue merging* and the *tube heuristic*. Finally, in section 4.4 we present simulation results and in section 4.5 we draw our conclusions.

The work in this chapter was performed in collaboration with Stephen L. Smith, and is based on the journal article [91] and the preliminary conference papers [82, 90].

4.1 Problem Statement

In this section we formalize the dynamic vehicle routing problem with priority classes.

4.1.1 Problem statement

Consider a compact environment $\mathcal{E}$ in the plane with area $|\mathcal{E}|$. The environment contains m vehicles, each with maximum speed v. Demands of type $\alpha \in \{1, \ldots, n\}$ (also called α-demands) arrive in the environment according to a Poisson process with rate λ_α. Upon arrival, demands assume an independently and uniformly distributed location in $\mathcal{E}$. An α-demand is serviced when the vehicle spends an on-site service time at the demand location, which is generally distributed with finite mean $\bar{s}_\alpha$.

Consider the arrival of the jth α-demand. The system time for the jth demand, $T_{j,\alpha}$, is the time elapsed between its arrival and its service completion. The wait time is defined as $W_{j,\alpha} = T_{j,\alpha} - s_{j,\alpha}$, where $s_{j,\alpha}$ is the on-site service time required by demand j. A policy for routing the vehicles is said to be *stable* if the expected number of demands in the system for each class is bounded uniformly at all times. A necessary condition for the existence of a stable policy is

$$\varrho \doteq \frac{1}{m} \sum_{\alpha=1}^{n} \lambda_\alpha \bar{s}_\alpha < 1. \tag{4.1}$$

As discussed in section 2.5, it is in general difficult to study a queueing system for all values of $\varrho \in [0, 1)$, and a common technique is to focus on the limiting regimes of $\varrho \to 1^-$, referred to as the *heavy-load* regime, and $\varrho \to 0^+$, referred to as the *light-load* regime.

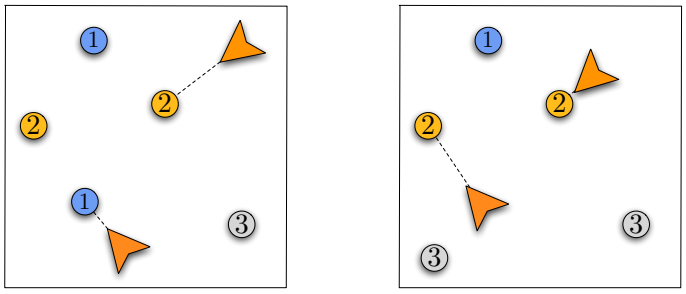


Figure 4-1: A depiction of the problem for two vehicles and three priority classes. Left figure: One vehicle is moving to a class 1 demand, and the other to a class 2 demand. Right figure: The bottom vehicle has serviced the class 1 demand and is moving to a class 2 demand. A new class 3 demand has arrived.

Given a stable policy π the steady-state system time is defined as

$$\overline{T}_{\pi,\alpha} \doteq \lim_{j\to+\infty} \mathbb{E}\left[T_{j,\alpha}\right],$$

and the steady-state waiting time is $\overline{W}_{\pi,\alpha} \doteq \overline{T}_{\pi,\alpha} - \bar{s}_\alpha$. Thus, for a stable policy π, the *average system time per demand* is

$$\overline{T}_\pi = \frac{1}{\Lambda}\sum_{\alpha=1}^{n} \lambda_\alpha \overline{T}_{\pi,\alpha},$$

where $\Lambda \doteq \sum_{\alpha=1}^{n} \lambda_\alpha$. The average system time per demand is the standard cost functional for queueing systems with multiple classes of demands. Notice that we can write $\overline{T}_\pi = \sum_{\alpha=1}^{n} c_\alpha \overline{T}_{\pi,\alpha}$ with $c_\alpha = \lambda_\alpha/\Lambda$. Thus, we can model priority among classes by allowing any convex combination of $\overline{T}_1, \ldots, \overline{T}_n$. If $c_\alpha > \lambda_\alpha/\Lambda$, then the system time of α-demands is being weighted more heavily than in the average case. Thus, the quantity $c_\alpha\Lambda/\lambda_\alpha$ gives the priority of α demands compared to that given in the average system time case. Without loss of generality we can assume that priority classes are labeled so that

$$\frac{c_1}{\lambda_1} \geq \frac{c_2}{\lambda_2} \geq \cdots \geq \frac{c_n}{\lambda_n}, \tag{4.2}$$

implying that if $\alpha < \beta$ for some $\alpha, \beta \in \{1, \ldots, n\}$, then the priority of α-demands is at least as high as that of β-demands. With these definitions, we are now ready to state our problem.

> **Problem Statement:** Let $\mathcal{P}$ be the set of all causal, stable and stationary policies for dynamic vehicle routing with priority classes. Given the coefficients $c_\alpha > 0$, $\alpha \in \{1, \ldots, n\}$, with $\sum_{\alpha=1}^{n} c_\alpha = 1$, and satisfying expression (4.2), let $\overline{T}_\pi \doteq \sum_{\alpha=1}^{n} c_\alpha \overline{T}_{\pi,\alpha}$ be the cost of policy $\pi \in \mathcal{P}$. Then, the problem is to determine a vehicle routing policy π^*, if one exists, such that
>
> $$\overline{T}_{\pi^*} = \inf_{\pi\in\mathcal{P}} \overline{T}_\pi. \tag{4.3}$$

We let $\overline{T}^*$ denote the right-hand side of equation (4.3). A policy π for which $\overline{T}_\pi/\overline{T}^*$

is bounded has a *constant-factor guarantee*. If $\limsup_{\varrho \to 1^-} \overline{T}_\pi / \overline{T}^* = \vartheta < +\infty$, then the policy π has a *heavy-load constant-factor guarantee* of ϑ. In this chapter we focus on the heavy-load regime, and look for policies with a heavy-load constant-factor guarantee that is *independent* of the number of vehicles, the arrival rates of demands, the on-site service times, and the convex combination coefficients. In the light-load regime, existing policies for the dynamic traveling repairman can be used, as is summarized in the following remark.

Remark 4.1.1 (Light-load regime)**.** *In light load, $\varrho \to 0^+$, optimal policies have been developed for the dynamic traveling repairman problem (i.e., the single-class dynamic vehicle routing problem, see section 2.5). In fact, by following the proof in [15], one can show that the SQM policy, described in section 2.5.3, is an optimal policy for dynamic vehicle routing with priority classes. The proof of this statement is rather straightforward, and we refer interested readers to [15] for details.* •

4.2 Lower Bound in Heavy Load

In this section we present two lower bounds on the system time in equation (4.3). The first holds only in heavy load (i.e., as $\varrho \to 1^-$), while the second (less tight) bound holds for all ϱ.

Theorem 4.2.1 (Heavy-load lower bound)**.** *For every routing policy π,*

$$\overline{T}_\pi \geq \frac{\beta_{\mathrm{TSP}}^2 |\mathcal{E}|}{2m^2 v^2 (1-\varrho)^2} \sum_{\alpha=1}^{n} \left(c_\alpha + 2 \sum_{j=\alpha+1}^{n} c_j \right) \lambda_\alpha \quad \textit{as } \varrho \to 1^-, \tag{4.4}$$

where $c_1, \ldots, c_n$ satisfy expression (4.2)*.*

Proof. Consider a tagged demand j of type α, and let us quantify its total service requirement. The demand requires on-site service time $s_{j,\alpha}$. Let us denote by $d_{j,\alpha}$ the distance from the location of the demand served prior to j, to j's location. In order to compute a lower bound on the wait time, we will allow "remote" servicing of some of the demands. For an α-demand j that can be serviced remotely, the travel distance $d_{j,\alpha}$ is zero (i.e., a service vehicle can service the jth α-demand from any location by simply stopping for the on-site service time $s_{j,\alpha}$). Thus, the waiting time for the modified remote servicing problem provides a lower bound on the waiting time for the problem of interest. To formalize this idea, we introduce the variables $r_\alpha \in \{0,1\}$ for each $\alpha \in \{1,\ldots,n\}$. If $r_\alpha = 0$, then α-demands can be serviced remotely. If $r_\alpha = 1$, then α-demands must be serviced on location. We assume that $r_\alpha = 1$ for at least one $\alpha \in \{1,\ldots,n\}$. Thus, the total service requirement of α-demand j is $r_\alpha\, d_{j,\alpha}/v + s_{j,\alpha}$, where v is the service vehicle speed. The steady-state expected service requirement is $r_\alpha\, \bar{d}_\alpha / v + \bar{s}_\alpha$, where $\bar{d}_\alpha \doteq \lim_{j \to +\infty} \mathbb{E}\,[d_{j,\alpha}]$. In order to maintain stability of the system we must require

$$\frac{1}{m} \sum_{\alpha=1}^{n} \lambda_\alpha \left(\frac{r_\alpha \bar{d}_\alpha}{v} + \bar{s}_\alpha \right) < 1. \tag{4.5}$$

Applying the definition of ϱ in equation (4.1), we write inequality (4.5) as

$$\sum_{\alpha=1}^{n} r_\alpha \lambda_\alpha \bar{d}_\alpha < (1-\varrho) m v. \tag{4.6}$$

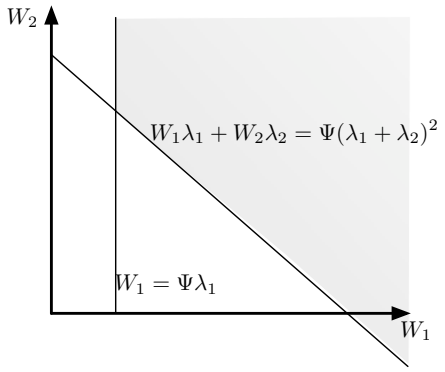


Figure 4-2: The feasible region of the linear program for 2 queues. When class 1 is of higher priority, the solution is given by the corner. Otherwise, the solution is $-\infty$.

For a stable policy π, let $\overline{N}_\alpha$ represent the steady-state expected number of unserviced α-demands. Then, the expected total number of outstanding demands that require on-site service (i.e., cannot be serviced remotely) is given by $\sum_{i=1}^n r_i \overline{N}_i$. We now apply a result from the dynamic traveling repairman problem (see [108], page 23) which states that in heavy load ($\varrho \to 1^-$), if the steady-state number of outstanding demands is N, then a lower bound on expected travel distance between demands is $(\beta_{\text{TSP}}/\sqrt{2})\sqrt{|\mathcal{E}|/N}$. Applying this result we have that

$$\bar{d}_\alpha \geq \frac{\beta_{\text{TSP}}}{\sqrt{2}} \sqrt{\frac{|\mathcal{E}|}{\sum_i r_i \overline{N}_i}} \doteq \bar{d}, \tag{4.7}$$

for each $\alpha \in \{1, \ldots, n\}$. Combining inequalities (4.6) and (4.7),

$$\frac{\sum_\alpha r_\alpha \lambda_\alpha}{mv(1-\varrho)} < \frac{1}{\bar{d}}.$$

Applying the definition of $\bar{d}$, squaring both sides, and rearranging we obtain

$$\frac{\beta_{\text{TSP}}^2}{2} \frac{|\mathcal{E}|(\sum_\alpha r_\alpha \lambda_\alpha)^2}{m^2 v^2 (1-\varrho)^2} < \sum_\alpha r_\alpha \overline{N}_\alpha.$$

From Little's law, $\overline{N}_\alpha = \lambda_\alpha \overline{W}_\alpha$ for each $\alpha \in \{1, \ldots, n\}$, and thus

$$\sum_\alpha r_\alpha \lambda_\alpha \overline{W}_\alpha > \frac{\beta_{\text{TSP}}^2}{2} \frac{|\mathcal{E}|}{m^2 v^2 (1-\varrho)^2} \left(\sum_\alpha r_\alpha \lambda_\alpha \right)^2. \tag{4.8}$$

Recalling that $\overline{W}_\alpha = \overline{T}_\alpha - \bar{s}_\alpha$ and $r_\alpha \in \{0, 1\}$ for each $\alpha \in \{1, \ldots, n\}$, we see that expression (4.8) gives us $2^n - 1$ constraints on the feasible values of $\overline{T}_{\pi,1}, \ldots, \overline{T}_{\pi,n}$. Hence, a lower bound on $\overline{T}^*$ can be found by minimizing $\sum_{\alpha=1}^n c_\alpha \overline{W}_\alpha$ subject to the constraints in expression (4.8). We can lower bound the solution to the optimization problem by minimizing the cost function subject to only a subset of the $2^n - 1$ constraints. In particular, consider the following linear program

$$\textbf{minimize} \quad \sum_{\alpha=1}^{n} c_\alpha \overline{W}_\alpha,$$

$$\textbf{subject to} \quad \begin{bmatrix} \lambda_1 & 0 & 0 & \cdots & 0 \\ \lambda_1 & \lambda_2 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & & 0 \\ \lambda_1 & \lambda_2 & \lambda_3 & \cdots & \lambda_n \end{bmatrix} \begin{bmatrix} \overline{W}_1 \\ \overline{W}_2 \\ \vdots \\ \overline{W}_n \end{bmatrix} \geq \Psi \begin{bmatrix} \lambda_1^2 \\ (\lambda_1+\lambda_2)^2 \\ \vdots \\ (\lambda_1+\cdots+\lambda_n)^2 \end{bmatrix},$$

where

$$\Psi \doteq \frac{\beta_{\text{TSP}}^2}{2} \frac{|\mathcal{E}|}{m^2 v^2 (1-\varrho)^2}.$$

The above problem is feasible (see Figure 4-2), it has only one basic feasible solution, and it is of the form: minimize $\mathbf{c}^T\mathbf{W}$ subject to $A\mathbf{W} \geq \mathbf{b}$. Thus, either the problem is unbounded, or the solution $\mathbf{W}^*$ is given by the basic feasible solution. To establish boundedness we consider the dual problem: maximize $\mathbf{b}^T\mathbf{y}$ subject to $A^T\mathbf{y} = \mathbf{c}$ and $\mathbf{y} \geq 0$. By the Duality Theorem of Linear Programming [64], if the dual is feasible, then the minimization problem is bounded. To check feasibility of the dual, we solve for $A^T\mathbf{y} = \mathbf{c}$, with $\mathbf{y} \geq 0$, to obtain

$$\begin{aligned} y_\alpha &= \frac{c_\alpha}{\lambda_\alpha} - \frac{c_{\alpha+1}}{\lambda_{\alpha+1}} \geq 0 \quad \text{for all } \alpha \in \{1,\ldots,n-1\}, \\ y_n &= \frac{c_n}{\lambda_n} \geq 0. \end{aligned}$$

Thus, the dual is feasible if and only if the priority classes are labeled as in expression (4.2). When expression (4.2) is satisfied, the minimization problem is bounded, and its solution $(\overline{W}_1^*, \ldots, \overline{W}_n^*)$ is given by

$$\overline{W}_\alpha^* = \frac{\Psi}{\lambda_\alpha}\left((\lambda_1+\cdots+\lambda_\alpha)^2 - (\lambda_1+\cdots+\lambda_{\alpha-1})^2\right) = \Psi\left(\lambda_\alpha + 2\sum_{j=1}^{\alpha-1}\lambda_j\right).$$

(In fact, this is the solution of the full optimization problem with $2^n - 1$ constraints. This fact can be verified, somewhat tediously, by writing the dual of the full problem and directly computing its solution. To shorten the presentation we omit the direct computation and use the above technique.) The optimal value of the cost function, and thus the lower bound on $\overline{T}^*$, is given by

$$\sum_{\alpha=1}^{n} c_\alpha \overline{W}_\alpha^* = \Psi \sum_{\alpha=1}^{n} c_\alpha \left(\lambda_\alpha + 2\sum_{j=1}^{\alpha-1}\lambda_j\right) = \Psi \sum_{\alpha=1}^{n}\left(c_\alpha + 2\sum_{j=\alpha+1}^{n} c_j\right)\lambda_\alpha.$$

Applying the definition of Ψ we obtain the desired result. □

Remark 4.2.2 (Lower bound for all $\varrho \in [0,1)$)**.** *With slight modifications, it is possible to obtain a less tight lower bound valid for all values of ϱ. In the above derivation, the assumption that $\varrho \to 1^-$ is used only in inequality* (4.7)*. It is possible to use, instead, a lower bound valid for all $\varrho \in [0,1)$ (see [15]):*

$$\bar{d}_\alpha \geq \gamma \sqrt{\frac{|\mathcal{E}|}{\sum_\alpha r_\alpha \overline{N}_\alpha + m/2}},$$

where $\gamma = 2/(3\sqrt{2\pi}) \approx 0.266$. *Using this bound we obtain the same linear program as in the proof of Theorem 4.2.1, with the difference that* Ψ *is now a function given by*

$$\Psi(x) \doteq \frac{\gamma^2|\mathcal{E}|}{m^2v^2(1-\varrho)^2}x - \frac{m}{2}.$$

Following the procedure in the proof of Theorem 4.2.1

$$\overline{W}^*_1 = \frac{\gamma^2|\mathcal{E}|}{m^2v^2(1-\varrho)^2}\lambda_1 - \frac{m}{2\lambda_1}$$
$$\overline{W}^*_\alpha = \frac{\gamma^2|\mathcal{E}|}{m^2v^2(1-\varrho)^2}\left(\lambda_\alpha + 2\sum_{j=1}^{\alpha-1}\lambda_j\right),$$

for each $\alpha \in \{2,\dots,n\}$. *Finally, for every policy* π, $\overline{T}_{\pi,\alpha} \geq \overline{W}^*_\alpha + \bar{s}_\alpha$, *and thus*

$$\overline{T}_\pi \geq \frac{\gamma^2|\mathcal{E}|}{m^2v^2(1-\varrho)^2}\sum_{\alpha=1}^{n}\left(\left(c_\alpha + 2\sum_{j=\alpha+1}^{n}c_j\right)\lambda_\alpha\right) - \frac{mc_1}{2\lambda_1} + \sum_{\alpha=1}^{n}c_\alpha\bar{s}_\alpha, \tag{4.9}$$

for all $\varrho \in [0,1)$ *under the labeling in expression (4.2).* •

In the remainder of the chapter we design a policy and establish a constant-factor guarantee relative to the heavy-load lower bound.

4.3 Separate Queues Policy

In this section we introduce and analyze the Separate Queues (SQ) policy. We show that this policy is within a factor $2n^2$ of the lower bound in heavy load.

To present the SQ policy we need some notation. We assume vehicle $k \in \{1,\dots,m\}$ has a service region $\mathcal{E}_k \subseteq \mathcal{E}$, such that $\{\mathcal{E}_k\}_{k=1}^m$ forms a partition of the environment $\mathcal{E}$. In general the partition could be time varying, but for the description of the SQ policy this will not be required. We assume that information on outstanding demands of type $\alpha \in \{1,\dots,n\}$ in region $\mathcal{E}_k$ at time t is summarized as a finite set of demand positions $Q_{k,\alpha}(t)$. Demands of type α with location in $\mathcal{E}_k$ are inserted in the set $Q_{k,\alpha}$ as soon as they are generated. Removal from the set $Q_{k,\alpha}$ requires that service vehicle k moves to the demand location, and provides the on-site service. The SQ policy is described in Algorithm 3. In this algorithm, the probability distribution $\mathbf{p}$ gives a set of parameters which will be used to optimize performance. Without loss of generality, we avoid pathological situations by restricting each p_α to be positive (if $p_\alpha = 0$ for some class α, then the average system time per demand is trivially unbounded).

Figure 4-3 shows an illustrative example of the SQ policy. In the first two frames the vehicle is servicing only class 1 (circle shaped) demands, whereas in the third frame, the vehicle is servicing class 2 (diamond shaped) demands.

4.3.1 Stability analysis of the SQ policy in heavy load

In this section we analyze the SQ policy in heavy load, i.e., as $\varrho \to 1^-$. In the SQ policy each region $\mathcal{E}_k$ has equal area, and contains a single vehicle. Thus, the m vehicle problem in

Algorithm 3: **Separate Queues (SQ) Policy**

Optimize: algorithm performance over probability distribution $\mathbf{p} = [p_1, \ldots, p_n]$, where $p_\alpha > 0$ for each $\alpha \in \{1, \ldots, n\}$.

```
Partition 𝓔 into m equal area regions and assign one vehicle to each region.
foreach vehicle-region pair k do
    if the set ∪_α Q_{k,α} is empty then
        Move vehicle toward the median of its own region until a demand arrives.
    else
        Select Q ∈ {Q_{k,1}, ..., Q_{k,n}} according to p.
        if Q is empty then
            Reselect until Q is nonempty.
        Compute TSP tour through all demands in Q.
        Service Q following the TSP tour, starting at the demand closest to the
        vehicle's current position.
    Repeat.
```

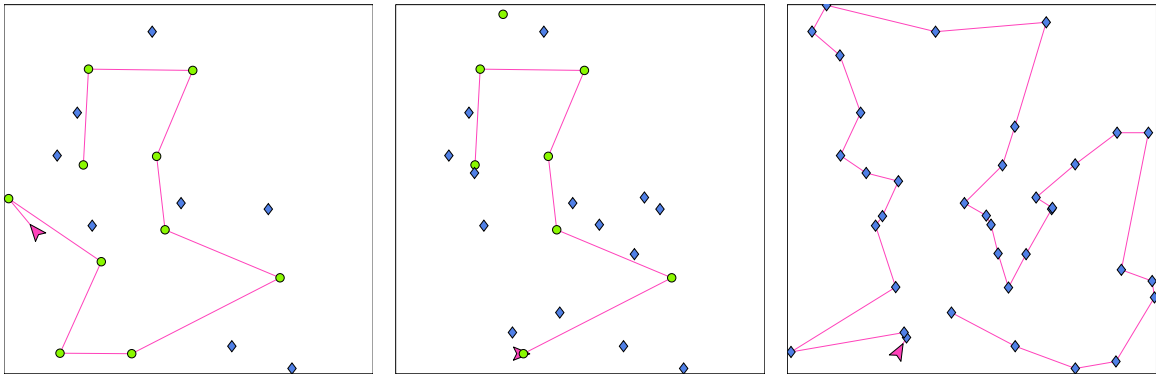


Figure 4-3: A representative simulation of the SQ policy for one vehicle and two priority classes. Circle shaped demands are high priority, and diamond shaped are low priority. The vehicle is marked by a chevron shaped object and TSP tour is shown in a solid line. The left figure shows the vehicle computing a tour through class 1 demands. The center figure shows the vehicle part-way through the class 1 tour and some newly arrived class 2 demands. The right figure shows the vehicle after completing the class 1 tour and computing a new tour through all class 2 demands.

a region of area $|\mathcal{E}|$ has been turned into m independent single-vehicle problems, each in a region of area $|\mathcal{E}|/m$, with arrival rates λ_α/m. To determine the performance of the policy we need only study the performance in a single region k. For simplicity of notation we omit the label k. We refer to the time instant t_i in which the vehicle computes a new TSP tour as the epoch i of the policy; we refer to the time interval between epoch i and epoch $i+1$ as the ith iteration and we will refer to its length as C_i. Finally, let $N_\alpha(t_i) \doteq N_{\alpha,i}$, $\alpha \in \{1,\dots,n\}$, be the number of outstanding α-demands at beginning of iteration i.

The following straightforward lemma, similar to Lemma 1 in [73], will be essential in deriving our main results.

Lemma 4.3.1 (Number of outstanding demands)**.** *In heavy load (i.e., $\varrho \to 1^-$), after a transient, the number of demands serviced in a single tour of the vehicle in the SQ policy is very large with high probability (i.e., the number of demands tends to $+\infty$ with probability that tends to 1, as ϱ approaches 1^-).*

Proof. Consider the case where the vehicle moves with infinite velocity (i.e., $v \to +\infty$); then the system is reduced to an M/G/1 queue (i.e., a queue with exponentially distributed inter-arrival times, generally distributed service times, and a single server; we refer the reader to [53] for more details). The infinite-velocity system has fewer demands (for every $\alpha \in \{1,\dots,n\}$) waiting in queue. A known result on M/G/1 queues [106] states that, after transients, the total number of demands, as $\varrho \to 1^-$, is very large with high probability. Thus, in the SQ policy, the number of demands in all n classes is very large with high probability. In particular, this implies that, after a transient, the number of demands is very large with high probability at the instances when the vehicle starts a new tour. □

Let TS_j be the event that Q_j is selected for service at iteration i of the SQ policy. By the total probability law

$$\mathbb{E}\left[N_{\alpha,i+1}\right] = \sum_{j=1}^{n} p_j \mathbb{E}\left[N_{\alpha,i+1} \,|\, TS_j\right], \quad \alpha \in \{1,\dots,n\},$$

where the conditioning is with respect to the task being performed during iteration i. During iteration i of the policy, demands arrive according to independent Poisson processes. Call $N^{\text{new}}_{\alpha,i}$ the number of α-demands ($\alpha \in \{1,\dots,n\}$) newly arrived during iteration i; then, by definition of the SQ policy

$$\mathbb{E}\left[N_{\alpha,i+1} \,|\, TS_j\right] = \begin{cases} \mathbb{E}\left[N^{\text{new}}_{\alpha,i} \,|\, TS_j\right], & \text{if } \alpha = j \\ \mathbb{E}\left[N_{\alpha,i} \,|\, TS_j\right] + \mathbb{E}\left[N^{\text{new}}_{\alpha,i} \,|\, TS_j\right], & \text{otherwise.} \end{cases}$$

By the law of iterated expectation, we have $\mathbb{E}\left[N^{\text{new}}_{\alpha,i} \,|\, TS_j\right] = (\lambda_\alpha/m)\mathbb{E}\left[C_i \,|\, TS_j\right]$, where C_i is the length (duration) of the ith iteration. Moreover, since the number of demands outstanding at the beginning of iteration i is independent of the task that will be chosen, we have $\mathbb{E}\left[N_{\alpha,i} \,|\, TS_j\right] = \mathbb{E}\left[N_{\alpha,i}\right]$. Thus we obtain

$$\mathbb{E}\left[N_{\alpha,i+1} \,|\, TS_j\right] = \begin{cases} \frac{\lambda_\alpha}{m}\mathbb{E}\left[C_i \,|\, TS_j\right], & \text{if } \alpha = j \\ \mathbb{E}\left[N_{\alpha,i}\right] + \frac{\lambda_\alpha}{m}\mathbb{E}\left[C_i \,|\, TS_j\right], & \text{otherwise.} \end{cases}$$

Therefore, we are left with computing the conditional expected values of C_i. The length

of C_i is given by the time needed by the vehicle to travel along the TSP tour plus the time spent to service demands. Assuming i large enough, lemma (4.3.1) holds, and we can apply equation (2.7) to estimate from the quantities $\mathbb{E}[N_{\alpha,i}]$, $\alpha \in \{1,\dots,n\}$, the length of the expected TSP tour at iteration i (see the proof of theorem 7.2.2 and especially equation (7.4) in chapter 7 for a rigorous justification of this statement). Conditioning on TS_j (when only demands of type j are serviced), we have

$$\begin{aligned}\mathbb{E}[C_i \,|\, TS_j] &\leq \frac{\beta_{\text{TSP}}\sqrt{|\mathcal{E}|/m}}{v}\mathbb{E}\left[\sqrt{N_{j,i}} \,|\, TS_j\right] + \mathbb{E}\left[\textstyle\sum_{k=1}^{N_{j,i}} s_{j,k} \,|\, TS_j\right] \\ &\leq \frac{\beta_{\text{TSP}}\sqrt{|\mathcal{E}|/m}}{v}\sqrt{\mathbb{E}[N_{j,i}]} + \mathbb{E}[N_{j,i}]\bar{s}_j,\end{aligned}$$

where we have

- applied equation (2.7);
- applied Jensen's inequality for concave functions, in the form $\mathbb{E}\left[\sqrt{X}\right] \leq \sqrt{\mathbb{E}[X]}$;
- removed the conditioning on TS_j, since the random variables $N_{\alpha,i}$ are independent from *future* events, and in particular from the choice of the task at iteration i; and
- used the crucial fact that the on-site service times are independent from the number of outstanding demands.

Collecting the above results (and using the shorthand $\bar{X}$ to indicate $\mathbb{E}[X]$, where X is any random variable), we have

$$\bar{N}_{\alpha,i+1} \leq (1-p_\alpha)\bar{N}_{\alpha,i} + \sum_{j=1}^{n} p_j \frac{\lambda_\alpha}{m}\left[\frac{\beta_{\text{TSP}}\sqrt{|\mathcal{E}|}}{\sqrt{m}v}\sqrt{\bar{N}_{j,i}} + \bar{N}_{j,i}\bar{s}_j\right], \tag{4.10}$$

for each $\alpha \in \{1,\dots,n\}$. The n inequalities above describe a system of recursive relations that describe an upper bound on $\bar{N}_{\alpha,i}$, $\alpha \in \{1,\dots,n\}$. The following theorem bounds the values to which they converge.

Theorem 4.3.2 (Steady-state queue length)**.** *For every set of initial conditions $\{\bar{N}_{\alpha,0}\}_{\alpha\in\{1,\dots,n\}}$, the trajectories $i \mapsto \bar{N}_{\alpha,i}$, $\alpha \in \{1,\dots,n\}$, resulting from inequalities* (4.10)*, satisfy*

$$\limsup_{i\to+\infty} \bar{N}_{\alpha,i} \leq \frac{\beta_{\text{TSP}}^2|\mathcal{E}|}{m^3v^2(1-\varrho)^2}\frac{\lambda_\alpha}{p_\alpha}\left(\sum_{j=1}^{n}\sqrt{\lambda_j p_j}\right)^2, \quad \text{as } \varrho \to 1^-.$$

Proof. Define $q_j \doteq 1-p_j$ and let $\hat{\lambda}_\alpha$ denote the arrival rate in region $\mathcal{E}_k$. Thus $\hat{\lambda}_\alpha \doteq \lambda_\alpha/m$ for each $\alpha \in \{1,\dots,n\}$. Let $x(i) \doteq (\bar{N}_{1,i}, \bar{N}_{2,i}, \dots, \bar{N}_{n,i}) \in \mathbb{R}^n$ and define two matrices

$$A \doteq \begin{bmatrix} \hat{\lambda}_1 p_1 \bar{s}_1 + q_1 & \hat{\lambda}_1 p_2 \bar{s}_2 & \dots & \hat{\lambda}_1 p_n \bar{s}_n \\ \hat{\lambda}_2 p_1 \bar{s}_1 & \hat{\lambda}_2 p_2 \bar{s}_2 + q_2 & \dots & \hat{\lambda}_2 p_n \bar{s}_n \\ \vdots & & \ddots & \vdots \\ \hat{\lambda}_n p_1 \bar{s}_1 & \hat{\lambda}_n p_2 \bar{s}_2 & \dots & \hat{\lambda}_n p_n \bar{s}_n + q_n \end{bmatrix},$$

and

$$B \doteq \frac{\beta_{\mathrm{TSP}}\sqrt{|\mathcal{E}|}}{\sqrt{m}v}\begin{bmatrix} \hat{\lambda}_1 p_1 & \hat{\lambda}_1 p_2 & \dots & \hat{\lambda}_1 p_n \\ \hat{\lambda}_2 p_1 & \hat{\lambda}_2 p_2 & \dots & \hat{\lambda}_2 p_n \\ \vdots & & \ddots & \vdots \\ \hat{\lambda}_n p_1 & \hat{\lambda}_n p_2 & \dots & \hat{\lambda}_n p_n \end{bmatrix}.$$

Then, letting the relation "$\leq$" in $\mathbb{R}^n$ denote the product order of n copies of $\mathbb{R}$ (in other words, for v, $w \in \mathbb{R}^n$, the relation $v \leq w$ is interpreted component-wise), inequalities (4.10) can be written as

$$x(i+1) \leq Ax(i) + B\begin{bmatrix} \sqrt{x_1(i)} \\ \sqrt{x_2(i)} \\ \vdots \\ \sqrt{x_n(i)} \end{bmatrix} \doteq f(x(i)), \tag{4.11}$$

where $f : \mathbb{R}^n_{\geq 0} \to \mathbb{R}^n_{\geq 0}$, and $x_j(i)$, $j \in \{1, \dots, n\}$, are the components of vector $x(i)$. We refer to the discrete system in inequality (4.11) as System-X. Next we define two auxiliary systems, System-Y and System-Z. The initial conditions of these two systems will be set equal to $x(0)$, and we will use their trajectories to bound the trajectories of original system (i.e., System-X). We define System-Y as

$$y(i+1) = f(y(i)). \tag{4.12}$$

System-Y is, therefore, equal to System-X, with the exception that we replaced the inequality with an equality.

Pick, now, any $\varepsilon > 0$. From Young's inequality

$$\sqrt{a} \leq \frac{1}{4\varepsilon} + \varepsilon a, \quad \text{for all } a \in \mathbb{R}_{\geq 0}. \tag{4.13}$$

Hence, for $i \mapsto y(i) \in \mathbb{R}^n_{\geq 0}$, the equation (4.12) becomes

$$\begin{aligned} y(i+1) &\leq Ay(i) + B\Big(\frac{1}{4\varepsilon}\mathbf{1}_n + \varepsilon\, y(i)\Big) \\ &= \Big(A + \varepsilon B\Big)y(i) + \frac{1}{4\varepsilon}B\mathbf{1}_n, \end{aligned}$$

where $\mathbf{1}_n$ is the vector $(1, 1, \dots, 1)^{\mathrm{T}} \in \mathbb{R}^n$. Next, define System-Z as

$$z(i+1) = \Big(A + \varepsilon B\Big)z(i) + \frac{1}{4\varepsilon}B\mathbf{1}_n \doteq g(z(i)). \tag{4.14}$$

The proof now proceeds as follows. First, we show that the initial conditions $x(0) = y(0) = z(0)$, imply that

$$x(i) \leq y(i) \leq z(i), \quad \text{for all } i \geq 0. \tag{4.15}$$

Second, we show that the trajectories of System-Z are bounded; this fact, together with expression (4.15), implies that also trajectories of System-Y and System-X are bounded. Third, and last, we will compute $\limsup_{i\to+\infty} y(i)$; this quantity, together with expression (4.15), will yield the desired result.

Let us consider the first issue. We have $y(1) = f(y(0))$ and $z(1) = g(z(0))$. By definition of System-Y and System-Z, it holds that $z(0) = y(0)$, and thus $g(z(0)) =$

$g(y(0)) \geq f(y(0))$, where the last inequality follows from inequality (4.13) and by definition of f and g. Therefore, we get $y(1) \leq z(1)$. Then, we have $y(2) = f(y(1))$ and $z(2) = g(z(1))$. Since $z(1), y(1) \in \mathbb{R}^n_{\geq 0}$, and the elements in matrices A and B are all non-negative, then $y(1) \leq z(1)$ implies $g(y(1)) \leq g(z(1))$. Using similar arguments, we can write $z(2) \geq g(y(1)) \geq f(y(1)) = y(2)$; therefore, we get $y(2) \leq z(2)$. Then, it is immediate by induction that $y(i) \leq z(i)$ for all $i \geq 0$.

Similarly, by definition of System-Y, it holds that $x(0) = y(0)$, and thus $x(1) \leq f(x(0)) = f(y(0)) = y(1)$. Then, we get $x(1) \leq y(1)$. Since $x(1), y(1) \in \mathbb{R}^n_{\geq 0}$, the elements in matrices A and B are nonnegative, and by the monotonicity of $\sqrt{\cdot}$, then $x(1) \leq y(1)$ implies $f(x(1)) \leq f(y(1))$. Therefore, we can write $x(2) \leq f(x(1)) \leq f(y(1)) = y(2)$; thus, we get $x(2) \leq y(2)$. Then, it is immediate to show by induction that $x(i) \leq y(i)$ for all $i \geq 0$, and expression (4.15) holds.

We now turn our attention to the second issue, namely boundedness of trajectories for System-Z (in equation (4.14)). Notice that System-Z is a discrete-time linear system. The eigenvalues of A are characterized in the following lemma.

Lemma 4.3.3. *The eigenvalues of A are real and have magnitude strictly less than 1 (i.e., A is a stable matrix).*

Proof. Let $w \in \mathbb{C}^n$ be an eigenvector of A, and $\mu \in \mathbb{C}$ be the corresponding eigenvalue. Then we have $Aw = \mu w$. Define $r \doteq (p_1 \bar{s}_1, p_2 \bar{s}_2, \ldots, p_n \bar{s}_n)$. Then the n eigenvalue equations are

$$\hat{\lambda}_j \, w \cdot r + q_j w_j = \mu \, w_j, \quad j \in \{1, \ldots, n\}, \tag{4.16}$$

where $w \cdot r$ is the scalar product of vectors w and r, and w_j is the jth component of w.

There are two possible cases. The first case is that $w \cdot r = 0$. (Note that since each $p_\alpha > 0$, this case can only occur if $\bar{s}_\alpha = 0$ for some $\alpha \in \{1, \ldots, n\}$.) In this case, equation (4.16) becomes $q_j \, w_j = \mu \, w_j$, for all j. Since $w \neq 0$, there exists j^* such that $w^*_j \neq 0$; thus, we have $\mu = q_{j^*}$. Since $q_{j^*} \in \mathbb{R}$ and $0 < q_{j^*} < 1$, we have that μ is real and $|\mu| < 1$.

Assume, now, that $w \cdot r \neq 0$. This implies that $\mu \neq q_j$ and $w_j \neq 0$ for all j, thus we can write, for all j,

$$w_j = \frac{\hat{\lambda}_j}{\mu - q_j} \, w \cdot r, \tag{4.17}$$

and hence

$$w_j = \frac{\hat{\lambda}_j}{\hat{\lambda}_1} \frac{\mu - q_1}{\mu - q_j} w_1.$$

Therefore, (4.17) can be rewritten as

$$\sum_{j=1}^{n} \frac{r_j \hat{\lambda}_j}{\mu - q_j} = 1. \tag{4.18}$$

Equation (4.18) implies that the eigenvalues are real. To see this, write $\mu = a + ib$, where i is the imaginary unit: then

$$\sum_{j=1}^{n} \frac{r_j \hat{\lambda}_j}{a + ib - q_j} = \sum_{j=1}^{n} \frac{r_j \hat{\lambda}_j [(a - q_j) - ib]}{(a - q_j)^2 + b^2}.$$

Thus equation (4.18) implies

$$\underbrace{b \sum_{j=1}^{n} \frac{r_j \hat{\lambda}_j}{(a-q_j)^2 + b^2}}_{>0} = 0,$$

that is, $b = 0$. Equation (4.18) also implies that the eigenvalues (that are real) have magnitude strictly less than 1. Indeed, assume, by contradiction, that $\mu \geq 1$. Then we have $\mu - q_j \geq 1 - q_j > 0$ (recall that the eigenvalues are real and $0 < q_j < 1$) and we can write

$$\sum_{j=1}^{n} \frac{r_j \hat{\lambda}_j}{\mu - q_j} \leq \sum_{j=1}^{n} \frac{r_j \hat{\lambda}_j}{1 - q_j} = \sum_{j=1}^{n} \bar{s}_j \hat{\lambda}_j = \varrho < 1,$$

which is a contradiction. Assume, again by contradiction, that $\mu \leq -1$. In this case we trivially get another contradiction $\sum_{j=1}^{n} r_j \hat{\lambda}_j / (\mu - q_j) < 0$, since $\mu - q_j < 0$. □

Hence, $A \in \mathbb{R}^{n \times n}$ has eigenvalues strictly inside the unit disk, and since the eigenvalues of a matrix depend continuously on the matrix entries, there exists a sufficiently small $\varepsilon > 0$ such that the matrix $A + \varepsilon B$ has eigenvalues strictly inside the unit disk. Accordingly, each solution $i \mapsto z(i) \in \mathbb{R}^n_{\geq 0}$ of System-Z converges exponentially fast to the unique equilibrium point

$$z^* = \Big(I_n - A - \varepsilon B\Big)^{-1} \frac{1}{4\varepsilon} B \mathbf{1}_n. \tag{4.19}$$

Combining expression (4.15) with the previous statement, we see that the solutions $i \mapsto x(i)$ and $i \mapsto y(i)$ are bounded. Thus

$$\limsup_{i \to +\infty} x(i) \leq \limsup_{i \to +\infty} y(i) < +\infty. \tag{4.20}$$

Finally, we turn our attention to the third issue, namely the computation of $y \doteq \limsup_{i \to +\infty} y(i)$. Taking the lim sup of the left- and right-hand sides of equation (4.12), and noting that

$$\limsup_{i \to +\infty} \sqrt{y_\alpha(i)} = \sqrt{\limsup_{i \to +\infty} y_\alpha(i)} \quad \text{for } \alpha \in \{1, 2, \ldots, n\},$$

since $x \mapsto \sqrt{x}$ is continuous and strictly monotone increasing on $\mathbb{R}_{>0}$, we obtain that

$$y_\alpha = (1 - p_\alpha) y_\alpha + \hat{\lambda}_\alpha \sum_{j=1}^{n} p_j \bigg(\frac{\beta_{\mathrm{TSP}} \sqrt{|\mathcal{E}|}}{\sqrt{m} v} \sqrt{y_j} + \bar{s}_j y_j \bigg).$$

Rearranging we obtain

$$p_\alpha y_\alpha = \hat{\lambda}_\alpha \sum_{j=1}^{n} p_j \bigg(\frac{\beta_{\mathrm{TSP}} \sqrt{|\mathcal{E}|}}{\sqrt{m} v} \sqrt{y_j} + \bar{s}_j y_j \bigg). \tag{4.21}$$

Dividing $p_\alpha y_\alpha$ by $p_1 y_1$ (recall that by Algorithm 3 each p_α is positive) we obtain

$$y_\alpha = \frac{\hat{\lambda}_\alpha p_1}{\hat{\lambda}_1 p_\alpha} y_1. \tag{4.22}$$

Combining equations (4.21) and (4.22), we obtain

$$p_1 y_1 = \varrho\, p_1 y_1 + \frac{\beta_{\mathrm{TSP}} \sqrt{|\mathcal{E}|}}{\sqrt{m} v} \sqrt{p_1 \hat{\lambda}_1 y_1} \sum_{j=1}^{n} \sqrt{\hat{\lambda}_j p_j}$$

Thus, recalling that $\hat{\lambda}_\alpha = \lambda_\alpha / m$, we obtain

$$y_\alpha = \frac{\beta_{\mathrm{TSP}}^2 |\mathcal{E}|}{m^3 v^2 (1-\varrho)^2} \frac{\lambda_\alpha}{p_\alpha} \left(\sum_{j=1}^{n} \sqrt{\lambda_j p_j} \right)^2 .$$

Noting that from inequality (4.20), $\limsup_{i \to +\infty} \bar{N}_{\alpha,i} \leq y_\alpha$, we obtain the desired result. □

4.3.2 System time of the SQ policy in heavy load

From Theorem 4.3.2, and using Little's law, the system time of α-demands under the SQ policy satisfies

$$\begin{aligned} \overline{T}_\alpha &\leq \frac{m}{\lambda_\alpha} \limsup_{i \to +\infty} \bar{N}_{\alpha,i} \\ &\leq \frac{\beta_{\mathrm{TSP}}^2 |\mathcal{E}|}{m^2 v^2 (1-\varrho)^2} \frac{1}{p_\alpha} \left(\sum_{j=1}^{n} \sqrt{\lambda_j p_j} \right)^2 , \end{aligned}$$

where we have neglected $\bar{s}_\alpha$, since as $\varrho \to 1^-$ the constant $\bar{s}_\alpha$ becomes negligible compared to the average system time, which scales as $(1-\varrho)^{-2}$.

Thus, the system time of the SQ policy satisfies

$$\overline{T}_{\mathrm{SQ}} \leq \frac{\beta_{\mathrm{TSP}}^2 |\mathcal{E}|}{m^2 v^2 (1-\varrho)^2} \sum_{\alpha=1}^{n} \frac{c_\alpha}{p_\alpha} \left(\sum_{i=1}^{n} \sqrt{\lambda_i p_i} \right)^2 , \quad \text{as } \varrho \to 1^-. \tag{4.23}$$

With this expression we prove our main result on the performance of the SQ policy.

Theorem 4.3.4 (SQ policy performance)**.** *As $\varrho \to 1^-$, the system time of the SQ policy is within a factor $2n^2$ of the optimal system time. This factor is independent of the arrival rates $\lambda_1, \ldots, \lambda_n$, coefficients $c_1, \ldots, c_n$, service times $\bar{s}_1, \ldots, \bar{s}_n$, and the number of vehicles m.*

Proof. We would like to compare the performance of this policy with the lower bound. To do this, consider setting

$$p_\alpha \doteq c_\alpha \quad \text{for each } \alpha \in \{1, \ldots, n\}.$$

Defining $\Psi \doteq \beta_{\text{TSP}}^2 |\mathcal{E}|/(m^2 v^2 (1-\varrho)^2)$, inequality (4.23) can be written as

$$\overline{T}_{\text{SQ}} \leq \Psi n \left(\sum_{i=1}^{n} \sqrt{c_i \lambda_i} \right)^2 .$$

Next, the lower bound in inequality (4.4) is

$$\overline{T}^* \geq \frac{\Psi}{2} \sum_{i=1}^{n} \left(c_i + 2 \sum_{j=i+1}^{n} c_j \right) \lambda_i \geq \frac{\Psi}{2} \sum_{i=1}^{n} (c_i \lambda_i) .$$

Thus, comparing the upper and lower bounds

$$\frac{\overline{T}_{\text{SQ}}}{\overline{T}^*} \leq 2n \frac{\left(\sum_{i=1}^{n} \sqrt{c_i \lambda_i} \right)^2}{\sum_{i=1}^{n} (c_i \lambda_i)}. \tag{4.24}$$

Letting $x_i \doteq \sqrt{c_i \lambda_i}$, and $\mathbf{x} \doteq [x_1, \ldots, x_n]$, the numerator of the fraction in inequality (4.24) is $\|\mathbf{x}\|_1^2$, and the denominator is $\|\mathbf{x}\|_2^2$. But the one- and two-norms of a vector $\mathbf{x} \in \mathbb{R}^n$ satisfy $\|\mathbf{x}\|_1 \leq \sqrt{n} \|\mathbf{x}\|_2$. Thus,

$$\frac{\overline{T}_{\text{SQ}}}{\overline{T}^*} \leq 2n \left(\frac{\|\mathbf{x}\|_1}{\|\mathbf{x}\|_2} \right)^2 \leq 2n^2, \quad \text{as } \varrho \to 1^-,$$

and the policy is a $2n^2$-factor approximation. □

Remark 4.3.5 (Relation to RP policy in [90])**.** *For $n = 2$ the SQ policy is within a factor of 8 of the optimal. This improves on the factor of 12 obtained for the Randomized Priority (RP) policy in [90]. However, it appears that the RP policy bound is not tight, since for two classes, simulations indicate it performs no worse than the SQ policy.* •

4.3.3 Separate Queues policy with queue merging

In this section we propose a modification the SQ policy based on *queue merging*. Queue merging is guaranteed to never increase the upper bound on the expected system time, and in certain instances it significantly decreases the upper bound. The modification can be used when we have a modest number of classes (fewer than, say, 20), which encompasses most applications of interest.

To motivate the modification, consider the case when all classes have equal priority (i.e., $c_1/\lambda_1 = \cdots = c_n/\lambda_n$), and we use the probability assignment $p_\alpha = c_\alpha$ for each class α. Then, the upper bound for the Separate Queues policy in inequality (4.23) becomes

$$\Psi n(\lambda_1 + \cdots + \lambda_n),$$

where $\Psi \doteq \beta_{\text{TSP}}^2 |\mathcal{E}|/(m^2 v^2 (1-\varrho)^2)$.

On the other hand, if we ignore priorities, merge the n classes into a single class, and run the SQ policy on the merged class (i.e., at each iteration, service all outstanding demands in $\mathcal{E}$ via the TSP tour), then the upper bound becomes

$$\Psi(\lambda_1 + \cdots + \lambda_n).$$

Thus, there is a factor of n separating the two upper bounds. This is due to the fact that the basic SQ policy services each of the n classes separately, even when they have the same priority.

The above discussion motivates the addition of queue merging to the SQ policy. We define a *merge configuration* to be a partition of n classes $\{1, \ldots, n\}$ into ℓ sets $S_1, \ldots, S_\ell$, where $\ell \in \{1, \ldots, n\}$. The upper bound for a merge configuration $\{S_1, \ldots, S_\ell\}$ is

$$\Psi \ell \left(\sum_{i}^{\ell} \sqrt{\sum_{\alpha \in S_i} c_\alpha \sum_{\beta \in S_i} \lambda_\beta} \right)^2 . \tag{4.25}$$

The SQ-policy with merging can be summarized as follows:

Separate Queues with Merging Policy

1 Find the merge configuration $\{S_1, \ldots, S_\ell\}$ which minimizes equation (4.25).
2 Run the Separate Queues policy on ℓ classes, where class i has arrival rate $\sum_{\alpha \in S_i} \lambda_\alpha$ and convex combination coefficient $\sum_{\alpha \in S_i} c_\alpha$.

Now, to minimize equation (4.25) in step 1 of the SQ with Merging policy, one must search over all possible partitions of a set of n elements. The number of partitions is given by the Bell Number B_n which is defined recursively as $B_n = \sum_{k=0}^{n-1} B_k \binom{n-1}{k}$. Thus, the search becomes infeasible for more than 10 classes.

If the search space is too large, then one can limit the search to all partitions such that if $i < j$, then each class in S_i has higher priority than all classes in S_j. This is the set of partitions in which only adjacent classes are merged. For n classes, there are 2^{n-1} such merge configurations, which is significantly less than the Bell number B_n, but is still infeasible for more than, say, 20 classes.

4.3.4 The Tube heuristic for improving performance

We now introduce a simple heuristic improvement for the SQ policy that can be used for implementation. The heuristic improvement is as follows:

> **Tube Heuristic:** When following the tour in step 10 of the SQ policy, service all newly arrived demands that lie within distance $\epsilon > 0$ of the tour.

The idea behind the heuristic is to utilize the fact that some newly arrived demands will be "close" to the demands in the current service batch, and thus can be serviced with minimal additional travel cost. Analysis of the tube heuristic is complicated by the fact that it introduces correlation between demand locations. A similar difficulty arises when attempting to analyze the nearest neighbor policy [14]. However, we can demonstrate the effectiveness of this heuristic through simulations.

The parameter ϵ should be chosen such that the total tour length is not increased by more than, say, 10%. A rough calculation shows that the area of the "tube" of width 2ϵ centered around a tour that passes through the $\text{card}(Q)$ demands in Q has area upper bounded by $2\epsilon\beta_{\text{TSP}}\sqrt{\text{card}(Q)|\mathcal{E}|}$. While following the tour, a vehicle will deviate to service no more than $2\epsilon\beta_{\text{TSP}}\sqrt{\text{card}(Q)/|\mathcal{E}|}(\bar{N}_1 + \cdots + \bar{N}_n)$ demands. Finally, since the vehicle will have to travel no more than 2ϵ to service each demand in the "tube," we see that ϵ should

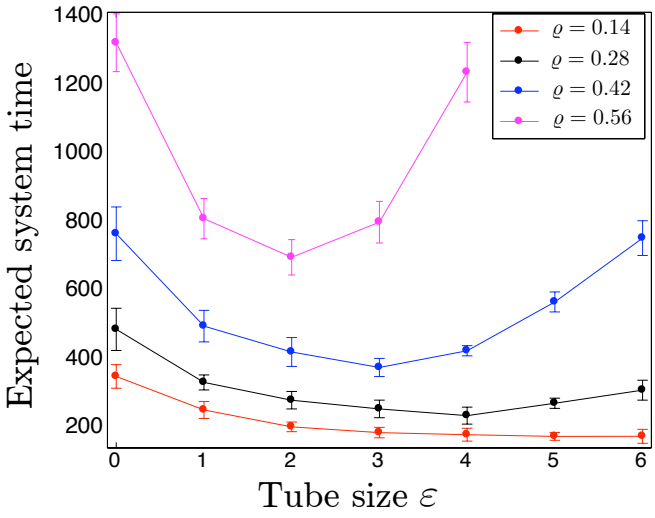


Figure 4-4: The tube heuristic for two classes of demands with $c = 0.8$, $\lambda_2 = 6\lambda_1$, and several different load factors ϱ. The system time at $\epsilon = 0$ corresponds to the basic SQ policy.

scale as

$$\epsilon \sim \sqrt{\frac{\varphi|\mathcal{E}|}{\bar{N}_1 + \cdots + \bar{N}_n}},$$

where $\bar{N}_\alpha$ is expected number of α-demands in the environment, and φ is the fractional increase in tour length (e.g., 10%).

Figure 4-4 shows numerical results for the Tube Heuristic for a single unit speed vehicle in a square environment with side length 50. The simulation is performed for two classes of demands with $c = 0.8$, $\lambda_2 = 6\lambda_1$, and several different load factors ϱ. Each experimental data point represents the average of the steady state system time of ten runs, where each run consists of 200 iterations of the SQ policy. To ensure convergence to steady state and avoid effects due to the transient response, only the last 50 iterations in each run are used to calculate the system time. The basic policy is shown in left-most data points (i.e., $\epsilon = 0$). Figure 4-4 demonstrates that as the load factor increases, the value of ϵ should be chosen smaller in order to achieve the best performance. Table 4.1 shows the improvement in expected system time when using the tube heuristic. For the load factors considered, the heuristic decreases the system time by a factor of approximately 2. One should note that the heuristic is difficult to accurately simulate for high load factors. This is due to the additional computations required to determine if a newly arrived demand lies within a distance ϵ of the current tour. A more sophisticated implementation of tube heuristic is to define an ϵ_α for each $\alpha \in \{1, \ldots, n\}$, where the magnitude of ϵ_α is proportional to its priority, and thus proportional to the probability p_α.

4.4 Simulations and Discussion

In this section we discuss, through the use of simulations, the performance of the SQ policy with the probability assignment $p_\alpha \doteq c_\alpha$, for each $\alpha \in \{1, \ldots, n\}$. In particular, we study

Load factor ϱ	System time	Best ϵ	System time with best ϵ	Heuristic improvement
0.14	358 (34)	5	183 (11)	0.51 (0.16)
0.28	496 (61)	4	244 (25)	0.49 (0.23)
0.42	774 (78)	3	384 (26)	0.50 (0.17)
0.56	1330 (84)	2	706 (52)	0.53 (0.14)
0.70	3380 (357)	1	1770 (121)	0.52 (0.17)

Table 4.1: A comparison between the expected system time of the basic SQ policy, and the SQ policy with the tube heuristic. The values in brackets give the standard deviation of the corresponding table entry.

(i) the tightness of the upper bound in inequality (4.23), (ii) conditions for which the gap between the lower bound in inequality (4.4) and the upper bound in inequality (4.23) is maximized, (iii) the suboptimality of the probability assignment $p_\alpha = c_\alpha$, and (iv) the difference in performance between the SQ policy and a policy that merges all classes together irrespective of priorities. Simulations of the SQ policy were performed using `linkern` (see section 2.4.3) as a solver to generate approximations to the optimal TSP tour.

4.4.1 Tightness of the upper bound

We consider one vehicle, four classes of demands, and several values of the load factor ϱ. For each value of ϱ we perform 100 runs. In each run we uniformly randomly generate arrival rates $\lambda_1, \ldots, \lambda_n$, convex combination coefficients $c_1, \ldots, c_n$, and on-site service times $\bar{s}_1, \ldots, \bar{s}_n$, and normalize the values such that the constraints $\sum_{\alpha=1}^{n} \lambda_\alpha \bar{s}_\alpha = \varrho$ and $\sum_{\alpha=1}^{n} c_\alpha = 1$ are satisfied. In each run we iterate the SQ policy 4000 times, and compute the steady-state expected system time by considering the number of demands in the last 1000 iterations. For each value of ϱ we compute the ratio χ between the expected system time and the theoretical upper bound in inequality (4.23). Table 4.2 reports the ratio, its standard deviation, and its minimum and maximum values for each ϱ value. One can see that the upper bound provides a reasonable approximation for load factors as low as $\varrho = 0.75$.

Load factor (ϱ)	$\mathbb{E}[\chi]$	σ_χ	$\max \chi$	$\min \chi$
0.75	0.803	0.092	1.093	0.354
0.8	0.778	0.108	0.943	0.256
0.85	0.773	0.111	1.150	0.417
0.9	0.733	0.159	1.162	0.203
0.95	0.716	0.131	0.890	0.257

Table 4.2: Ratio χ between experimental results and upper bound for various values of ϱ.

4.4.2 Maximum deviation from lower bound

In Theorem 4.3.4 we showed that the SQ policy performs within a factor of $2n^2$ of the lower bound for all initial conditions. The ratio between the upper bound inequality (4.23) and the lower bound in inequality (4.4) can be made arbitrarily close to $2n^2$ by choosing $\lambda_1 \ll \lambda_2 \ll \cdots \ll \lambda_n$ and $c_1 \gg c_2 \gg \cdots \gg c_n$, with $\lambda_\alpha c_\alpha = a$, for each $\alpha \in \{1, \ldots, n\}$ and for some positive constant a. In these "unfavorable conditions", the upper bound is equal to Bn^3a and the lower bound is approximately $Bna/2$.

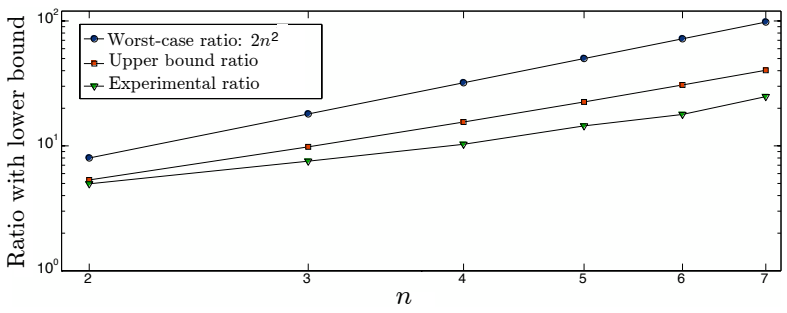


Figure 4-5: Experimental results for the SQ policy in worst-case conditions plotted on a log-log scale; $\varrho = 0.85$ and $\lambda_1 = 1$.

It is also of interest to consider the actual deviation of the experimental performance from the lower bound in the unfavorable conditions described above. We simulated the SQ policy for $\varrho = 0.85$ and for several values of n, with parameter values of $\lambda_n = b\lambda_{n-1} = b^2\lambda_{n-2} = \cdots = b^{n-1}\lambda_1$ and $c_1 = bc_2 = \cdots = b^{n-1}c_n$, where $b = 2$. Figure 4-5 (plotted on a log-log scale) shows that the ratio between the actual performance and the lower bound (averaged over 10 simulation runs) increases as n^η, where $\eta \approx 1.25$ according to a least square fit. The figure also shows that the ratio between the analytic upper bound and the lower bound increases as n^η, where $\eta \approx 1.61$ according to a least square fit. For completeness, the figure also shows the worst-case ratio between the upper bound and lower bound, which increases as n^2. These experimental results suggest that the upper bound is somewhat conservative.

4.4.3 Suboptimality of the approximate probability assignment

To prove theorem 4.3.4 we used the probability assignment

$$p_\alpha \doteq c_\alpha \quad \text{for each } \alpha \in \{1, \ldots, n\}. \tag{4.26}$$

However, one would like to select $[p_1, \ldots, p_n] \doteq \mathbf{p}$ that minimizes the right-hand side of inequality (4.23). The minimization of the right-hand side of inequality (4.23) is a constrained multi-variable nonlinear optimization problem over $\mathbf{p}$, that is, in n dimensions. Thus, for a general n class problem, solving the optimization problem is difficult. However, for two classes of demands the optimization is over a single variable p_1 (with the constraint that $p_2 = 1 - p_1$), and it can be readily solved. A comparison of optimized upper bound, denoted upbd_{opt}, with the upper bound obtained using the probability assignment in equation (4.26), denoted upbd_c, is shown in Figure 4-6. In this figure the ratio of upper bounds is bounded by two.

For $n > 2$ we approximate the solution of the optimization problem as follows. For each value of n we perform 1000 runs. In each run we randomly generate $\lambda_1, \ldots, \lambda_n$, $c_1, \ldots, c_n$, and five sets of initial probability assignments $\mathbf{p}_1, \ldots, \mathbf{p}_5$. From each initial probability assignment we use a line search to locally optimize the probability assignment. We take the ratio between upbd_c and the least upper bound $\text{upbd}_{\text{local opt}}$ obtained from the five locally optimized probability assignments. We also record the maximum variation in the five locally optimized upper bounds. This is summarized in Table 4.3. The second column shows the largest ratio obtained over the 1000 runs. The third column shows the largest

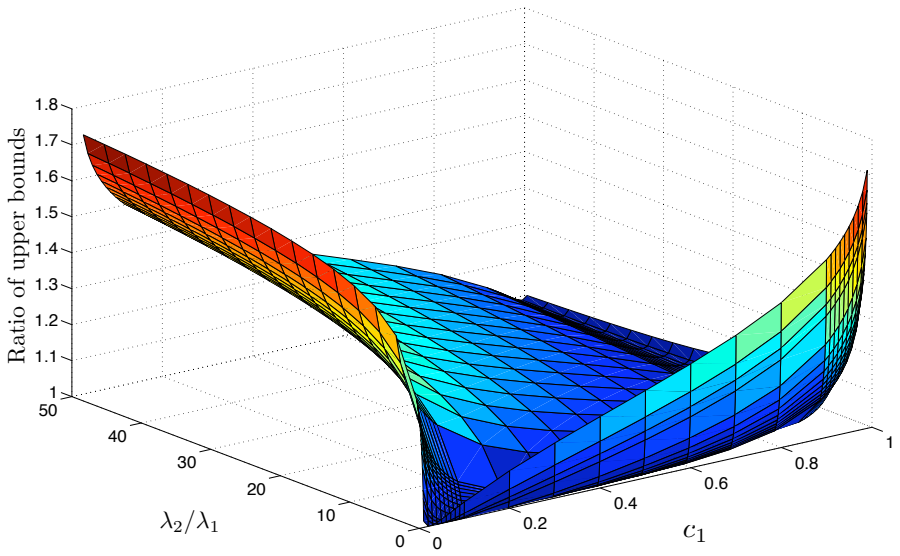


Figure 4-6: The ratios $\text{upbd}_c/\text{upbd}_{opt}$ for 2 classes of demands.

Number of classes (n)	$\text{upbd}_c/\text{upbd}_{\text{local opt}}$	Max. % variation in ratio
3	1.60	0.12
4	1.51	0.04
5	1.51	0.08
6	1.74	0.02
7	1.88	0.08
8	1.63	0.15

Table 4.3: Ratio of upper bound with $p_\alpha = c_\alpha$ for each $\alpha \in \{1, \ldots, n\}$ and the upper bound with a locally optimized probability assignment.

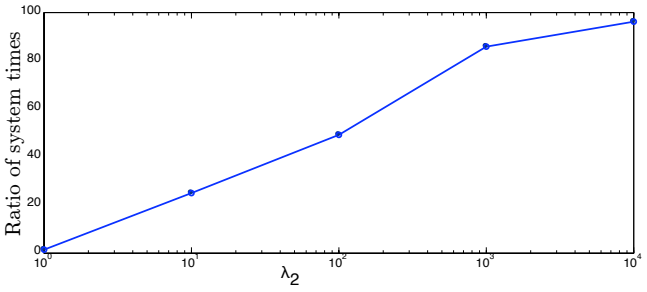


Figure 4-7: Ratio of experimental system times between Complete Merge policy and SQ policy as a function of λ_2, with $n = 2$, $\lambda_1 = 1$, c $= 0.995$ and $\varrho = 0.9$.

% variation in the 1000 runs. The assignment in equation (4.26) seems to perform within a factor of two of the optimized assignment, and the optimization appears to converge to values close to a global optimum since all five random conditions converge to values that are within $\sim 0.1\%$ of each other on every run.

4.4.4 The Complete Merge policy

As described in Section 4.3.3, a naive policy for our problem is to ignore priorities, merge all classes into a single class, and repeatedly form TSP tours through all outstanding demands. We call this policy the Complete Merge (CM) policy. In this section we briefly verify by simulation that the performance of the Complete Merge policy can be very poor when compared to that of the SQ policy. In addition, the poor performance occurs under conditions of interest for most applications—when low priority demands arrive much more frequently than high priority demands.

To upper bound the performance of the Complete Merge policy, define the total arrival rate $\Lambda \doteq \sum_{\alpha=1}^{n} \lambda_\alpha$ and total mean on-site service $\bar{S} \doteq \sum_{\alpha=1}^{n} \bar{s}_\alpha$. Using the upper bounds in [14], we immediately obtain that $\overline{T}_{\mathrm{CM}} \leq \frac{\beta_{\mathrm{TSP}}^2 |\mathcal{E}| \Lambda}{m^2 v^2 (1-\varrho)^2}$. Thus, the ratio of upper bounds can be made arbitrarily large by choosing $\lambda_1 \ll \lambda_2 \ll \cdots \ll \lambda_n$ and $c_1 \gg c_2 \gg \cdots \gg c_n$. This suggests that the ratio between the system time of the CM policy and that of the SQ policy, $\overline{T}_{\mathrm{CM}}/\overline{T}_{\mathrm{SQ}}$, can be made very large. Figure 4-7 shows the experimentally obtained ratio between the system time of the Complete Merge policy and that of the SQ policy (averaged over 10 simulation runs), and verifies that the above choice of arrival rates and convex combination coefficients results in large performance ratios.

4.5 Conclusion

In this chapter we introduced a dynamic vehicle routing problem with priority classes. We believe that it may be possible to improve the lower bound and remove, or reduce, the constant factor's dependence on the number of classes. For future work it would be interesting to combine the aspects of multi-class vehicle routing with problems in which demands require teams of vehicles for their service. Also, it would be interesting to extend the results in this chapter to the case of non-uniform demand densities (possibly class dependent).

Chapter 5

DVR in Transportation Systems

In this chapter we apply the algorithmic queueing theory approach to rigorously study dynamic transportation-on-demand (TOD) systems, where users dynamically formulate requests for transportation from a pick-up point to a delivery point. Typical examples of TOD systems are cab-services and dial-a-ride transportation services for the elderly and the disabled. Furthermore, radically new types of TOD systems are being developed, including mobility-on-demand systems (MOD) [67], which will provide stacks and racks of light electric vehicles at closely spaced intervals throughout a city: when a person wants to go somewhere, he simply walks to the nearest rack, swipes a card to pick up a vehicle, drives it to the rack nearest to his destination, and drops it off. MOD systems will enable convenient point-to-point travel within urban areas and very high vehicle utilization rates, and will extend availability to those who cannot or do not want to own their own vehicles. Large-scale systems employing traditional, non-electric bicycles have already demonstrated the feasibility of mobility-on-demand in several cities throughout Europe, e.g., Paris, Lyon, Milano, Trento, Zurich and so on [66].

The fundamental problem in transportation-on-demand systems is to route the vehicles with the objective that customers' inconvenience (e.g., in terms of waiting time) is minimized. (In the case of MOD systems, we assume the cars can autonomously drive from a delivery location to the next pick-up location -autonomous driving is an active research topic, see for example [24, 23]-.) The static version of this problem, where all requests are known before the routing process begins, is known in the operations research community as the one-to-one Pick-up and Delivery (PD) problem. Several exact and heuristic routing algorithms have been proposed for this problem (see [31] for an authoritative survey). Few rigorous studies exist instead for the dynamic counterpart of the one-to-one PD problem, which often is treated as a sequencing of static subproblems. Dynamic one-to-one PD problems can be divided into three main categories [12]: (i) Dynamic Stacker Crane Problem (where the vehicles have unit capacity), (ii) Dynamic Vehicle Routing Problem with Pick-ups and Deliveries (where the vehicles can transport more than one request), and (iii) Dynamic Dial-a-Ride Problem (where additional constraints such as time windows are considered). Excellent surveys on heuristics, metaheuristics and online algorithms for Dynamic one-to-one PD problems can be found in [12] and [74]. Even though these algorithms are quite effective in addressing Dynamic one-to-one PD problems (or, in other words, dynamic TOD and MOD systems), alone they do not answer critical questions such as: given a certain number of vehicles, what are the fundamental limitations of performance? Is it possible to characterize optimal routing policies? How do customer inconvenience levels

scale down as the number of vehicles increases (in other words, what is the marginal benefit of one more vehicle)? How should one pre-position vehicles when there are no outstanding demands?

To the best of our knowledge, the only analytical studies for Dynamic one-to-one PD problems are [101] and [98]. Specifically, in [101] the authors consider the *uncapacitated* multiple vehicle case of this problem, and provide lower and upper bounds on the achievable performance. In the same vein, in [98] the authors study the *unit*-capacity single vehicle case of this problem, again providing bounds on the achievable performance. The results in [101] and [98] are interesting and insightful, however they are not directly applicable to transportation-on-demand systems, since such systems are characterized by multiple *and* capacitated vehicles.

In this chapter, by employing an algorithmic queueing theory approach, we rigorously study routing problems for dynamic transportation-on-demand systems, where pick-up requests arrive according to a Poisson process and are randomly located according to a general probability density. Corresponding delivery locations are also randomly distributed according to a general probability density, and a *fleet* of *unit-capacity* vehicles must transport demands from their pick-up locations to their delivery locations. The objective is to minimize the expected waiting time for the demands. We assume that the vehicles have single-integrator dynamics and that the environment is a bounded, convex subset within the three-dimensional Euclidean space. These two assumptions are made mainly to ease the exposition: we will in fact argue that the results derived in this chapter for this rather artificial but analytically convenient setting hold also for the more realistic setting where vehicles have differential constraints and operate within a two-dimensional manifold (e.g., planar kinematic vehicles with bounded curvature). The contributions of this chapter are: First, we carefully formulate the problem. Second, we establish lower bounds on the expected waiting time in terms of the number of vehicles and other problem's characteristics (e.g., arrival rate of the demands). Finally, we rigorously study a vehicle routing policy whose performance exhibits the same growth rate (in terms of the traffic congestion) as the lower bound.

5.1 Problem Statement

In this section we present a simple yet insightful model for TOD and MOD systems, which fits within the algorithmic queueing theory framework.

5.1.1 The problem

A total of m vehicles operate in a compact, convex environment $\mathcal{E} \subset \mathbb{R}^3$. The vehicles are free to move, traveling at a maximum velocity v, within the environment $\mathcal{E}$. The vehicles are identical, have unlimited range and are of *unit* capacity (i.e., they can transport one demand at a time). Each vehicle is associated with a depot whose location is $g_k \in \mathcal{E}$, $k \in \{1, \dots, m\}$. Demands are generated according to a homogeneous (i.e., time-invariant) Poisson process, with time intensity $\lambda \in \mathbb{R}_{>0}$. A newly arrived demand has an associated pick-up location which is independent and identically distributed (i.i.d.) in $\mathcal{E}$ according to a density f_{P}. (Note that while a uniform distribution can be a reasonable model for TOD systems, it is not for a MOD system, where pick-ups only happen at specific locations throughout a city.) Each demand must be transported from its pick-up location to its delivery location. The delivery locations are also i.i.d. in $\mathcal{E}$ according to a density f_{D}. In this chapter we

will assume that $f_{\mathrm{P}} = f_{\mathrm{D}} = f$. We will also pose the following technical conditions on f (identical to the ones in [15]):

1. The density f is K-Lipschitz, i.e., $|f(x) - f(y)| \leq K \|x - y\|$, $\forall x, y \in \mathcal{E}$.

2. The density f is bounded below and above, i.e., $0 < \underline{f} \leq f(x) \leq \overline{f} < \infty$, $\forall x \in \mathcal{E}$.

We denote the travel time between the pick-up location of demand j and its delivery location as s_j. A realized demand is removed from the system after one of the vehicles has brought it to its delivery location. Because the sites are generated independently, the expected travel time for demand j is

$$\mathbb{E}[s_j] = \frac{1}{v} \int_{x,y \in \mathcal{E}} \|y - x\| \, f(x) f(y) \, dx \, dy \doteq \bar{s}.$$

We define the *load factor* $\varrho \doteq \lambda \bar{s}/m$.

The system time of demand j, denoted by T_j, is defined as the elapsed time between the arrival of demand j and the time one of the vehicles completes its service (i.e., it delivers the demand to its delivery location). The waiting time of demand j, W_j, is defined by $W_j = T_j - s_j$. The steady-state system time is defined by $\overline{T} \doteq \limsup_{j \to \infty} \mathbb{E}[T_j]$; moreover, we let $\overline{W} \doteq \overline{T} - \bar{s}$. A policy for routing the vehicles is said to be *stable* if the expected number of demands in the system is uniformly bounded at all times. A necessary condition for the existence of a stable policy is that $\varrho < 1$; we shall assume $\varrho < 1$ throughout this chapter. When we refer to *light-load* conditions, we consider the case $\varrho \to 0^+$, in the sense that $\lambda \to 0^+$; when we refer to *heavy-load* conditions, we consider the case $\varrho \to 1^-$, in the sense that $\lambda \to [m/\bar{s}]^-$.

Let $\mathcal{P}$ be the set of all causal, stable, and stationary policies with the additional (technical) property that decisions occur only at service completion epochs, except for vehicles waiting idle at the depot locations. Letting $\overline{T}_\pi$ denote the system time of a particular policy $\pi \in \mathcal{P}$, the problem is to find a policy π^* (if one exists) such that

$$\overline{T}_{\pi^*} = \inf_{\pi \in \mathcal{P}} \overline{T}_\pi.$$

We let $\overline{T}^*$ denote the infimum of the right hand side above.

We call this problem the *Dynamic Pick-up Delivery Problem with m vehicles of unit capacity* (DPDP/m/1).

5.1.2 Discussion

A related problem has been previously studied in [98]. In that paper, the DPDP/m/1 is analyzed under the following assumptions: (i) there is only one vehicle (i.e., $m = 1$), (ii) the distribution of pick-up and delivery locations is uniform (i.e., $f = 1/|\mathcal{E}|$, where $|\mathcal{E}|$ is the area of $\mathcal{E}$), and $\mathcal{E} \subset \mathbb{R}^2$. First, the authors find a policy that is optimal in light load; then, they derive a lower bound on the system time of the order $(1 - \varrho)^{-2}$, and propose a sectoring policy whose bound on the system time is of the order $(1 - \varrho)^{-3}$. Finally, they use simulation to analyze other policies. Note that the lower bound is of the order $(1 - \varrho)^{-2}$, while the growth rate of the sectoring policy is of the order $(1 - \varrho)^{-3}$; therefore, as $\varrho \to 1^-$, the lower bound and the bound for the sectoring policy are arbitrarily far apart.

In this chapter we consider the unit-capacity dynamic Pick-up and Delivery problem in the setting of multiple vehicles with single-integrator dynamics, and arbitrary spatial distribution of demands in three-dimensional environments. Our key contribution is that we are able to find lower bounds and policies that have the *same* growth rate. As mentioned in the introduction, we assume single-integrator dynamics and a three-dimensional environment mostly for analytical convenience: we will argue that the results in this chapter hold also for planar vehicles with differential constraints on their motion.

As in chapters 3 and 4, we will focus the analysis on two limiting regimes, namely $\varrho \to 0^+$ (light load) and $\varrho \to 1^-$ (heavy load). We conclude this section by mentioning three major limitations of the DPDP/m/1: (i) the vehicles can freely travel in $\mathcal{E}$, i.e., there are no "street constraints", (ii) the delivery locations are *independent* of pick-up locations, and (iii) the densities f_{P} and f_{D} are equal.

5.2 Lower Bounds

In this section we present two lower bounds: the first one is most useful as $\varrho \to 0^+$, while the second one holds as $\varrho \to 1^-$.

5.2.1 A light load lower bound

A lower bound that is most useful when $\varrho \to 0^+$ is the following.

Theorem 5.2.1. *The optimal expected time spent in the system by a demand is lower bounded by*

$$\overline{T}^* \geq \frac{1}{v} H_m^*(\mathcal{E}) + \bar{s}. \tag{5.1}$$

Proof. The proof is rather straightforward. Assume that we can place the vehicles in the best *a-priori* positions, i.e., at the locations $G_m^*(\mathcal{E})$, where, as in chapter 2,

$$G_m^*(\mathcal{E}) = \underset{(g_1,\ldots,g_m)\in\mathcal{E}^m}{\arg\min} \mathbb{E}\left[\min_{k\in\{1,\ldots,m\}} \|g_k - x\|\right].$$

The expectation is over demand pick-up sites, i.e., f. By definition, the locations $G_m^*(\mathcal{E})$ minimize the expected distance between the pick-up site of a newly arrived demand and the closest vehicle.

Clearly, the expected travel time for the vehicle assigned to a newly arrived demand to reach the corresponding pick-up site is at least as large as $H_m^*(\mathcal{E})/v = H_m(G_m^*(\mathcal{E}), \mathcal{E})/v$ (see chapter 2 for the definition of H_m). By adding to this the expected time to transfer the demand from its pick-up to its delivery location we obtain the claim. □

5.2.2 A heavy load lower bound

In this section we present a lower bound that holds as $\varrho \to 1^-$; specifically, we will find a heavy-load lower bound for the class of *unbiased* policies within $\mathcal{P}$ (see section 2.5.2 for the definition of unbiased policy). To derive this bound we make heavy usage of the proof techniques developed in [15].

The expected number of outstanding pick-up sites in an arbitrary region $\mathcal{S}$ of the environment can be expressed as

$$\overline{N}_{\mathrm{P}}(\mathcal{S}) = \lambda(\mathcal{S})\overline{W}(\mathcal{S}) = \lambda \int_{\mathcal{S}} f(x)dx \; \overline{W} = \overline{N}_{\mathrm{P}} \int_{\mathcal{S}} f(x)dx, \tag{5.2}$$

where in the first equality we have applied Little's theorem (see [44]) and $\overline{W}(\mathcal{S}) = \overline{W}$ because we are considering unbiased policies.

Because of equation (5.2), and because f is Lipschitz, given a ball

$$\mathcal{B}(x,z) \doteq \left\{x' \in \mathcal{E} \mid \left\|x' - x\right\| \leq z\right\}$$

we have

$$\overline{N}_{\mathrm{P}}(\mathcal{B}(x,z)) = \overline{N}_{\mathrm{P}} f(x)\, V_3\, z^3 + \overline{N}_{\mathrm{P}}\, o(z^3), \tag{5.3}$$

where $V_3 = 4\pi/3$ is the volume of a unit ball in $\mathbb{R}^3$.

In what follows, to ease the exposition, we assume that there is a single depot $g_0 \in \mathcal{E}$. Let Z be the steady-state distance from a vehicle (at the completion epoch of its demand) to the closest outstanding pick-up location, or the depot if closer. We now show a technical lemma, which relates the expected distance $\mathbb{E}[Z]$ to the number of outstanding pick-up locations.

Lemma 5.2.2. *For any unbiased policy in $\mathcal{P}$*

$$\lim_{\overline{N}_{\mathrm{P}} \to \infty} \overline{N}_{\mathrm{P}}^{1/3} \mathbb{E}[Z] \geq \frac{(3/4)^{4/3}}{\sqrt[3]{\pi}} \int_{\mathcal{E}} f^{2/3}(x)\, dx.$$

Proof. We first condition on the event that a randomly tagged demand is delivered at the location $X_{\mathrm{D}} = x$. Let us fix a neighborhood $\mathcal{D}(\overline{N}_{\mathrm{P}}) = \{x' \mid \|x' - g_0\| \leq c^{-1/3}(x)\}$, where $c(x) = \overline{N}_{\mathrm{P}}\, V_3\, f(x)$. There are two possible cases.

Case 1: $x \notin \mathcal{D}(\overline{N}_{\mathrm{P}})$. For z sufficiently small, i.e., such that $\mathcal{B}(x,z)$ does not contain the depot,

$$\mathbb{P}[Z \leq z |\, X_{\mathrm{D}} = x] = \mathbb{P}\left[n_{\mathrm{P}}^{+}(\mathcal{B}(x,z)) > 0\right] \leq \overline{N}_{\mathrm{P}}^{+}(\mathcal{B}(x,z)),$$

where n_{P}^{+} is the number of outstanding pick-up sites in the ball $\mathcal{B}(x,z)$ at the delivery time of the current demand and $\overline{N}_{\mathrm{P}}^{+}$ is its expectation. The inequality above holds because n_{P}^{+} is a non-negative, integer-valued random variable.

For Poisson arrival processes, it holds $\overline{N}_{\mathrm{P}}^{+}(\mathcal{S}) = \overline{N}_{\mathrm{P}}(\mathcal{S})$ (this is a consequence of the PASTA property, see [105]), and recalling equation (5.3) we obtain

$$\overline{N}_{\mathrm{P}}^{+}(\mathcal{B}(x,z)) = \overline{N}_{\mathrm{P}} f(x)\, V_3\, z^3 + \overline{N}_{\mathrm{P}}\, o(z^3).$$

Hence, we can write

$$\begin{aligned}\mathbb{E}\left[Z \,|\, X_{\mathrm{D}} = x\right] &\geq \int_0^{c^{-1/3}(x)} \mathbb{P}\left[Z > z |\, X_{\mathrm{D}} = x\right] dz \\ &\geq \int_0^{c^{-1/3}(x)} 1 - \overline{N}_{\mathrm{P}} f(x)\, V_3\, z^3 - \overline{N}_{\mathrm{P}}\, o(z^3)\, dz \\ &= \int_0^{c^{-1/3}(x)} 1 - c(x)\, z^3 - \overline{N}_{\mathrm{P}}\, o(z^3)\, dz \\ &= \frac{3}{4}\left[\overline{N}_{\mathrm{P}} V_3\, f(x)\right]^{-1/3} - o(\overline{N}_{\mathrm{P}}^{-1/3}).\end{aligned}$$

Case 2: $x \in \mathcal{D}(N_{\mathrm{P}})$. In this case we consider the trivial lower bound $\mathbb{P}\left[Z > z |\, X_{\mathrm{D}} = x\right] \geq 0$.

We now remove the conditioning on the current delivery site, and we obtain (recall that by assumption f is bounded below by $\underline{f}$, and thus $\int_{\mathcal{D}(\overline{N})} dz \leq O(1/\overline{N})$)

$$\begin{aligned}\mathbb{E}\left[Z\right] &= \int_{\mathcal{E}} \mathbb{E}\left[Z \,|\, X_{\mathrm{D}} = x\right] f(x) dx \\ &\geq \left[\overline{N}_{\mathrm{P}} V_3\right]^{-1/3} \frac{3}{4}\left[\int_{\mathcal{E}-\mathcal{D}(\overline{N}_{\mathrm{P}})} f^{-1/3}(x) f(x) dx\right] - o(\overline{N}_{\mathrm{P}}^{-1/3}) \\ &\geq \left[\overline{N}_{\mathrm{P}} V_3\right]^{-1/3} \frac{3}{4}\left[\int_{\mathcal{E}} f^{2/3}(x) dx - \underline{f}^{-1/3}\overline{f} \int_{\mathcal{D}(\overline{N}_{\mathrm{P}})} dx\right] - o(\overline{N}_{\mathrm{P}}^{-1/3}) \\ &= \left[\overline{N}_{\mathrm{P}} V_3\right]^{-1/3} \frac{3}{4}\left[\int_{\mathcal{E}} f^{2/3}(x) dx\right] - o(\overline{N}_{\mathrm{P}}^{-1/3}).\end{aligned}$$

Multiplying by $\overline{N}_{\mathrm{P}}^{1/3}$ and taking the limit as $\overline{N}_{\mathrm{P}} \to \infty$, we obtain the claim. □

We are now in a position to prove the main results of this section.

Theorem 5.2.3 (Heavy-load lower bound)**.** *Within the class of unbiased policies in $\mathcal{P}$*

$$\lim_{\varrho \to 1^-} \overline{T}^*(1-\varrho)^3 \geq \gamma_3^3 \frac{\lambda^2}{m^3 v^3}\left[\int_{\mathcal{E}} f^{2/3}(x)\, dx\right]^3$$

where $\gamma_3 \geq (3/4)^{4/3}/\sqrt[3]{\pi}$.

Proof. Let $\mathbb{E}\left[D\right]$ denote the steady-state expected distance traveled empty between the delivery site of a randomly tagged demand and the pick-up site of the next demand to be serviced by the same vehicle. A necessary condition for stability is that

$$\bar{s} + \frac{\mathbb{E}\left[D\right]}{v} \leq \frac{m}{\lambda}. \tag{5.4}$$

Since, by definition, $\mathbb{E}\left[Z\right] \leq \mathbb{E}\left[D\right]$, equation (5.4) implies

$$\frac{\lambda}{m}\frac{\mathbb{E}\left[Z\right]}{v} \leq 1 - \varrho.$$

By multiplying both sides by $\overline{N}_{\mathrm{P}}^{1/3}$ and raising to the 3rd power we obtain

$$\overline{N}_{\mathrm{P}}(1-\varrho)^3 \geq \frac{\lambda^3[\overline{N}_{\mathrm{P}}^{1/3}\mathbb{E}\left[Z\right]]^3}{m^3v^3}.$$

Applying Little's Law, i.e. $\overline{N}_{\mathrm{P}} = \lambda\overline{W}$, we get

$$\overline{T}(1-\varrho)^3 \geq \overline{W}(1-\varrho)^3 \geq \frac{\lambda^2[\overline{N}^{1/3}\mathbb{E}\left[Z\right]]^3}{m^3v^3}.$$

Taking the limit as $\varrho \to 1^-$ we trivially have that $\overline{N}_{\mathrm{P}} \to \infty$, hence we can apply lemma 5.2.2 and obtain the claim. □

5.2.3 Lower bounds with other vehicle's models

It is significant to mention that the order of the lower bounds derived in this section holds for a number of problems in $\mathbb{R}^2$ where service vehicles have more complex dynamics. Consider, for example, Dubins vehicles, which are planar vehicles constrained to move along paths of bounded curvature, without reversing direction and maintaining a constant speed. A Dubins vehicle at position x (with minimum turning radius ρ) has a reachable set $\mathcal{B}_{\mathrm{Dubins}}(x,z)$ with area $z^3/3\rho$ for small distances z, regardless of heading (see [41]). In the heavy load case, to obtain a result similar to theorem 5.2.3, we simply rewrite equation (5.3) as $\overline{N}_{\mathrm{P}}\left(\mathcal{B}_{\mathrm{Dubins}}(x,z)\right) = \overline{N}_{\mathrm{P}} f(x)\, z^3/3\rho + \overline{N}_{\mathrm{P}}\, o(z^3)$ and re-apply the analysis of lemma 5.2.2.

5.3 Light Load Policies

In this section we briefly describe a policy that achieves asymptotic optimality in the light-load limit. This policy is very similar to the SQM policy discussed in section 2.5.3. For an instance of the problem, we consider the placement of m depots at the m-median of $\mathcal{E}$, i.e., at $G_m^*(\mathcal{E})$. Each depot will correspond to a queue, and is assigned a service vehicle.

> **The m Stochastic Queue Median Policy with Pick-ups (SQMP)** — Upon arrival, a demand is assigned to the depot closest to its pick-up location. The depot's vehicle services its demands in *first-come first-served* order, returning to the depot after each delivery, and waiting there if its queue is empty.

Each of the m resulting queues forms an M/G/1 queue with time intensity $\lambda_k > 0$, such that $\sum_{k=1}^{m}\lambda_k = \lambda$. By applying the Pollaczek-Khinchin formula for the M/G/1 queue [44], we see that the time spent waiting for the vehicle to service other demands goes to zero as $\lambda_k \to 0^+$, and the system time for demands serviced by the kth vehicle tends to $\mathbb{E}\left[\|g_k^* - x\|\right]/v + \bar{s}_k$, where $\bar{s}_k$ is the expected pickup-to-delivery distance conditioned on the depot. When we remove the conditioning with respect to the depot and take $\lambda \to 0^+$, we find that the expected waiting time under this policy approaches exactly

$$\overline{T}_{\mathrm{SQMP}} \to \frac{1}{v}H_m^*(\mathcal{E}) + \bar{s},$$

showing the tightness of the lower bound.

5.4 Heavy Load Policies

Before presenting and analyzing a policy that is particularly effective in heavy load, we define the concept of *bipartite matching tour*.

5.4.1 Bipartite matching tour

Let $X_0, X_1, \ldots, X_n, Y_1, \ldots, Y_n$ be points in $\mathbb{R}^d$. The point X_0 will represent the initial location of a vehicle, the points $\mathcal{X} = \{X_1, \ldots, X_n\}$ will be pick-up locations and the points $\mathcal{Y} = \{Y_1, \ldots, Y_n\}$ will be the corresponding delivery locations. A bipartite matching tour is essentially an approximation of a shortest length tour through the points $X_0, \mathcal{X}, \mathcal{Y}$ with the constraint that when a vehicle visits a pick-up point, the next point to be visited is the corresponding delivery point (such a tour is know in the literature as the *stacker crane tour*).

The bipartite matching tour is constructed as follows. First we add n directed edges $X_i \to Y_i$ that connect pick-up locations to the corresponding delivery locations. Second, we find a bipartite matching for the $\mathcal{X}$ and $\mathcal{Y}$ locations. By adding the n edges of the bipartite matching to the n pick-up to delivery edges $X_i \to Y_i$ we obtain one or more tours, which we call *secondary tours*. Finally, we find a TSP tour (which we call the *primary tour*) across the locations $X_0, X_1, \ldots, X_n$ and we add the corresponding edges. A bipartite matching tour is then as follows: we start at X_0, and follow the primary tour until the first location in $\mathcal{X}$ is reached, say X_j. Then, we follow the secondary tour starting at X_j until we reach again X_j. We resume the primary tour and follow it until we find the next unvisited point in $\mathcal{X}$, say X_k. The procedure is iterated in this way until we reach X_0 again (see Figure 5-1). This concept was originally introduced in [108].

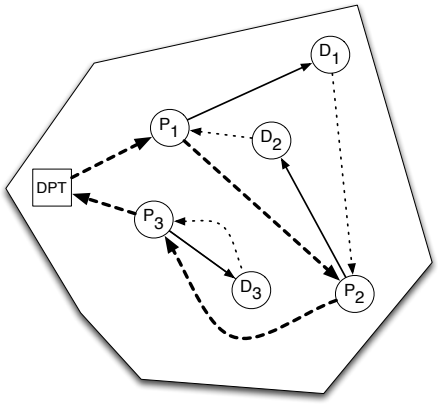


Figure 5-1: A bipartite matching tour. The square represents the current location of the vehicle. P_1, P_2, P_3 are pick-up locations and D_1, D_2, D_3 are the corresponding delivery locations. Solid arrows show links between pick-up and delivery sites. Dotted arrows show links obtained by the bipartite matching between delivery and pick-up sites. Finally, dashed arrows show the primary tour (TSP) through pick-up sites. The bipartite matching tour is: $DPT \to P_1 \to D_1 \to P_2 \to D_2 \to P_1 \to P_2 \to P_3 \to D_3 \to P_3 \to DPT$.

5.4.2 The randomized batch policy

In this section we present an *unbiased* service policy, which we call Randomized Batch policy (RB).

> **The Randomized Batch policy (RB)** — Each newly arrived demand is assigned with probability $p_k = 1/m$ to the vehicle k, $k \in \{1, \dots, m\}$. Then, for each vehicle k: Let Q_k be the set of outstanding demands waiting for service. If $Q_k = \emptyset$, move to g_k (the depot). If, instead, $Q_k \neq \emptyset$, compute a bipartite matching tour through the current vehicle position and all demands in Q_k and service all demands by following such tour. Repeat.

5.4.3 Analysis

The performance of the RB policy in heavy load is characterized by the following theorem.

Theorem 5.4.1 (Performance of RB policy in heavy load)**.** *As $\varrho \to 1^-$, the system time for the RB policy satisfies*

$$\overline{T}_{\mathrm{RB}} \leq \frac{\lambda^2 \left(\beta_{\mathrm{TSP}} + \beta_{\mathrm{M}}\right)^3 \left(\int_{\mathcal{E}} f^{2/3}(x)\, dx\right)^3}{v^3\, m^2\, (1-\varrho)^3} \tag{5.5}$$

The proof of Theorem 5.4.1 requires techniques very similar to those employed in the analysis of the SQ policy in section 4.3; however, we include it for completeness. We start with the following lemma, similar to lemma 4.3.1, characterizing the number of outstanding demands in heavy load.

Lemma 5.4.2 (Number of demands in heavy load)**.** *In heavy load (i.e., $\varrho \to 1^-$), after a transient, the number of demands serviced in a single tour is very large with high probability (i.e., the number of demands tends to $+\infty$ with probability that tends to 1, as ϱ approaches 1^-).*

Proof. The proof is virtually identical to that of lemma 4.3.1 and it is omitted. □

In this policy, each vehicle sees a demand arrival process which is Poisson with rate λ/m and operates within the entire workspace $\mathcal{E}$. Thus, the m-vehicle problem has been turned into m independent and (statistically) identical single-vehicle problems, each with a Poisson arrival process with rate λ/m. As a consequence, we have

$$\mathbb{E}\left[T \,|\, \text{demand assigned to vehicle } j\right] = \mathbb{E}\left[T \,|\, \text{demand assigned to vehicle } k\right],$$

and

$$\begin{aligned} \overline{T}_{\mathrm{RB}} &= \sum_{k=1}^{m} \frac{1}{m}\, \mathbb{E}\left[T \,|\, \text{demand assigned to vehicle } k\right] \\ &= \mathbb{E}\left[T \,|\, \text{demand assigned to vehicle 1}\right]. \end{aligned} \tag{5.6}$$

Therefore it is enough to study the system time for the demands assigned to vehicle 1. For simplicity of notation we omit the label 1 in what follows.

Lemma 5.4.2 has two implications. First, since the number of demands is very large at the time instants when the vehicle starts a new bipartite matching tour, we can apply

equation (2.7) to estimate the average length of the TSP tour and equation (2.9) to estimate the average length of the bipartite matching (see the proof of theorem 7.2.2 and especially equation (7.4) in chapter 7 for a rigorous justification of this statement). Second, since $Q \neq \emptyset$ with high probability, the policy operates with the condition $Q = \emptyset$ *always* false.

We refer to the time instant t_i, $i \geq 0$, in which the vehicle starts a new bipartite matching tour as the epoch i of the policy; we refer to the time interval between epoch i and epoch $i+1$ as the ith iteration. Let n_i be the number of outstanding demands serviced during iteration i. Finally, let C_i be the time interval between the time instant the vehicle starts to service demands during iteration i and the time instant the vehicle starts to service demands during next iteration $i+1$. Demands arrive according to a Poisson process with rate $\hat{\lambda} \doteq \lambda/m$; then, we have $\mathbb{E}[n_{i+1}] = \hat{\lambda}\,\mathbb{E}[C_i]$. The time interval C_i is equal to the time to traverse the bipartite matching tour through the n_i demands, which in turn is the sum of three components:

1. the time to traverse the edges of the TSP tour;
2. the time to traverse the edges of the bipartite matching;
3. the n_i travel times from pick-up locations to delivery locations.

Assume that i is large enough (say, $i \geq \bar{i}$) so that, according to Lemma 5.4.2, the number of outstanding demands is large. Then, the expected time to traverse the bipartite matching tour through the n_i demands can be upper bounded as (with a slight abuse of notation, we call the length of the TSP tour through n_i demands $\mathrm{TSP}(n_i)$, and we denote the length of the bipartite matching through n_i pick-ups and deliveries as $\mathrm{BM}(n_i)$)

$$\begin{aligned}\mathbb{E}[C_i] &= \frac{1}{v}\mathbb{E}[\mathrm{TSP}(n_i+1)] + \frac{1}{v}\mathbb{E}[\mathrm{BM}(n_i)] + \mathbb{E}\left[\textstyle\sum_{j=1}^{n_i} s_j\right] \\ &\leq \mathbb{E}[n_i]^{2/3}\,\frac{\beta_{\mathrm{TSP}} + \beta_{\mathrm{M}}}{v}\int_{\mathcal{E}} f^{2/3}(x)\,dx + \mathbb{E}[n_i]\,\bar{s} + O(1),\end{aligned} \tag{5.7}$$

where we use equations (2.7) and (2.9), and we apply Jensen's inequality for concave functions in the form $\mathbb{E}\left[X^{2/3}\right] \leq \mathbb{E}[X]^{2/3}$.

Then, we obtain the following recurrence relation (where we define $\bar{n}_i \doteq \mathbb{E}[n_i]$):

$$\begin{aligned}\bar{n}_{i+1} &= \hat{\lambda}\mathbb{E}[C_i] \\ &\leq \hat{\lambda}\Big(\bar{n}_i^{2/3}\,\frac{\beta_{\mathrm{TSP}} + \beta_{\mathrm{M}}}{v}\int_{\mathcal{E}} f^{2/3}(x)\,dx + \bar{n}_i\bar{s} + O(1)\Big).\end{aligned} \tag{5.8}$$

The above inequality describes a system of recurrence relations that allows us to find an upper bound on $\limsup_{i\to+\infty} \bar{n}_i$. The following lemma bounds the value to which the limit $\limsup_{i\to+\infty} \bar{n}_i$ converges.

Lemma 5.4.3 (Steady state number of demands)**.** *In heavy load, for every initial condition $\bar{n}_1$, the trajectory $i \mapsto \bar{n}_i$ satisfies*

$$\bar{n} \doteq \limsup_{i\to+\infty} \bar{n}_i \leq \frac{\lambda^3(\beta_{\mathrm{TSP}} + \beta_{\mathrm{M}})^3\Big(\int_{\mathcal{E}} f^{2/3}(x)\,dx\Big)^3}{v^3\,m^3\,(1-\varrho)^3}.$$

Proof. By Lemma 5.4.2, n_i tends to $+\infty$ with probability that tends to 1, as ϱ approaches 1^-; then, after a transient, the term $O(1)$ is negligible compared to the other terms in the

right hand side of equation (5.8), and therefore it can be ignored (its inclusion in the proof is conceptually straightforward, but makes the analysis cumbersome).

Next we define two auxiliary systems, System-Y and System-Z. We define System-Y (with state $y \in \mathbb{R}$) as

$$y(i+1) = \hat{\lambda}\Big(y(i)^{2/3}\frac{\beta_{\text{TSP}}+\beta_{\text{M}}}{v}\int_{\mathcal{E}} f^{2/3}(x)\,dx + y(i)\bar{s}\Big). \tag{5.9}$$

System-Y is obtained by replacing the inequality in equation (5.8) with an equality. Pick, now, any $\varepsilon > 0$. From Young's inequality

$$a = \frac{a\,(p\,\varepsilon)^{\alpha}}{(p\,\varepsilon)^{\alpha}} \leq \Big(a\,(p\,\varepsilon)^{\alpha}\Big)^{p}\frac{1}{p} + \Big(\frac{1}{(p\,\varepsilon)^{\alpha}}\Big)^{q}\frac{1}{q},$$

where $a \in \mathbb{R}_{\geq 0}$, $p, q \in \mathbb{R}_{>0}$, $1/p + 1/q = 1$ and $\alpha, \varepsilon \in \mathbb{R}_{>0}$. By letting $a = y^{2/3}$, $p = 3/2$, $q = 3$ and $\alpha = 2/3$ we obtain:

$$y^{2/3} \leq \varepsilon\, y + \frac{4}{27\,\varepsilon^2}.$$

By applying the above inequality in equation (5.9) we obtain

$$y(i+1) \leq \hat{\lambda}\Big(\bar{s} + \varepsilon\,\frac{\beta_{\text{TSP}}+\beta_{\text{M}}}{v}\int_{\mathcal{E}} f^{2/3}(x)\,dx\Big)\,y(i) + \underbrace{\frac{4\,\hat{\lambda}}{27\,\varepsilon^2}\,\frac{\beta_{\text{TSP}}+\beta_{\text{M}}}{v}\int_{\mathcal{E}} f^{2/3}(x)\,dx}_{=O(1)\,\hat{\lambda}/\varepsilon^2}. \tag{5.10}$$

Next, define System-Z as

$$z(i+1) = \hat{\lambda}\Big(\bar{s} + \varepsilon\,\frac{\beta_{\text{TSP}}+\beta_{\text{M}}}{v}\int_{\mathcal{E}} f^{2/3}(x)\,dx\Big)\,z(i) + O(1)\,\frac{\hat{\lambda}}{\varepsilon^2}. \tag{5.11}$$

It is immediate to show that if $\bar{n}_{\bar{i}} \leq y(\bar{i}) \leq z(\bar{i})$, then

$$\bar{n}_i \leq y(i) \leq z(i), \quad \text{for all } i \geq \bar{i}. \tag{5.12}$$

(Note that System-Y and System-Z are virtual systems for which we can arbitrarily pick the initial conditions.) The proof now proceeds as follows. First, we show that the trajectories of System-Z are bounded; this fact, together with equation (5.12), implies that also trajectories of variables $\bar{n}_i$ and $y(i)$ are bounded. Then, we will compute $\limsup_{i\to+\infty} y(i)$; this quantity, together with equation (5.12), will yield the desired result.

Let us consider the first issue, namely boundedness of trajectories for System-Z (described in equation (5.11)). System-Z is a discrete-time linear system and can be rewritten in compact form as

$$z(i+1) = \Big(\varrho + \varepsilon\, b\Big)z(i) + O(1)\,\frac{\hat{\lambda}}{\varepsilon^2},$$

where $\varrho = \hat{\lambda}\bar{s}$ and $b = \hat{\lambda}(\beta_{\text{TSP}}+\beta_{\text{M}})\int_{\mathcal{E}} f^{2/3}(x)\,dx/v$. Since, by assumption, $\varrho < 1$, there exists a sufficiently small $\varepsilon > 0$ such that $\varrho + \varepsilon\, b < 1$. Accordingly, having selected a sufficiently small ε, each solution $i \mapsto z(i) \in \mathbb{R}_{\geq 0}$ of System-Z converges exponentially fast

to the unique equilibrium point

$$z^*(\varepsilon) = \Big(1 - \varrho - \varepsilon\, b\Big)^{-1} O(1)\, \frac{\hat{\lambda}}{\varepsilon^2}. \tag{5.13}$$

Combining equation (5.12) with the previous statement, we see that the solutions $i \mapsto \bar{n}_i$ and $i \mapsto y(i)$ are bounded. Thus

$$\limsup_{i \to +\infty} \bar{n}_i \leq \limsup_{i \to +\infty} y(i) < +\infty. \tag{5.14}$$

Finally, we turn our attention to the computation of $y \doteq \limsup_{i \to +\infty} y(i)$. Taking the lim sup of the left- and right-hand sides of equation (5.9), and noting that

$$\limsup_{i \to +\infty} y^{2/3}(i) = \Big(\limsup_{i \to +\infty} y(i)\Big)^{2/3},$$

since $x \to x^{2/3}$ is continuous and strictly monotone increasing on $\mathbb{R}_{>0}$, we obtain that

$$y = y^{2/3}\, \hat{\lambda}\, \frac{\beta_{\mathrm{TSP}} + \beta_{\mathrm{M}}}{v} \int_{\mathcal{E}} f^{2/3}(x)\, dx + y\, \varrho; \tag{5.15}$$

rearranging we obtain

$$y = \frac{\hat{\lambda}^3 (\beta_{\mathrm{TSP}} + \beta_{\mathrm{M}})^3 \Big(\int_{\mathcal{E}}\, f^{2/3}(x)\, dx\Big)^3}{v^3\, (1 - \varrho)^3}.$$

Noting that from equation (5.14) $\limsup_{i \to +\infty} \bar{n}_i \leq y$, we obtain the desired result. □

We are now in a position to prove Theorem 5.4.1.

Proof of Theorem 5.4.1. Define $\overline{C} \doteq \limsup_{i \to \infty} \mathbb{E}\,[C_i]$; then we have, by using the upper bound on $\mathbb{E}\,[C_i]$ in equation (5.7) (neglecting $O(1)$ terms),

$$\begin{aligned}
\overline{C} &\doteq \limsup_{i \to \infty} \mathbb{E}\,[C_i] \\
&\leq \Big(\bar{n}^{2/3} \frac{\beta_{\mathrm{TSP}} + \beta_{\mathrm{M}}}{v} \int_{\mathcal{E}}\, f^{2/3}(x)\, dx + \bar{n}\, \bar{s}\Big) \\
&\leq \frac{\lambda^2 (\beta_{\mathrm{TSP}} + \beta_{\mathrm{M}})^3 \Big(\int_{\mathcal{E}}\, f^{2/3}(x)\, dx\Big)^3}{v^3\, m^2\, (1 - \varrho)^2} + \frac{\lambda^2\, \varrho\, (\beta_{\mathrm{TSP}} + \beta_{\mathrm{M}})^3 \Big(\int_{\mathcal{E}}\, f^{2/3}(x)\, dx\Big)^3}{v^3\, m^2\, (1 - \varrho)^3}.
\end{aligned}$$

Hence, in the limit $\varrho \to 1^-$, we have

$$\overline{C} \leq \frac{\lambda^2\, (\beta_{\mathrm{TSP}} + \beta_{\mathrm{M}})^3 \Big(\int_{\mathcal{E}}\, f^{2/3}(x)\, dx\Big)^3}{v^3\, m^2\, (1 - \varrho)^3}.$$

The expected steady-state system time of a random demand, $\overline{T}_{\mathrm{RB}}$, is then upper

bounded, as $\varrho \to 1^-$, by

$$\begin{aligned} \overline{T}_{\mathrm{RB}} &\leq \frac{1}{2}\,\overline{C} + \frac{1}{2}\bar{s}\,\bar{n} \\ &\leq \frac{1}{2}\,\overline{C} + \frac{1}{2}\frac{\lambda^2\,(\beta_{\mathrm{TSP}} + \beta_{\mathrm{M}})^3 \Big(\int_{\mathcal{E}}\, f^{2/3}(x)\,dx\Big)^3}{v^3\,m^2\,(1-\varrho)^3}, \end{aligned} \tag{5.16}$$

where we used the fact that, as $\varrho \to 1^-$, the travel time along the bipartite matching tour is negligible compared to the pick-up to delivery transfer times. Collecting the results we obtain the claim. □

5.4.4 RB policy for Dubins vehicles in $\mathbb{R}^2$

The RB policy, with some modifications, can be adapted to handle the case where vehicles have more complex dynamics. Consider, for example, Dubins vehicles in $\mathbb{R}^2$. We can adapt the concept of bipartite matching tour as follows. Assume there are n outstanding demands. First, we find a Dubins TSP tour (i.e. a TSP tour that respects the differential constraint of a Dubins vehicle) through the pick-up sites and the vehicle's current location. In [87], an algorithm is proposed to construct such a tour, based on a tiling of the plane into $2n$ "beads", geometric shapes adapted to the Dubins dynamics (see [87] for a rigorous description of a bead tiling). The algorithm returns a tour whose length is of order $n^{2/3}$. Note that the Dubins TSP induces a heading constraint for the pick-up sites. Second, we consider again a bead tiling of the plane into $2n$ beads, and we find a minimum length, maximum cardinality bipartite matching from delivery sites to bead entrance points. The assignment induces heading constraints for delivery sites, and by combining theorems 4.1 and 4.8 in [87] the length of this matching is of order $n^{2/3}$. We then find a minimum length, maximum cardinality bipartite matching from bead entrance points to pick-up sites. By combining again theorems 4.1 and 4.8 in [87], the length of this matching is of order $n^{2/3}$. At this stage, each entrance point is associated with one delivery point and with one pick-up point, hence we have found a bipartite matching from delivery points to pick-up points whose length is of order $n^{2/3}$. Finally, we find minimum length paths from the pick-up sites to their corresponding delivery sites with constrained heading. Since the sum of the lengths of the Dubins TSP and of the (Dubins) bipartite matching is of the order $n^{2/3}$, the analysis in the proof of theorem 5.4.1 holds, and a theorem analogous to theorem 5.4.1 can be stated for Dubins vehicles in $\mathbb{R}^2$.

5.4.5 Comparison with the lower bound

With theorem 5.4.1 we can readily prove that the steady-state system time under the RB policy differs from the heavy-load lower bound in theorem 5.2.3 by a known *constant* factor; specifically, the system time under the RB policy has the *same* growth rate of the lower bound (this is not the case in the work [98]).

Theorem 5.4.4. *Let $\overline{T}^*$ be the optimal system time within the class of unbiased policies in $\mathcal{P}$; then*

$$\frac{\overline{T}_{\mathrm{RB}}}{\overline{T}^*} \leq m\,\frac{(\beta_{\mathrm{TSP}} + \beta_{\mathrm{M}})^3}{\gamma_3^3}, \quad \textit{as } \varrho \to 1^-.$$

5.5 Simulation

In this section we present simulation results for the RB policy. We mention that we also studied by simulation several other unbiased policies (e.g., policies for which the demand assignment is not random but follows more advance criteria), however the performance of these policies was similar to that of the RB policy, and so they will not be discussed.

In all simulations we assumed the environment $\mathcal{E}$ to be the three-dimensional unit cube $[0, 1]^3$ and the spatial demand density f to be uniform over $\mathcal{E}$. For each set of parameters (e.g., ϱ, m etc.) we generated 20 instances by simulation, and computed the mean demand system time.

Simulations of the RB policy were performed using `linkern` as a solver to generate approximations to the optimal TSP tour. A Python implementation of the Kuhn-Munkres assignment algorithm [70] was used to generate Euclidean bipartite matchings.

In Figure 5-2(a) we show the dependance of the system time $\overline{T}_{\mathrm{RB}}$ on the load factor ϱ with a number of vehicles $m = 3$. We consider values of $\varrho \in [0.6, 0.75]$, which correspond to a moderate/heavy load. One can observe that the experimental results are within the theoretical lower and upper bounds (even though these bounds formally hold only in the limit $\varrho \to 1^-$); moreover, one can observe that the performance of the RB policy is significantly better than what is predicted by the upper bound; hence we believe that the upper bound in theorem 5.4.1 is rather conservative. We also study how $\overline{T}_{\mathrm{RB}}$ scales with m. To this end, we set the load factor $\varrho = 0.6$ and we simulate the RB policy with $m = 1, 2, 3, 4, 5$. Note that, by fixing ϱ, we are implicitly letting λ be a function of m, since, by definition, $\lambda = \varrho\, m/\bar{s}$. Hence, by increasing m while keeping ϱ fixed the upper bound in theorem 5.4.1 stays constant, while the lower bound in theorem 5.2.3 scales as $1/m$. Figure 5-2(b) reports the value of $\overline{T}_{\mathrm{RB}}$ for each value of m and shows that $\overline{T}_{\mathrm{RB}}$ stays constant. Hence, recalling our previous discussion, we argue that the RB policy indeed scales as $1/m^2$.

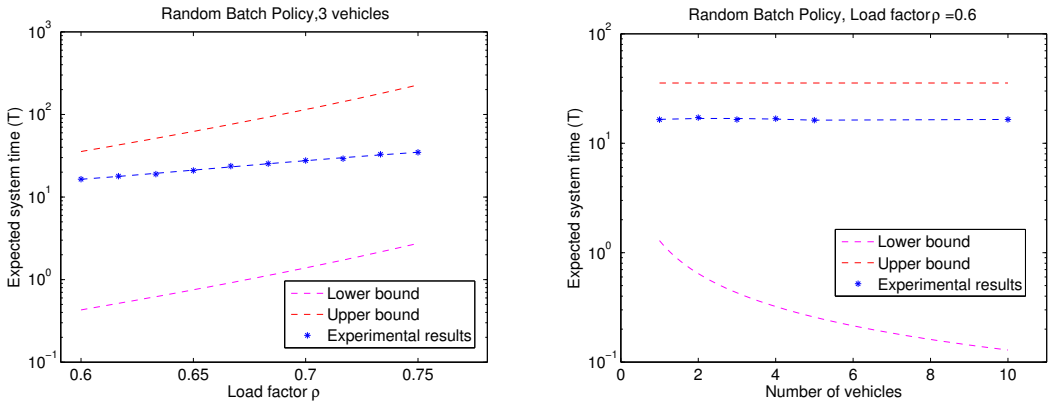


Figure 5-2: Performance of RB policy and comparison with upper and lower bounds. Left figure: $\overline{T}_{\mathrm{RB}}$ versus ϱ. Right figure: scaling of $\overline{T}_{\mathrm{RB}}$ with respect to m.

5.6 Conclusion

In this chapter we studied a dynamic pick-up delivery problem with multiple vehicles of unit capacity and we argued that this is a reasonable model for TOD and MOD systems. This chapter leaves numerous important extensions open for further research. An open issue is

that while the lower bound scales with the number of vehicles as $O(1/m^3)$, the upper bound on the RB policy scales as $O(1/m^2)$. Hence, the optimal scaling of the system time with respect to the number of vehicles is between $O(1/m^3)$ and $O(1/m^2)$, but the exact value is still unknown. It is of strong economic interest to precisely characterize such optimal scaling; this goal would require a tighter lower bound (we conjecture, however, that the scaling $O(1/m^3)$ is indeed correct) and/or devising a policy with a better scaling in terms of m. Furthermore, our initial motivation was to study TOD and MOD systems, for which f_P and f_D might indeed be drastically different (even with different support). Hence it is of interest to extend the analysis in this chapter to the case $f_P \neq f_D$.

Part II

Modes of implementation for DVR algorithms

Chapter 6

Spatially Distributed Algorithms for Environment Partitioning

The best previously known control policies for the m-vehicle Dynamic Traveling Repairman Problem (discussed in section 2.5) rely on centralized task assignment and are not robust against changes in the environment, in particular changes in load conditions; therefore, they are of limited applicability in scenarios involving ad hoc networks of autonomous agents operating in a time-varying environment. In this chapter we present spatially distributed algorithms for environment partitioning, which will be pivotal to design adaptive and distributed routing policies for DVR problems. The application of the algorithms developed in this chapter to DVR problems will be discussed in chapter 7.

Indeed, the distributed partitioning algorithms we present here are of interest in their own right, since they allow a mobile robotic network to share the workload among its members in many scenarios of interest, beyond vehicle routing. Hence, in this chapter we present the problem and the corresponding algorithms in a setting more general than that of DVR.

From an abstract viewpoint, mobile robots can be interpreted as *resources* to be *shared* among *customers*. In surveillance and exploration missions, customers are points of interest to be visited; in transportation and distribution applications, customers are people demanding some service (e.g., utility repair) or goods; in logistics tasks, customers could be troops in the battlefield. Finally, consider a possible architecture for networks of autonomous agents performing distributed sensing: a set of n cheap sensing devices (sensing nodes), distributed in the environment, provides sensor measurements, while m sophisticated agents (cluster heads) collect information from the sensing nodes and transmit it (possibly after some computation) to the outside world. In this case, the sensing nodes represent customers, while the agents, acting as cluster heads, represent resources to be allocated.

The most widely applied strategy for workload sharing among resources is to equalize the total workload assigned to each resource. While, in principle, several strategies are able to guarantee workload-balancing in multi-agent systems, *equitable partitioning policies* are predominant [15, 10, 62, 26]. A partitioning policy is an algorithm that, as a function of the number m of agents and, possibly, of their position and other information, partitions a bounded environment $\mathcal{E} \subset \mathbb{R}^d$ into m openly disjoint regions $\mathcal{E}_i$, for $i \in \{1, \dots, m\}$. In the resource allocation problem, each agent i is assigned to subregion $\mathcal{E}_i$, and each customer in $\mathcal{E}_i$ receives service by the agent assigned to $\mathcal{E}_i$. Accordingly, if we model the *workload* for subregion $\mathcal{S} \subseteq \mathcal{E}$ as $f_{\mathcal{S}} \doteq \int_{\mathcal{S}} f(x)\, dx$, where $f(x)$ is a measure over $\mathcal{E}$, then the workload for

agent i is $f_{\mathcal{E}_i}$. Given this preface, load-balancing calls for equalizing the workload $f_{\mathcal{E}_i}$ in the m subregions or, in equivalent words, to compute an *equitable* partition of the environment $\mathcal{E}$ (i.e., a partition where $f_{\mathcal{E}_i} = f_{\mathcal{E}}/m$, for all i).

Equitable partitioning policies are predominant for three main reasons: (i) efficiency, (ii) ease of design and (iii) ease of analysis. Equitable partitioning policies are, therefore, ubiquitous in multi-agent system applications. To date, nevertheless, to the best of our knowledge, all equitable partitioning policies inherently assume a *centralized* computation of the environment partition. This fact is in sharp contrast with the desire of a fully distributed architecture for a multi-agent system.

The contributions of this chapter are as follows. First, we design provably correct, spatially-distributed, and adaptive policies that allow a team of agents to achieve a convex and equitable partition of a convex environment. Our approach is related to the classic Lloyd algorithm from vector quantization theory [63, 25], and exploits the unique features of power diagrams, a generalization of Voronoi diagrams (see [57] for another interesting application of power diagrams in mobile sensor networks). Then, we provide extensions of our algorithms to take into account *secondary* objectives, as for example, control on the shapes of the subregions. Our motivation, here, is that equitable partitions in which subregions are thin slices are, in most applications, impractical: in the case of vehicle routing, for example, a thin slice partition might be fuel inefficient.

Finally, we mention that our algorithms, although motivated in the context of multi-agent systems, are a novel contribution to the field of computational geometry. In particular we address, using a dynamical system framework, the well-studied equitable convex partition problem (see [27] and references therein); moreover, our results provide new insights in the geometry of Voronoi diagrams and power diagrams.

This chapter is organized as follows. In section 6.1 we provide the necessary tools from calculus, degree theory and geometry. Section 6.2 contains the problem formulation, while in section 6.3 we present preliminary algorithms for equitable partitioning based on leader-election, and we discuss their limitations. Section 6.4 is the core of the chapter: we first prove some existence results for power diagrams, and then we design provably correct, spatially-distributed, and adaptive equitable partitioning policies that do not require any leader election. In section 6.5 we extend the algorithms developed in section 6.4 to take into account *secondary* objectives. Section 6.6 contains simulations results. Finally, in section 6.7, we draw our conclusions.

The work in this chapter was performed in collaboration with Alessandro Arsie, and is based on the journal article [75] and the preliminary conference papers [80, 76].

6.1 Background

In this section, we introduce some notation (that complements the notation introduced in chapter 2) and briefly review some concepts from calculus, degree theory and geometry, on which we will rely extensively later in this chapter.

6.1.1 Notation

Given a vector space $\mathbb{V}$, let $\mathbb{F}(\mathbb{V})$ be the collection of finite subsets of $\mathbb{V}$. Accordingly, $\mathbb{F}(\mathbb{R}^d)$ is the collection of finite point sets in $\mathbb{R}^d$. Let $\mathbb{G}(\mathbb{R}^d)$ be the set of undirected graphs whose vertex set is an element of $\mathbb{F}(\mathbb{R}^d)$ (we assume the reader is familiar with the standard notions of graph theory as defined, for instance, in [37, Chapter 1]).

We define the saturation function $\mathrm{sat}_{a,b}(x)$, with $a < b$, as:

$$\mathrm{sat}_{a,b}(x) = \begin{cases} 1 & \text{if } x > b \\ (x-a)/(b-a) & \text{if } a \leq x \leq b \\ 0 & \text{otherwise} \end{cases} \tag{6.1}$$

6.1.2 Variation of an integral function due to a domain change.

The following result is related to classic divergence theorems [28]. Let $\Omega = \Omega(y) \subset \mathcal{E}$ be a region that depends smoothly on a real parameter $y \in \mathbb{R}$ and that has a well-defined boundary $\partial\Omega(y)$ for all y. Let h be a density function over $\mathcal{E}$. Then

$$\frac{d}{dy}\int_{\Omega(y)} h(x)\,dx = \int_{\partial\Omega(y)} \left(\frac{dx}{dy}\cdot n(x)\right) h(x)\,dx, \tag{6.2}$$

where $v \cdot w$ denotes the scalar product between vectors v and w, where $n(x)$ is the unit outward normal to $\partial\Omega(y)$, and where dx/dy denotes the derivative of the boundary points with respect to y.

6.1.3 A basic result in degree theory

In this section, we state some results in degree theory that will be useful in the remainder of the chapter. For a thorough introduction to the theory of degree we refer the reader to [46].

Let us just recall the simplest definition of degree of a map g. Let $g : X \to Y$ be a smooth map between connected compact manifolds X and Y of the same dimension, and let $p \in Y$ be a regular value for g (regular values abound due to Sard's lemma). Since X is compact, $g^{-1}(p) = \{x_1, \ldots, x_n\}$ is a finite set of points and since p is a regular value, it means that $g_{U_i} : U_i \to g(U_i)$ is a local diffeomorphism, where U_i is a suitable open neighborhood of x_i. Diffeomorphisms can be either orientation preserving or orientation reversing. Let d^+ be the number of points x_i in $g^{-1}(p)$ at which g is orientation preserving (i.e. $\det(\mathrm{Jac}(g)) > 0$, where $\mathrm{Jac}(g)$ is the Jacobian matrix of g) and d^- be the number of points in $g^{-1}(p)$ at which g is orientation reversing (i.e. $\det(\mathrm{Jac}(g)) < 0$). Since X is connected, it can be proved that the number $d^+ - d^-$ is independent of the choice of $p \in Y$ and one defines $\deg(g) \doteq d^+ - d^-$. The degree can be also defined for a *continuous* map $g : X \to Y$ among connected oriented topological manifolds of the same dimensions, this time using homology groups or the local homology groups at each point in $g^{-1}(p)$ whenever the set $g^{-1}(p)$ is finite. For more details see [46].

The following result will be fundamental to prove some existence theorems and it is a direct consequence of the theory of degree of continuous maps among spheres.

Theorem 6.1.1. *Let $g : B^m \to B^m$ be a continuous map from a closed m-ball to itself. Call S^{m-1} the boundary of B^m, namely the $(m-1)$-sphere and assume that $g_{S^{m-1}} : S^{m-1} \to S^{m-1}$ is a map with $\deg(g) \neq 0$. Then g is* onto *B^m.*

Proof. Since g as a map from S^{m-1} to S^{m-1} is different from zero, then the map $g_{S^{m-1}}$ is onto the sphere. If g is not onto B^m, then it is homotopic to a map $B^m \to S^{m-1}$, and then $g_{S^{m-1}} : S^{m-1} \to S^{m-1}$ is homotopic to the trivial map (since it extends to the ball). Therefore $g_{S^{m-1}} : S^{m-1} \to S^{m-1}$ has zero degree, contrary to the assumption that it has degree different from zero. □

In the sequel we will need also the following result.

Lemma 6.1.2. *Let $g : S^m \to S^m$ a continuous bijective map from the m-dimensional sphere to itself ($m \geq 1$). Then* $\deg(g) = \pm 1$.

Proof. The map g is a continuous bijective map from a compact space to a Hausdorff space, and therefore it is a homeomorphism. Now, a homeomorphism $g : S^m \to S^m$ has degree ± 1 (see, for instance, [46, Page 136]). □

6.1.4 Proximity graphs and spatially-distributed control policies for robotic networks

Next, we shall present some relevant concepts on proximity graph functions and spatially-distributed control policies; we refer the reader to [32] for a more detailed discussion. A *proximity graph function* $\mathcal{G} : \mathbb{F}(\mathbb{R}^d) \to \mathbb{G}(\mathbb{R}^d)$ associates to a point set $\mathcal{PS} \in \mathbb{F}(\mathbb{R}^d)$ an undirected graph with vertex set $\mathcal{PS}$ and edge set $\mathcal{ES}_{\mathcal{G}}(\mathcal{PS})$, where $\mathcal{ES}_{\mathcal{G}} : \mathbb{F}(\mathbb{R}^d) \mapsto \mathbb{F}(\mathbb{R}^d \times \mathbb{R}^d)$ has the property that $\mathcal{ES}_{\mathcal{G}}(\mathcal{PS}) \subset \mathcal{PS} \times \mathcal{PS} \setminus \text{diag}(\mathcal{PS} \times \mathcal{PS})$ for any $\mathcal{PS}$. Here, $\text{diag}(\mathcal{PS} \times \mathcal{PS}) = \{(p, p) \in \mathcal{PS} \times \mathcal{PS} |\ p \in \mathcal{PS}\}$. In other words, the edge set of a proximity graph depends on the location of its vertices. To each proximity graph function, one can associate the *set of neighbors map* $N_{\mathcal{G}} : \mathbb{R}^d \times \mathbb{F}(\mathbb{R}^d) \to \mathbb{F}(\mathbb{R}^d)$, defined by

$$N_{\mathcal{G}}(p, \mathcal{PS}) = \{q \in \mathcal{PS} |\ (p, q) \in \mathcal{ES}_{\mathcal{G}}(\mathcal{PS} \cup \{p\})\}.$$

Two examples of proximity graph functions are:

(i) the *Delaunay* graph $G \mapsto \mathcal{G}_{\text{V}}(G) = (G, \mathcal{ES}_{\mathcal{G}_{\text{V}}}(G))$ has edge set

$$\mathcal{ES}_{\mathcal{G}_{\text{V}}}(G) = \{(g_i, g_j) \in G \times G \setminus \text{diag}(G \times G) |\ V_i(G) \cap V_j(G) \neq \emptyset\},$$

where $V_i(G)$ is the ith cell in the Voronoi diagram $\mathcal{V}(G)$;

(ii) the *power-Delaunay* graph $G_W \mapsto \mathcal{G}_{\text{P}}(G_W) = (G_W, \mathcal{ES}_{\mathcal{G}_{\text{P}}}(G_W))$ has edge set

$$\mathcal{ES}_{\mathcal{G}_{\text{P}}}(G_W) = \Big\{\Big((g_i, w_i), (g_j, w_j)\Big) \in G_W \times G_W \setminus \text{diag}(G_W \times G_W) |\ V_i(G_W) \cap V_j(G_W) \neq \emptyset\Big\},$$

where $V_i(G_W)$ is the ith cell in the power diagram $\mathcal{V}(G_W)$.

We are now in a position to discuss spatially-distributed algorithms for robotic networks in formal terms. Let $P(t) = (p_1(t), \ldots, p_m(t)) \in \mathcal{E}^m$ be the ordered set of positions of m agents in a robotic network. We denote the state of each agent $i \in I_m$ at time t as $x_i(t) \in \mathbb{R}^q$ ($x_i(t)$ can include the position of agent i as well as other information). With a slight abuse of notation, let us denote by $I_i(t)$ the information available to agent i at time t. The information vector $I_i(t)$ is a subset of $x(t) \doteq (x_1(t), \ldots, x_m(t))$ of the form $I_i(t) = \{x_{i_1}(t), \ldots, x_{i_k}(t)\}$, $k \leq m$. We assume that $I_i(t)$ always includes $x_i(t)$. Let $\mathcal{G}$ be a proximity graph function defined over $P(t)$ (respectively over $P_W(t)$ if we also consider a weight $w_i(t)$ for each robot $i \in I_m \doteq \{1, \ldots, m\}$); we define $I_i^{N_{\mathcal{G}}}(t)$ as the information vector with the property $x_i(t) \in I_i^{N_{\mathcal{G}}}(t)$, and, for $j \neq i$,

$$\begin{aligned} x_j(t) \in I_i^{N_{\mathcal{G}}}(t) &\Leftrightarrow p_j(t) \in N_{\mathcal{G}}(p_i(t), P(t)) \\ &\Bigg(\Leftrightarrow (p_j(t), w_j(t)) \in N_{\mathcal{G}}\Big((p_i(t), w_i(t)), P_W(t)\Big), \text{ respectively} \Bigg). \end{aligned} \tag{6.3}$$

In words, the information vector $I_i^{N_\mathcal{G}}(t)$ coincides with the states of the neighbors (as induced by $\mathcal{G}$) of agent i together with the state of agent i itself.

Let $\mu(t) = (\mu_1(I_1(t)), \ldots, \mu_m(I_m(t))$ be a feedback control policy for the robotic network. The policy μ is *spatially distributed over* $\mathcal{G}$ if for each agent $i \in I_m$ and for all t

$$\mu_i(I_i(t)) = \mu_i\Big(I_i^{N_\mathcal{G}}(t)\Big).$$

In other words, through information about its neighbors (as induced by $\mathcal{G}$), each agent i has sufficient information to compute the control μ_i.

6.2 Problem Formulation

In this section we state the problem in a rather general setting. We will show in chapter 7 how to apply the distributed partitioning algorithms devised in this chapter to DVR problems.

A total of m identical mobile agents provide service in a compact, convex service region $\mathcal{E} \subseteq \mathbb{R}^d$. Let f be a measure whose bounded support is $\mathcal{E}$ (in other words, f is not zero only on $\mathcal{E}$); for any set $\mathcal{S}$, we define the *workload* for region $\mathcal{S}$ as $f_\mathcal{S} \doteq \int_\mathcal{S} f(x)\,dx$. The measure f models service requests, and might represent, for example, the density of customers over $\mathcal{E}$, or, in a stochastic setting, their arrival rate. Given the measure f, a partition $\{\mathcal{E}_i\}_i$ of the environment $\mathcal{E}$ is *equitable* if $f_{\mathcal{E}_i} = f_{\mathcal{E}_j}$ for all $i, j \in I_m$.

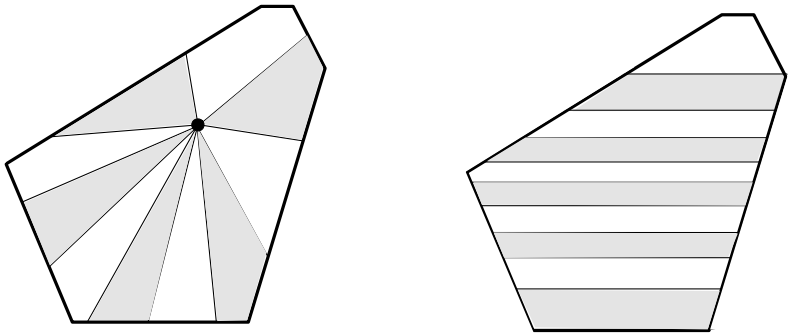
A *partitioning policy* is an algorithm that, as a function of the number m of agents and, possibly, of their position and other information, partitions a bounded environment $\mathcal{E}$ into m openly disjoint subregions $\mathcal{E}_i$, $i \in I_m$. Then, each agent i is assigned to subregion $\mathcal{E}_i$, and each service request in $\mathcal{E}_i$ receives service from the agent assigned to $\mathcal{E}_i$. We refer to subregion $\mathcal{E}_i$ as the *region of dominance* of agent i. Given a measure f and a partitioning policy, m agents are in a *convex equipartition configuration* with respect to f if the associated partition is equitable and convex.

In this chapter we study the following problem: find a *spatially-distributed* (in the sense discussed in Section 6.1) equitable partitioning policy that allows m mobile agents to achieve a convex equipartition configuration (with respect to f). Moreover, we consider the issue of convergence to equitable partitions with some special properties, e.g., where subregions have shapes similar to those of regular polygons.

6.3 Leader-Election Policies

We first describe two simple algorithms that provide equitable partitions. A first possibility is to "sweep" $\mathcal{E}$ from a point in the interior of $\mathcal{E}$ using an arbitrary starting ray until $f_{\mathcal{E}_1} = f_\mathcal{E}/m$, continuing the sweep until $f_{\mathcal{E}_2} = f_\mathcal{E}/m$, etc. A second possibility is to slice, in a similar fashion, $\mathcal{E}$. The resulting equitable partitions are depicted in Figure 6-1.

Then, a possible solution could be to (i) run a distributed leader election algorithm over the graph associated to some proximity graph function $\mathcal{G}$ (e.g., the Delaunay graph); (ii) let each agent send its state $x_i(t)$ to the leader; (iii) let the leader execute either the sweeping or the slicing algorithms described above; finally, (iv) let the leader broadcast subregion's assignments to all other agents. Such conceptually simple solution, however, can be impractical in scenarios where the density f changes over time, or agents can fail, since at every parameter's change a new time-consuming leader election is needed. Moreover,



(a) Sweeping $\mathcal{E}$ (b) Slicing $\mathcal{E}$

Figure 6-1: Equitable partitions by sweeping and slicing (assuming a uniform measure f).

the sweeping and the slicing algorithms provide long and skinny subregions that are not suitable in most applications of interest (e.g., vehicle routing).

We now present spatially-distributed algorithms, based on the concept of power diagrams, that solve *both issues* at once.

6.4 Spatially-Distributed Gradient-Descent Law for Equitable Partitioning

We start this section with an existence theorem for equitable power diagrams.

6.4.1 On the existence of equitable power diagrams

The key advantage of power diagrams is that an equitable power diagram always exists for any f (notice that in general, when f is non-constant, an equitable Voronoi diagram may *fail* to exist, as we will show in section 6.4.5). Indeed, as shown in the next theorem, an equitable power diagram (with respect to any f) exists for *any* vector of *distinct* points $G = (g_1, \dots, g_m)$ in $\mathcal{E}$.

Theorem 6.4.1. *Let $\mathcal{E}$ be a bounded, connected domain in $\mathbb{R}^2$, and f be a measure on $\mathcal{E}$. Let $G = (g_1, \dots, g_m)$ be the positions of $1 \leq m < \infty$ distinct points in $\mathcal{E}$. Then, there exist weights w_i, $i \in I_m$, such that the power points $\Big((g_1, w_1), \dots, (g_m, w_m)\Big)$ generate a power diagram that is equitable with respect to f. Moreover, given a vector of weights W^* that yields an equitable partition, the set of all vectors of weights yielding an equitable partition is $\mathcal{W}_t^* \doteq \{W^* + t[1, \dots, 1]\}$, with $t \in \mathbb{R}$.*

Proof. It is not restrictive to assume that $f_{\mathcal{E}} = 1$ (i.e., we normalize the measure of $\mathcal{E}$), since $\mathcal{E}$ is bounded. The strategy of the proof is to use a topological argument to force existence.

First, we construct a weight space. For simplicity of notation, let D be the diameter of $\mathcal{E}$ (i.e., $D = \operatorname{diam}(\mathcal{E})$), and consider the cube $\mathcal{C} \doteq [-D, D]^m$. This is the weight space and we consider weight vectors W taking value in $\mathcal{C}$. Second, consider the standard *m-simplex* of measures $f_{\mathcal{E}_1}, \dots, f_{\mathcal{E}_m}$ (where $\mathcal{E}_1, \dots, \mathcal{E}_m$ are the power cells). This can be realized in $\mathbb{R}^m$ as the subset of defined by $\sum_{i=1}^m f_{\mathcal{E}_i} = 1$ with the condition $f_{\mathcal{E}_i} \geq 0$. Let us call this set "the measure simplex $\mathcal{A}$" (notice that it is $(m-1)$-dimensional).

There is a map $g : \mathcal{C} \to \mathcal{A}$ associating, according to the power distance, a weight vector W with the corresponding vector of measures $(f_{\mathcal{E}_1}, \ldots, f_{\mathcal{E}_m})$. Since the points in G are assumed to be distinct, this map is continuous.

We will now use induction on m, starting with the base case $m = 3$ (the statement for $m = 1$ and $m = 2$ is trivially checked). We study in detail the case for $m = 3$, where visualization is easier, in order to make the proof more transparent. When $m = 3$, the weight space $\mathcal{C}$ is a three dimensional cube with vertices $v_0 = [-D, -D, -D]$, $v_1 = [D, -D, -D]$, $v_2 = [-D, D, -D]$, $v_3 = [-D, -D, D]$, $v_4 = [D, -D, D]$, $v_5 = [-D, D, D]$, $v_6 = [D, D, -D]$ and $v_7 = [D, D, D]$. The measure simplex $\mathcal{A}$ is, instead, a triangle with vertices u_1, u_2, u_3 that correspond to the cases $f_{\mathcal{E}_1} = 1, f_{\mathcal{E}_2} = 0, f_{\mathcal{E}_3} = 0$, $f_{\mathcal{E}_1} = 0, f_{\mathcal{E}_2} = 1, f_{\mathcal{E}_3} = 0$, and $f_{\mathcal{E}_1} = 0, f_{\mathcal{E}_2} = 0, f_{\mathcal{E}_3} = 1$, respectively. Moreover, call e_1, e_2 and e_3 the edges opposite to the vertices u_1, u_2, u_3 respectively. The edges e_i are, therefore, given by the condition $\{f_{\mathcal{E}_i} = 0\}$ (see Figure 6-2).

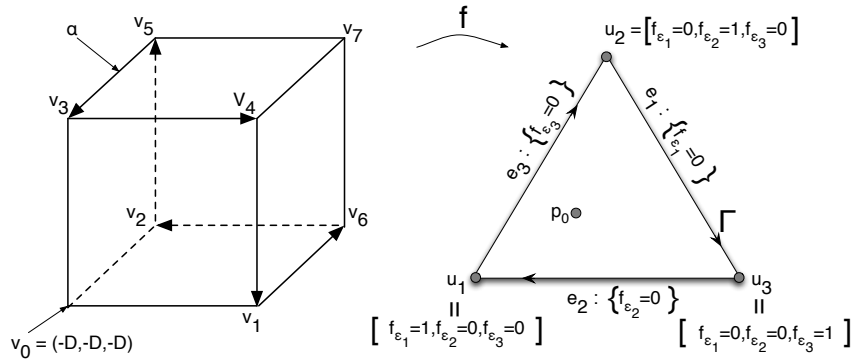


Figure 6-2: Construction used for the proof of existence of equitable power diagrams.

Let us return to the map $g : \mathcal{C} \to \mathcal{A}$. The map g sends v_0 to the unique point p_0 of $\mathcal{A}$ corresponding to the measures of usual Voronoi cells (since the weights are all equal). Observe that only the *differences* among the weights change the vector $(f_{\mathcal{E}_1}, f_{\mathcal{E}_2}, f_{\mathcal{E}_3})$, i.e., if all weights are increased by the same quantity, the vector $(f_{\mathcal{E}_1}, f_{\mathcal{E}_2}, f_{\mathcal{E}_3})$ does not change. In particular, the image of the diagonal v_0v_7 is exactly the point for which the measures are those of a Voronoi partition. Now let us understand what are the "fibers" of g, that is to say, the loci where g is constant. Since the measures of the regions of dominance do not change if the differences among the weights are kept constant, then the fibers of g in the weight space $\mathcal{C}$ are given by the equations $w_1 - w_2 = c_1$ *and* $w_2 - w_3 = c_2$. Rearranging these equations, it is immediate to see that $w_1 = w_3 + c_1 + c_2$, $w_2 = w_3 + c_2$ and $w_3 = w_3$, therefore taking w_3 as parameter we see that the fibers of g are straight lines *parallel* to the main diagonal v_0v_7. Therefore we can conclude that if a particular weight vector W^* yields a specific measure vector $f^* = (f_{\mathcal{E}_1}, f_{\mathcal{E}_2}, f_{\mathcal{E}_3})$, then all the weight vectors of the form $W^* + t[1, \ldots, 1]$, $t \in \mathbb{R}$, will give rise to the same area vector f^*. On the weight space $\mathcal{C}$ let us define the following equivalence relation: $w \equiv w'$ if and only if they are on a line parallel to the main diagonal v_0v_7. Map $g : \mathcal{C} \to \mathcal{A}$ induces a continuous map (still called g by abuse of notation) from $\mathcal{C}/\equiv$ to $\mathcal{A}$ having the same image. Let us identify $\mathcal{C}/\equiv$. Any line in the cube parallel to the main diagonal v_0v_7 is entirely determined by its intersections with the three faces $F_3 = \{w_3 = -D\} \cap \mathcal{C}$, $F_2 = \{w_2 = -D\} \cap \mathcal{C}$ and $F_1 = \{w_1 = -D\} \cap \mathcal{C}$. Call $\mathcal{F}$

the union of these faces. We therefore have a continuous map $g : \mathcal{F} \to \mathcal{A}$ that has the same image of the original g; besides, the induced map $g : \mathcal{F} \to \mathcal{A}$ is *injective* by construction, since each fiber intersects $\mathcal{F}$ in only one point.

Observe that $\mathcal{F}$ is homeomorphic to B^2, the 2-dimensional ball, like $\mathcal{A}$ itself. Up to homeomorphisms, therefore, the map $g : \mathcal{F} \to \mathcal{A}$ can be viewed as a map (again called g by abuse of notation), $g : B^2 \to B^2$. Consider the closed loop α given by v_2v_5, v_5v_3, v_3v_4, v_4v_1, v_1v_6, v_6v_2 with this orientation (see Figure 6-2). This loop is the boundary of $\mathcal{F}$ and we think of it also as the boundary of B^2. Taking into account the continuity of g, it is easy to see that g maps α onto the boundary of $\mathcal{A}$. For example, while we move on the edges v_2v_5 and v_5v_3, that are characterized by having $w_1 = -D$, the corresponding point on the measure simplex moves on the edge e_1.

Moreover, since g is injective by construction, the inverse image of any point in the boundary of $\mathcal{A}$ is just one element of α. Identifying the boundary of $\mathcal{A}$ with S^1 (up to homeomorphisms) and the loop α with S^1 (up to homeomorphisms) we have a bijective continuous map $g_{S^1} : S^1 \to S^1$. By lemma 6.1.2 $\deg(f) = \pm 1$ and therefore g is onto $\mathcal{A}$, using theorem 6.1.1.

We now move to the inductive step, i.e., we suppose that we have proved that the map g is surjective for $m-1$ generators[1] and we show how to use this to show that the map is surjective for m generators.

If we have m generators, the weight space is given by an m dimensional cube $\mathcal{C} = [-D, D]^m$, in complete analogy with the case of 3 generators. The m simplex of the areas $\mathcal{A}$ is again defined as a set $\{(f_{\mathcal{E}_1}, \ldots, f_{\mathcal{E}_m}) \in \mathbb{R}^m\}$ such that $f_{\mathcal{E}_i} \geq 0$ for $i \in \{1, \ldots, m\}$ and $\sum_{i=1}^m f_{\mathcal{E}_i} = 1$. Notice that $\mathcal{A}$ is homeomorphic to the $(m-1)$-dimensional ball B^{m-1}. As before we have a continuous map $g : \mathcal{C} \to \mathcal{A}$. It is easy to see that g is constant on the sets of the form $\mathcal{W}_t \doteq \{\{W + t(1, \ldots, 1)\} \cap \mathcal{C}, \quad t \in \mathbb{R}\}$, that is whenever two sets of weights differ by a common quantity, they are mapped to the same point in $\mathcal{A}$. Moreover, fixing a point $p \in \mathcal{A}$ we have that $g^{-1}(p)$ is given just by a set of the form $\mathcal{W}_t$ for a suitable W. Indeed, assume this is not the case, then the vector of measures $(f_{\mathcal{E}_1}, \ldots, f_{\mathcal{E}_m})$ is obtained via g using two sets of weights: $W^1 \doteq (w_1^1, \ldots w_m^1)$ and $W^2 \doteq (w_1^2, \ldots w_m^2)$, and W^1 and W^2 do not belong to the same $\mathcal{W}_t$, namely it is not possible to obtain W^2 as $W^1 + t(1, \ldots, 1)$ for a suitable t. This means that the vector difference $W^2 - W^1$ is not a multiple of $(1, \ldots, 1)$. Therefore, there exists a nonempty set of indexes J, such that $w_j^2 - w_j^1 \geq w_k^2 - w_k^1$, whenever $j \in J$ and for all $k \in \{1, \ldots m\}$ and such that the previous inequality is strict for at least one k^*. Now among the indexes in J, there exists at least one of them, call it j^*, such that the generator j^* is a neighbor of generator k^*, due to the fact that the domain $\mathcal{E}$ is connected. But since $w_{j^*}^2 - w_{j^*}^1 > w_{k^*}^2 - w_{k^*}^1$, and $w_{j^*}^2 - w_{j^*}^1 \geq w_k^2 - w_k^1$ for all $k \in \{1, \ldots, m\}$, this implies that the measure $f_{\mathcal{E}_{j^*}}$ corresponding to the choice of weights W^2 is strictly greater that $f_{\mathcal{E}_{j^*}}$ corresponding to the choice of weights W^1. This proves that $g^{-1}(p)$ is given only by sets of the form $\mathcal{W}_t$.

We introduce an equivalence relation on $\mathcal{C}$, declaring that two sets of weights W^1 and W^2 are equivalent if and only if they belong to the same $\mathcal{W}_t$. Let us call $\equiv$ this equivalence relation. It is immediate to see that g descends to a map, still called g by abuse of notation, $g : \mathcal{C}/\equiv \to \mathcal{A}$ and that g is now injective. It is easy also to identify $\mathcal{C}/\equiv$ with the union of the $(m-1)$-dimensional faces of $\mathcal{C}$ given by $\mathcal{F} = \cup_{i=1}^m (\mathcal{C} \cap \{w_i = -D\})$. In this way we get a continuous injective map $g : \mathcal{F} \to \mathcal{A}$ that has the same image as the original g. Notice also that $\mathcal{F}$ is homeomorphic to the closed $(m-1)$-dimensional ball, so up to homeomorphism

[1]For short, henceforth we will refer to a power generator simply as a generator.

g can be viewed as a map $g : B^{m-1} \to B^{m-1}$.

We want to prove that the map $g_{\partial\mathcal{F}}$, given by the restriction of g to $\partial\mathcal{F}$ is onto $\partial\mathcal{A}$. To see this, consider one of the $(m-2)$-dimensional faces $\partial\mathcal{A}_i$ of $\partial\mathcal{A}$, which is identified by the condition $f_{\mathcal{E}_i} = 0$. Consider the face F_i in $\mathcal{F}$, where F_i is given by $F_i \doteq \mathcal{C} \cap \{w_i = -D\}$. We claim that the $S_i \doteq \partial F_i \cap \partial\mathcal{F}$ is mapped onto $\partial\mathcal{A}_i$ by g. Observe that the S_i is described by the following equations $S_i = \cup_{j\neq i}(\{w_i = -D, w_j = D\} \cap \mathcal{F})$, so S_i is exactly equivalent to a set of type $\mathcal{F}$ for $m-1$ generators. Moreover observe that $\partial\mathcal{A}_i$ can also be identified with the measure simplex for $m-1$ generators. By inductive hypothesis therefore, the map $g : S_i \to \partial\mathcal{A}_i$ is surjective, and therefore also the map $g_{\partial\mathcal{F}}$ is onto $\partial\mathcal{A}$. Since $g_{\partial\mathcal{F}}$ is a bijective continuos map among $(m-2)$-dimensional spheres, (up to homeomorphism), it has degree ± 1 by lemma 6.1.2. Finally we conclude that g is onto $\mathcal{A}$, using again theorem 6.1.1.

□

Some remarks are in order.

Remark 6.4.2. *The above theorem holds for any bounded, connected domain in $\mathbb{R}^d$. Thus, the case of a compact, convex subset of $\mathbb{R}^d$ is included as a special case. Moreover, it holds for any measure f absolutely continuous with respect to the Lebesgue measure, and for* any *vector of distinct points in $\mathcal{E}$.*

Remark 6.4.3. *In the proof of the above theorem, we actually proved that for* any *measure vector $\{f_{\mathcal{E}_i}\}_{i=1,\dots m}$ in $\mathcal{A}$, there exists a weight vector $W \in \mathcal{C}$ realizing it through the map g. This could be useful in some applications.*

Remark 6.4.4. *Since all vectors of weights in $\mathcal{W}_t$ yield exactly the* same *power diagram, we conclude that the positions of the generators* uniquely *induce an equitable power diagram.*

6.4.2 State, region of dominance, and locational optimization

The first step is to define the state for each agent i. We let $x_i(t)$ be the power generator $(g_i(t), w_i(t)) \in \mathcal{E} \times \mathbb{R}$, which is an artificial variable locally controlled by agent i. Henceforth, we assume that $\mathcal{E}$ is a compact, convex subset of $\mathbb{R}^2$. We, then, define the region of dominance for agent i as the power cell $V_i = V_i(G_W)$, where $G_W = \Big((g_1, w_1), \dots, (g_m, w_m)\Big)$. We refer to the partition into regions of dominance induced by the set of generators G_W as $\mathcal{V}(G_W)$.

In light of theorem 6.4.1, the key idea is to enable the weights of the generators to follow a spatially-distributed gradient descent law (while maintaining the positions of the generators *fixed*) such that an equitable partition is reached. Assume, henceforth, that the positions of the generators are *distinct*, i.e., $g_i \neq g_j$ for $i \neq j$. Define the set

$$S \doteq \Big\{(w_1, \dots, w_m) \in \mathbb{R}^m \mid f_{V_i} > 0 \ \ \forall i \in I_m\Big\}. \tag{6.4}$$

Set S contains all possible vectors of weights such that no region of dominance has measure equal to *zero*.

We introduce the locational optimization function $H_V : S \mapsto \mathbb{R}_{>0}$:

$$H_V(W) \doteq \sum_{i=1}^{m} \Big(\int_{V_i(W)} f(x)dx\Big)^{-1} = \sum_{i=1}^{m} f^{-1}_{V_i(W)}, \tag{6.5}$$

where $W \doteq (w_1, \ldots, w_m)$.

Lemma 6.4.5. *A vector of weights that yields an equitable power diagram is a global minimum of H_V.*

Proof. Consider the relaxation of our minimization problem:

$$\min_{x_1,\ldots,x_m} \sum_{i=1}^{m} x_i^{-1}; \quad \text{s.t. } \sum_{i=1}^{m} x_i = a > 0, \quad x_i > 0,\ i \in I_m,$$

where the linear equality constraint follows from the fact that

$$\sum_{i=1}^{m} \int_{V_i(W)} f(x)dx = \int_{\mathcal{E}} f(x)\, dx \doteq a,$$

and where the constant a is greater than zero since f is a measure whose bounded support is $\mathcal{E}$. By Lagrange multiplier arguments, it is immediate to show that the global minimum for the relaxation is $x_i = a/m$ for all i. Since theorem 6.4.1 establishes that there exists a vector of weights in S that yields an equitable power diagram, we conclude that this vector is a global minimum of H_V. □

6.4.3 Smoothness and gradient of H_V

We now analyze the smoothness properties of H_V. In the following, let $\gamma_{ij} \doteq \|g_j - g_i\|$.

Theorem 6.4.6. *Assume that the positions of the generators are* distinct, *i.e., $g_i \neq g_j$ for $i \neq j$. Given a measure f, the function H_V is continuously differentiable on S, where for each $i \in \{1, \ldots, m\}$*

$$\frac{\partial H_V}{\partial\, w_i} = \sum_{j \in N_i} \frac{1}{2\gamma_{ij}} \Big(\frac{1}{f_{V_j}^2} - \frac{1}{f_{V_i}^2} \Big) \int_{\Delta_{ij}} f(x)\, dx, \tag{6.6}$$

where N_i denotes the set of indices of the power neighbors of (g_i, w_i) (see section 2.3.3). Furthermore, the critical points of H_V are vectors of weights that yield an equitable power diagram.

Proof. By assumption, $g_i \neq g_j$ for $i \neq j$, thus the power diagram is well defined. Since the motion of a weight w_i affects only power cell V_i and its neighboring cells V_j for $j \in N_i$, we have

$$\frac{\partial H_V}{\partial w_i} = -\frac{1}{f_{V_i}^2} \frac{\partial f_{V_i}}{\partial w_i} - \sum_{j \in N_i} \frac{1}{f_{V_j}^2} \frac{\partial f_{V_j}}{\partial w_i}. \tag{6.7}$$

Now, the result in equation (6.2) provides the means to analyze the variation of an integral function due to a domain change. Since the boundary of V_i satisfies $\partial V_i = \cup_j \Delta_{ij} \cup B_i$, where $\Delta_{ij} = \Delta_{ji}$ is the edge between V_i and V_j, and B_i is the boundary between V_i and $\mathcal{E}$ (if any, otherwise $B_i = \emptyset$), we have

$$\frac{\partial f_{V_i}}{\partial w_i} = \sum_{j \in N_i} \int_{\Delta_{ij}} \Big(\frac{\partial x}{\partial w_i} \cdot n_{ij}(x) \Big)\, f(x)\, dx + \underbrace{\int_{B_i} \Big(\frac{\partial x}{\partial w_i} \cdot n_{ij}(x) \Big)\, f(x)\, dx}_{=0}, \tag{6.8}$$

where we defined n_{ij} as the unit normal to Δ_{ij} outward of V_i (therefore we have $n_{ji} = -n_{ij}$). The second term is trivially equal to zero if $B_i = \emptyset$; it is also equal to zero if $B_i \neq \emptyset$, since the integrand is zero almost everywhere. Similarly,

$$\frac{\partial f_{V_j}}{\partial w_i} = \int_{\Delta_{ij}} \Big(\frac{\partial x}{\partial w_i} \cdot n_{ji}(x) \Big) \, f(x) \, dx. \tag{6.9}$$

To evaluate the scalar product between the boundary points and the unit outward normal to the border in equation (6.8) and in equation (6.9), we differentiate equation (2.5) with respect to w_i at every point $x \in \Delta_{ij}$; we get

$$\frac{\partial x}{\partial w_i} \cdot (g_j - g_i) = \frac{1}{2}. \tag{6.10}$$

From equation (2.5) we have $n_{ij} = (g_j - g_i) / \|g_j - g_i\|$, and the desired explicit expressions for the scalar products in equation (6.8) and in equation (6.9) follow immediately (recalling that $n_{ji} = -n_{ij}$).

Collecting the above results, we get the partial derivative with respect to w_i. The proof of the characterization of the critical points is an immediate consequence of the expression for the gradient of H_V; we omit it in the interest of brevity. □

Remark 6.4.7. *The computation of the gradient in theorem 6.4.6 is spatially-distributed over the power-Delaunay graph, since it depends only on information from the agents with neighboring power cells.*

Example 6.4.8 (Gradient of H_V for uniform measure)**.** *The gradient of H_V simplifies considerably when f is constant. In such case, it is straightforward to verify that (assuming that f is normalized)*

$$\frac{\partial H_V}{\partial \, w_i} = \frac{1}{2|\mathcal{E}|} \sum_{j \in N_i} \frac{\delta_{ij}}{\gamma_{ij}} \Big(\frac{1}{|V_j|^2} - \frac{1}{|V_i|^2} \Big), \tag{6.11}$$

where δ_{ij} is the length of the border Δ_{ij}.

6.4.4 Spatially-distributed algorithm for equitable partitioning

Consider the set $U \doteq \Big\{ (w_1, \ldots, w_m) \in \mathbb{R}^m \mid \sum_{i=1}^m w_i = 0 \Big\}$. Since adding an identical value to every weight leaves all power cells unchanged, there is *no loss of generality* in restricting the weights to U; let $\Omega \doteq S \cap U$. Assume that the generators' weights obey a first order dynamical behavior described by

$$\dot{w}_i = u_i.$$

Consider H_V an objective function to be minimized and impose that the weight w_i follows the gradient descent given by (6.6). In more precise terms, we set up the following control law defined over the set Ω

$$u_i = -\frac{\partial H_V}{\partial w_i}(W), \tag{6.12}$$

where we assume that the partition $\mathcal{V}(W) = \{V_1(W), \ldots, V_m(W)\}$ is continuously updated. One can prove the following result.

Theorem 6.4.9. *Assume that the positions of the generators are* distinct*, i.e.* $g_i \neq g_j$ *for* $i \neq j$*. Consider the gradient vector field on* Ω *defined by equation* (6.12)*. Then generators' weights starting at* $t = 0$ *at* $W(0) \in \Omega$*, and evolving under* (6.12) *remain in* Ω *and converge asymptotically to a critical point of* H_V*, i.e., to a vector of weights yielding an equitable power diagram.*

Proof. We first prove that generators' weights evolving under (6.12) remain in Ω and converge asymptotically to the *set* of critical points of H_V. By assumption, $g_i \neq g_j$ for $i \neq j$, thus the power diagram is well defined. First, we prove that set Ω is positively invariant with respect to (6.12). Recall that $\Omega = S \cap U$. Noticing that control law (6.12) is a gradient descent law, we have

$$f_{V_i(W(t))}^{-1} \leq H_V(W(t)) \leq H_V(W(0)), \quad i \in I_m,\ t \geq 0.$$

Since the measures of the power cells depend continuously on the weights, we conclude that the measures of all power cells will be bounded away from zero; thus, the weights will belong to S for all $t \geq 0$, that is, $W(t) \in S\ \forall t \geq 0$. Moreover, the sum of the weights is invariant under control law (6.12). Indeed,

$$\frac{\partial \sum_{i=1}^m w_i}{\partial t} = -\sum_{i=1}^m \frac{\partial H_V}{\partial w_i} = -\sum_{i=1}^m \sum_{j \in N_i} \frac{1}{2\gamma_{ij}} \Big(\frac{1}{f_{V_j}^2} - \frac{1}{f_{V_i}^2} \Big) \int_{\Delta_{ij}} f(x)\, dx = 0,$$

since $\gamma_{ij} = \gamma_{ji}$, $\Delta_{ij} = \Delta_{ji}$, and $j \in N_i \Leftrightarrow i \in N_j$. Thus, we have $W(t) \in U\ \forall t \geq 0$. Since $W(t) \in S\ \forall t \geq 0$ and $W(t) \in U\ \forall t \geq 0$, we conclude that $W(t) \in S \cap U = \Omega$, $\forall t \geq 0$, that is, set Ω is positively invariant.

Second, $H_V : \Omega \to \mathbb{R}_{\geq 0}$ is clearly non-increasing along system trajectories, that is, $\dot{H}_V \leq 0$ in Ω.

Third, all trajectories with initial conditions in Ω are bounded. Indeed, we have already shown that $\sum_{i=1}^m w_i = 0$ along system trajectories. This implies that weights remain within a bounded set: If, by contradiction, a weight could become arbitrarily positive large, another weight would become arbitrarily negative large (since the sum of weights is constant), and the measure of at least one power cell would vanish, which contradicts the fact that S is positively invariant.

Finally, by theorem 6.4.6, H_V is continuously differentiable in Ω. Hence, by invoking the LaSalle invariance principle (see, for instance, [25]), under the descent flow (6.12), weights will converge asymptotically to the *set* of critical points of H_V, that is not empty as confirmed by theorem 6.4.1.

Indeed, by theorem 6.4.1, we know that all vectors of weights yielding an equitable power diagram differ by a common translation. Thus, the largest invariant set of H_V in Ω contains only one point. This implies that $\lim_{t\to\infty} W(t)$ exists and it is equal to a vector of weights that yields an equitable power diagram.

□

Some remarks are in order.

Remark 6.4.10. *By theorem 6.4.9, for* any *set of generators' distinct positions, convergence to an equitable power diagram is* global *with respect to* Ω*. Indeed, there is a very natural choice for the initial values of the weights. Assuming that at* $t = 0$ *agents are in* $\mathcal{E}$

and in distinct positions, each agent initializes its weight to zero. Then, the initial partition is a Voronoi tessellation; since f is positive on $\mathcal{E}$, each initial cell has nonzero measure, and therefore $W(0) \in \Omega$ (the sum of the initial weights is clearly zero).

Remark 6.4.11. *The partial derivative of H_V with respect to the ith weight only requires information from the agents with neighboring power cells. Therefore, the gradient descent law* (6.12) *is indeed a* spatially-distributed *control law over the power-Delaunay graph. We mention that, in a power diagram, each power generator has an average number of neighbors less than or equal to six [8]; therefore, the computation of gradient* (6.12) *is scalable with the number of agents.*

Remark 6.4.12. *The focus here is on equitable partitions. Notice, however, that it is easy to extend the previous algorithm to obtain a spatially-distributed (again over the power-Delaunay graph) control law that provides* any *desired measure vector $\{f_{\mathcal{E}_i}\}_i$. In particular, assume that we desire a partition such that*

$$f_{\mathcal{E}_i} = \beta_i f_{\mathcal{E}},$$

where $\beta_i \in (0,\, 1)$, $\sum_{j=1}^{m} \beta_j = 1$. If we redefine $H_V : S \mapsto \mathbb{R}_{>0}$ as

$$H_V(W) \doteq \sum_{i=1}^{m} \frac{\beta_i^2}{f_{V_i(W)}},$$

then, following the same steps as before, it is possible to show that, under control law

$$\dot{w}_i = -\frac{\partial H_V}{\partial w_i}(W) = \sum_{j \in N_i} \frac{1}{2\gamma_{ij}} \left(\frac{\beta_j^2}{f_{V_j}^2} - \frac{\beta_i^2}{f_{V_i}^2} \right) \int_{\Delta_{ij}} f(x)\, dx,$$

the solution converges to a vector of weights that yields a power diagram with the property $f_{\mathcal{E}_i} = \beta_i f_{\mathcal{E}}$ (whose existence is guaranteed by Remark 6.4.3).

Remark 6.4.13. *Define the set $\Gamma \doteq \mathcal{E}^m \setminus \Gamma_{coinc}$ (where Γ_{coinc} is defined in equation* (2.2)*). It is of interest to define and characterize the vector-valued function $W^* : \Gamma \mapsto \Omega$ that associates to each non-degenerate vector of generators' positions a set of weights such that the corresponding power diagram is equitable. Precisely, we define $W^*(G)$ as $W^*(G) \doteq \lim_{t \to \infty} W(t)$, where $W(t) = (w_1(t), \ldots, w_m(t))$ is the solution of the differential equation, with* fixed *vector of generators' positions G,*

$$\dot{w}_i(t) = -\frac{\partial H_V}{\partial w_i}(W(t)), \quad w_i(0) = 0, \quad i \in I_m.$$

By theorem 6.4.9, $W^*(G)$ is a well-defined quantity (since the limit exists) and corresponds to an equitable power diagram. The next lemma characterizes $W^*(G)$.

Lemma 6.4.14. *The function W^* is continuous on Γ.*

Proof. See Appendix. □

6.4.5 On the use of power diagrams

A natural question that arises is whether a similar result can be obtained by using Voronoi diagrams (of which power diagrams are a generalization). The answer is positive if we

constrain f to be constant over $\mathcal{E}$, but it is negative for general measures f, as we briefly discuss next.

Indeed, when f is *constant* over $\mathcal{E}$, an equitable Voronoi diagram always exists. We prove this result in a slightly more general setup.

Definition 6.4.14.1 (Unimodal Property)**.** *Let $\mathcal{E} \subset \mathbb{R}^d$ be a bounded, measurable set (not necessarily convex). We say that $\mathcal{E}$ enjoys the Unimodal Property if there exists a unit vector $v \in \mathbb{R}^d$ such that the following holds. For each $s \in \mathbb{R}$, define the slice $\mathcal{E}^s \doteq \{x \in \mathcal{E}, \quad v \cdot x = s\}$, and call $\psi(s) \doteq m_{d-1}(\mathcal{E}^s)$ the $(d-1)$-dimensional Lebesgue measure of the slice. Then, the function ψ is unimodal. In other words, ψ attains its global maximum at a point $\bar{s}$, is increasing on $(-\infty, \bar{s}]$, and decreasing on $[\bar{s}, \infty)$.*

The Unimodal Property (notice that every compact, convex set enjoys such property) turns out to be a sufficient condition for the existence of equitable Voronoi diagrams for bounded measurable sets (with respect to constant f).

Theorem 6.4.15. *If $\mathcal{E} \subset \mathbb{R}^d$ is any bounded measurable set satisfying the Unimodal Property and f is constant on $\mathcal{E}$, then for every $m \geq 1$ there exists an equitable Voronoi diagram with m (Voronoi) generators.*

Proof. See Appendix. □

Then, an equitable Voronoi diagram can be achieved by using a gradient descent law conceptually similar to the one discussed previously (the details are presented in [79]). We emphasize that the existence result on equitable Voronoi diagrams seems to be new in the rich literature on Voronoi tessellations.

While an equitable Voronoi diagram always exists when f is *constant* over $\mathcal{E}$, in general, for non-constant f, an equitable Voronoi diagram fails to exist, as the following counterexample shows.

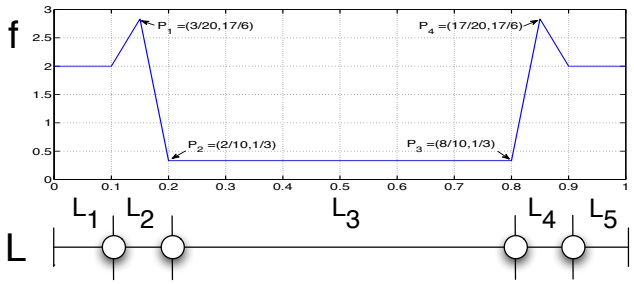


Figure 6-3: Example of non-existence of an equitable Voronoi diagram on a line. The above tessellation is an equitable partition, but not a Voronoi diagram.

Example 6.4.16 (Existence problem on a line)**.** *Consider a one-dimensional Voronoi diagram. In this case a Voronoi cell is a half line or a line segment (called a Voronoi line), and Voronoi vertices are end points of Voronoi lines. It is easy to notice that the boundary point between two adjacent Voronoi lines is the mid-point of the generators of those Voronoi lines. Consider the measure f in Figure 6-3, whose support is the interval $[0, 1]$. Assume $m = 5$. Let b_i $(i = 1, \ldots, 4)$ be the position of the ith rightmost boundary point and g_i be the position of the ith rightmost generator $(i = 1, \ldots, 5)$. It is easy to verify that the only*

admissible configuration for the boundary points in order to obtain an equitable Voronoi diagram is the one depicted in Figure 6-3. Now, by the Perpendicular Bisector Property, we require:

$$\begin{cases} g_3 - b_2 = b_2 - g_2 \\ g_4 - b_3 = b_3 - g_3 \end{cases}$$

Therefore, we would require $g_4 - g_2 = 2(b_3 - b_2) = 1.2$*; this is impossible, since* $g_2 \in [0.1, 0.2]$ *and* $g_4 \in [0.8, 0.9]$.

6.5 Distributed Algorithms for Equitable Partitions with Special Properties

The gradient descent law (6.12), although effective in providing convex equitable partitions, can yield long and "skinny" subregions. In this section, we provide spatially-distributed algorithms to obtain convex equitable partitions with special properties. The key idea is that, to obtain an equitable power diagram, changing the weights, while maintaining the generators *fixed*, is sufficient. Thus, we can use the degrees of freedom given by the positions of the generators to optimize *secondary* cost functionals. Specifically, we now assume that both generators' weights and their positions obey a first order dynamical behavior

$$\begin{cases} \dot{w}_i = u_i^w, \\ \dot{g}_i = u_i^g. \end{cases}$$

Define the set

$$\tilde{S} \doteq \Big\{ \Big((g_1, w_1), \dots, (g_m, w_m) \Big) \in (\mathcal{E} \times \mathbb{R})^m \,|\, g_i \neq g_j \text{ for all } i \neq j, \text{and } f_{V_i} > 0 \ \ \forall i \in I_m \Big\}. \tag{6.13}$$

The *primary* objective is to achieve a convex equitable partition and is captured, similarly as before, by the cost function $\tilde{H}_V : \tilde{S} \mapsto \mathbb{R}_{>0}$

$$\tilde{H}_V(G_W) \doteq \sum_{i=1}^{m} f_{V_i(G_W)}^{-1}.$$

We have the following

Theorem 6.5.1. *Given a measure* f*, the function* $\tilde{H}_V$ *is continuously differentiable on* $\tilde{S}$*, where for each* $i \in \{1, \dots, m\}$

$$\begin{aligned} \frac{\partial \tilde{H}_V}{\partial\, g_i} &= \sum_{j \in N_i} \Big(\frac{1}{f_{V_j}^2} - \frac{1}{f_{V_i}^2} \Big) \int_{\Delta_{ij}} \frac{(x - g_i)}{\gamma_{ij}} f(x)\, dx, \\ \frac{\partial \tilde{H}_V}{\partial\, w_i} &= \sum_{j \in N_i} \Big(\frac{1}{f_{V_j}^2} - \frac{1}{f_{V_i}^2} \Big) \int_{\Delta_{ij}} \frac{1}{2\gamma_{ij}} f(x)\, dx. \end{aligned} \tag{6.14}$$

Furthermore, the critical points of $\tilde{H}_V$ *are generators' positions and weights with the property that all power cells have measure equal to* $f_{\mathcal{E}}/m$.

Proof. The proof of this theorem is very similar to the proof of theorem 6.4.6; we omit

it in the interest of brevity (the derivation of the partial derivative $\frac{\partial \tilde{H}_V}{\partial g_i}$ can be found in [80]). □

Notice that the computation of the gradient in theorem 6.5.1 is spatially distributed over the power-Delaunay graph. For short, we define the vectors $v_{\pm\partial\tilde{H}_i} \doteq \pm\frac{\partial \tilde{H}_V}{\partial g_i}$.

Three possible *secondary* objectives are discussed in the remainder of this section.

6.5.1 Obtaining power diagrams similar to centroidal power diagrams

Define the *mass* and the *centroid* of the power cell V_i, $i \in I_m$, as

$$M_{V_i} = \int_{V_i} f(x)\,dx, \quad C_{V_i} = \frac{1}{M_{V_i}} \int_{V_i} x f(x)\,dx.$$

In this section we are interested in the situation where $g_i = C_{V_i}$, for all $i \in I_m$. We call such a power diagram a *centroidal power diagram*. The main motivation to study centroidal power diagrams is that, as it will be discussed in Section 6.5.3, the corresponding cells, under certain conditions, are *similar* in shape to regular polygons.

A natural candidate control law to try to obtain a centroidal and equitable power diagram (or at least a *good* approximation of it) is to let the positions of the generators move toward the centroids of the corresponding regions of dominance, when this motion does not increase the disagreement between the measures of the cells (i.e., it does not make the time derivative of $\tilde{H}_V$ positive).

First we introduce a C^∞ saturation function as follows:

$$\Theta(x) \doteq \begin{cases} 0 & \text{for } x \leq 0\ , \\ \exp\left(-\frac{1}{(\beta x)^2}\right) & \text{for } x > 0, \quad \beta \in \mathbb{R}_{>0}. \end{cases}$$

Moreover, we define the vector $v_{C,g_i} \doteq C_{V_i} - g_i$. Then, we set up the following control law defined over the set $\tilde{S}$, where we assume that the partition $\mathcal{V}(G_W) = \{V_1(G_W), \ldots, V_m(G_W)\}$ is continuously updated,

$$\begin{aligned} \dot{w}_i &= -\frac{\partial \tilde{H}_V}{\partial w_i}, \\ \dot{g}_i &= \frac{2}{\pi} \arctan\left[\frac{\|v_{-\partial\tilde{H}_i}\|^2}{\alpha}\right] \Theta(v_{C,g_i} \cdot v_{-\partial\tilde{H}_i})\ v_{C,g_i}, \qquad \alpha \in \mathbb{R}_{>0}. \end{aligned} \tag{6.15}$$

In other words, g_i moves toward the centroid of its cell *if and only if* this motion is compatible with the minimization of $\tilde{H}_V$. In particular, the term $\arctan\left(\|v_{-\partial\tilde{H}_i}\|^2/\alpha\right)$ is needed to make the right hand side of (6.15) C^1, while the term $\Theta(v_{C,g_i} \cdot v_{-\partial\tilde{H}_i})$ is needed to make the right hand side of (6.15) compatible with the minimization of $\tilde{H}_V$. To prove that the vector field is C^1 it is simply sufficient to observe that it is the composition and product of C^1 functions. Furthermore, the compatibility condition of the flow (6.15) with the minimization of $\tilde{H}_V$ stems from the fact that $\dot{g}_i = 0$ as long as $v_{C,g_i} \cdot v_{-\partial\tilde{H}_i} \leq 0$, due to the presence of $\Theta(\cdot)$. Notice that the computation of the right hand side of (6.15) is spatially distributed over the power-Delaunay graph.

As in many algorithms that involve the update of generators of Voronoi diagrams, it is possible that under control law (6.15) there exists a time t^* and $i, j \in I_m$ such that

$g_i(t^*) = g_j(t^*)$. In such a case, either the power diagram is not defined (when $w_i(t^*) = w_j(t^*)$), or there is an empty cell ($w_i(t^*) \neq w_j(t^*)$), and there is no obvious way to specify the behavior of control law (6.15) for these singularity points. Then, to make the set $\tilde{S}$ positively invariant, we have to slightly modify the update equation for the positions of the generators. The idea is to stop the positions of two generators when they are *close* and on a *collision course*.

Define, for $\Delta \in \mathbb{R}_{>0}$, the set

$$M_i(G, \Delta) \doteq \{g_j \in G \,|\, \|g_i - g_j\| \leq \Delta,\, g_j \neq g_i\}.$$

In other words, M_i is the set of generators' positions within an (Euclidean) distance Δ from g_i. For $\delta \in \mathbb{R}_{>0}$, $\delta < \Delta$, define the gain function $\psi(\rho, \vartheta) : [0, \Delta] \times [0, 2\pi] \mapsto \mathbb{R}_{\geq 0}$ (see Figure 6-4):

$$\psi(\rho, \vartheta) = \begin{cases} \frac{\rho - \delta}{\Delta - \delta} & \text{if } \delta < \rho \leq \Delta \quad \text{and } 0 \leq \vartheta < \pi, \\ \frac{\rho - \delta}{\Delta - \delta}(1 + \sin\vartheta) - \sin\vartheta & \text{if } \delta < \rho \leq \Delta \quad \text{and } \pi \leq \vartheta \leq 2\pi, \\ 0 & \text{if } \rho \leq \delta \quad \text{and } 0 \leq \vartheta < \pi, \\ -\frac{\rho}{\delta}\sin\vartheta & \text{if } \rho \leq \delta \quad \text{and } \pi \leq \vartheta \leq 2\pi, \end{cases} \tag{6.16}$$

It is easy to see that $\psi(\cdot)$ is a continuous function on $[0, \Delta] \times [0, 2\pi]$ and it is globally Lipschitz there. Function $\psi(\cdot)$ has the following motivation. Let ρ be equal to $\|g_j - g_i\|$, and let v_x be a vector such that the tern $\{v_x, (g_j - g_i), v_x \times (g_j - g_i)\}$ is an orthogonal basis of $\mathbb{R}^3$, co-orientied with the standard basis. In the Figure 6-4, v_x corresponds to the x axis and $g_j - g_i$ corresponds to the y axis. Then the angle ϑ_{ij} is the angle between v_x and v_{C,g_i}, where $0 \leq \vartheta_{ij} \leq 2\pi$. If $\rho \leq \delta$ and $0 \leq \vartheta_{ij} < \pi$, then g_i is *close* to g_j and it is on a *collision course*, thus we set the gain to zero. Similar considerations hold for the other three cases; for example, if $\rho \leq \delta$ and $\pi < \vartheta_{ij} < 2\pi$, the generators are *close*, but they are not on a collision course, thus the gain is positive.

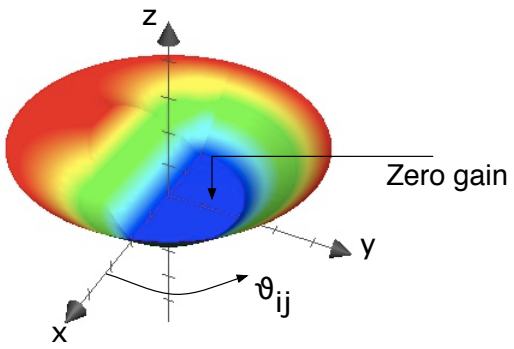


Figure 6-4: Gain function used to avoid that the positions of two power generators can coincide.

Thus, we modify control law (6.15) as follows:

$$\begin{aligned}
\dot{w}_i &= -\frac{\partial \tilde{H}_V}{\partial w_i} \doteq u_i^{\text{cent,w}}, \\
\dot{g}_i &= \frac{2}{\pi} \arctan\left[\frac{\|v_{-\partial \tilde{H}_i}\|^2}{\alpha}\right] \Theta(v_{C,g_i} \cdot v_{-\partial \tilde{H}_i})\, v_{C,g_i} \prod_{g_j \in M_i(G,\Delta)} \psi\Big(\|g_j - g_i\|,\ \vartheta_{ij}\Big) \doteq u_i^{\text{cent,g}}.
\end{aligned} \tag{6.17}$$

If $M_i(G, \Delta)$ is the empty set, then we have an empty product, whose numerical value is 1. Notice that the right hand side of (6.17) is Lipschitz continuous, since it is a product of C^1 functions and Lipschitz continuous functions, and it can be still computed in a spatially-distributed way (in fact, it only depends on generators that are neighbors in the power diagram, and whose positions are within a distance Δ). One can prove the following result.

Theorem 6.5.2. *Consider the vector field on $\tilde{S}$ defined by equation* (6.17)*. Then generators' positions and weights starting at $t = 0$ at $G_W(0) \in \tilde{S}$, and evolving under* (6.17) *remain in $\tilde{S}$ and converge asymptotically to the set of critical points of the primary objective function $\tilde{H}_V$ (i.e., to the set of vectors of generators' positions and weights that yield an equitable power diagram).*

Proof. The proof is virtually identical to the one of theorem 6.4.9, and we omit it in the interest of brevity. We only notice that $\tilde{H}_V$ is non-increasing along system trajectories

$$\dot{\tilde{H}}_V = \sum_{i=1}^{m} \frac{\partial \tilde{H}_V}{\partial g_i} \cdot \dot{g}_i + \frac{\partial \tilde{H}_V}{\partial w_i} \dot{w}_i = \sum_{i=1}^{m} \underbrace{\frac{\partial \tilde{H}_V}{\partial g_i} \cdot \dot{g}_i}_{\leq 0} - \Big(\frac{\partial \tilde{H}_V}{\partial w_i}\Big)^2 \leq 0.$$

Moreover, the components of the vector field (6.17) for the position of each generator are either zero or point toward $\mathcal{E}$ (since the centroid of a cell must be within $\mathcal{E}$); therefore, each generator will remain within the compact set $\mathcal{E}$. □

6.5.2 Obtaining power diagrams "close" to Voronoi diagrams

In some applications it could be preferable to have power diagrams as *close* as possible to Voronoi diagrams (e.g., in DVR problems, see chapter 7). This issue is of particular interest for the setting with non-uniform density, when an equitable Voronoi diagram could fail to exist. The objective of obtaining a power diagram *close* to a Voronoi diagram can be translated into the minimization of the function $K : \mathbb{R}^m \to \mathbb{R}_{\geq 0}$:

$$K(W) \doteq \frac{1}{2} \sum_{i=1}^{m} w_i^2;$$

when $w_i = 0$ for all $i \in I_m$, we have $K = 0$ and the corresponding power diagram coincides with a Voronoi diagram. To include the minimization of the secondary objective K, it is natural to consider, instead of (6.12), the following update law for the weights:

$$\dot{w}_i = -\frac{\partial H_V}{\partial w_i} - \frac{\partial K}{\partial w_i} = -\frac{\partial H_V}{\partial w_i} - w_i. \tag{6.18}$$

However, H_V is no longer a valid Lyapunov function for system (6.18). The idea, then, is to let the positions of the generators move so that $\frac{\partial \tilde{H}_V}{\partial g_i} \cdot \dot{g}_i - \frac{\partial \tilde{H}_V}{\partial w_i}\frac{\partial K}{\partial w_i} = 0$. In other words, the dynamics of generators' positions are used to compensate the effect of the term $-w_i$ (present in the weights' dynamics) on the time derivative of $\tilde{H}_V$.

Thus, we set up the following control law, with ε_1, ε_2 and ε_3 positive *small* constants, $\varepsilon_2 > \varepsilon_1$,

$$\begin{aligned} \dot{w}_i &= -\frac{\partial \tilde{H}_V}{\partial w_i} - w_i \mathrm{sat}_{\varepsilon_1,\varepsilon_2}\Big(\|v_{\partial \tilde{H}_i}\|\Big)\,\mathrm{sat}_{0,\varepsilon_3}\Big(\mathrm{dist}(g_i, \partial V_i)\Big), \\ \dot{g}_i &= w_i \frac{\partial \tilde{H}_V}{\partial w_i}\frac{v_{\partial \tilde{H}_i}}{\| v_{\partial \tilde{H}_i}\|^2}\,\mathrm{sat}_{\varepsilon_1,\varepsilon_2}\Big(\|v_{\partial \tilde{H}_i}\|\Big)\,\mathrm{sat}_{0,\varepsilon_3}\Big(\mathrm{dist}(g_i, \partial V_i)\Big). \end{aligned} \tag{6.19}$$

The gain $\mathrm{sat}_{\varepsilon_1,\varepsilon_2}\Big(\|v_{\partial \tilde{H}_i}\|\Big)$ is needed to make the right hand side of (6.19) Lipschitz continuous, while the gain $\mathrm{sat}_{0,\varepsilon_3}\Big(\mathrm{dist}(g_i, \partial V_i)\Big)$ avoids that generators' positions can leave the environment. Notice that the computation of the right hand side of (6.19) is spatially distributed over the power-Delaunay graph.

As before, it is possible that under control law (6.19) there exists a time t^* and $i, j \in I_m$ such that $g_i(t^*) = g_j(t^*)$. Thus, similarly as before, we modify the update equations (6.19) as follows

$$\begin{aligned} \dot{w}_i &= -\frac{\partial \tilde{H}_V}{\partial w_i} - w_i \mathrm{sat}_{\varepsilon_1,\varepsilon_2}\Big(\|v_{\partial \tilde{H}_i}\|\Big)\,\mathrm{sat}_{0,\varepsilon_3}\Big(\mathrm{dist}(g_i, \partial V_i)\Big) \cdot \prod_{g_j \in M_i(G,\Delta)} \psi\Big(\|g_j - g_i\|,\ \vartheta_{ij}\Big) \\ &\doteq u_i^{\mathrm{vor,w}}, \\ \dot{g}_i &= w_i \frac{\partial \tilde{H}_V}{\partial w_i}\frac{v_{\partial \tilde{H}_i}}{\| v_{\partial \tilde{H}_i}\|^2}\mathrm{sat}_{\varepsilon_1,\varepsilon_2}\Big(\|v_{\partial \tilde{H}_i}\|\Big)\,\mathrm{sat}_{0,\varepsilon_3}\Big(\mathrm{dist}(g_i, \partial V_i)\Big) \cdot \prod_{g_j \in M_i(G,\Delta)} \psi\Big(\|g_j - g_i\|,\ \vartheta_{ij}\Big) \\ &\doteq u_i^{\mathrm{vor,g}}, \end{aligned} \tag{6.20}$$

where ϑ_{ij} is defined as in Section 6.5.1, with $w_i \frac{\partial \tilde{H}_V}{\partial w_i} v_{\partial \tilde{H}_i}$ replacing v_{C,g_i}.

One can prove the following result.

Theorem 6.5.3. *Consider the vector field on $\tilde{S}$ defined by equation* (6.20)*. Then generators' positions and weights starting at $t = 0$ at $G_W(0) \in \tilde{S}$, and evolving under* (6.20) *remain in $\tilde{S}$ and converge asymptotically to the set of critical points of the primary objective function $\tilde{H}_V$ (i.e., to the set of vectors of generators' positions and weights that yield an equitable power diagram).*

Proof. Consider $\tilde{H}_V$ as a Lyapunov function candidate. First, we prove that set $\tilde{S}$ is positively invariant with respect to (6.20). Indeed, by definition of (6.20), we have $g_i \neq g_j$ for $i \neq j$ for all $t \geq 0$ (therefore, the power diagram is always well defined). Moreover, it is straightforward to see that $\dot{\tilde{H}}_V \leq 0$. Therefore, it holds

$$f^{-1}_{V_i(G_W(t))} \leq \tilde{H}_V(G_W(t)) \leq \tilde{H}_V(G_W(0)), \quad i \in I_m,\ t \geq 0.$$

Since the measures of the power cells depend continuously on generators' positions and

weights, we conclude that the measures of all power cells will be bounded away from zero. Finally, since $\dot{g}_i = 0$ on the boundary of $\mathcal{E}$ for all $i \in I_m$, each generator will remain within the compact set $\mathcal{E}$. Thus, generators' positions and weights will belong to $\tilde{S}$ for all $t \geq 0$, that is, $G_W(t) \in \tilde{S}\ \forall t \geq 0$.

Second, as shown before, $\tilde{H}_V : \tilde{S} \to \mathbb{R}_{\geq 0}$ is non-increasing along system trajectories, i.e., $\dot{\tilde{H}}_V(G_W) \leq 0$ in $\tilde{S}$.

Third, all trajectories with initial conditions in $\tilde{S}$ are bounded. Indeed, we have already shown that each generator remains within the compact set $\mathcal{E}$ under control law (6.20). As far as the weights are concerned, we start by noticing that the time derivative of the sum of the weights is

$$\frac{\partial \sum_{i=1}^m w_i}{\partial t} = -\sum_{i=1}^m w_i \text{sat}_{\varepsilon_1,\varepsilon_2}\Big(\|v_{\partial \tilde{H}_i}\|\Big)\, \text{sat}_{0,\varepsilon_3}\Big(\text{dist}(g_i, \partial V_i)\Big) \prod_{g_j \in M_i(G,\Delta)} \psi\Big(\|g_j - g_i\|,\ \vartheta_{ij}\Big),$$

since, similarly as in the proof of theorem 6.4.9, $\sum_{i=1}^m \frac{\partial \tilde{H}_V}{\partial w_i} = 0$. Moreover, the magnitude of the difference between any two weights is bounded by a constant, that is,

$$|w_i - w_j| \leq \alpha \quad \text{for all } i,\, j \in I_m; \tag{6.21}$$

if, by contradiction, the magnitude of the difference between any two weights could become arbitrarily large, the measure of at least one power cell would vanish, since the positions of the generators are confined within $\mathcal{E}$. Assume, by the sake of contradiction, that weights' trajectories are unbounded. This means that

$$\forall R > 0 \quad \exists t \geq 0 \text{ and } \exists\, j \in I_m \quad \text{such that} \quad |w_j(t)| \geq R.$$

For simplicity, assume that $w_i(0) = 0$ for all $i \in I_m$ (the extension to arbitrary initial conditions in $\tilde{S}$ is straightforward). Choose $R = 2m\alpha$ and let t_2 be the time instant such that $|w_j(t_2)| = R$, for some $j \in I_m$. Without loss of generality, assume that $w_j(t_2) > 0$. Because of constraint (6.21), we have $\sum_{i=1}^m w_i(t_2) \geq \frac{\alpha}{2} m(3m+1)$. Let t_1 be the *last* time before t_2 such that $w_j(t) = m\alpha$; because of continuity of trajectories, t_1 is well-defined. Then, because of constraint (6.21), we have (i) $\sum_{i=1}^m w_i(t_1) \leq \frac{\alpha}{2} m(3m-1) < \sum_{i=1}^m w_i(t_2)$, *and* (ii) $\frac{\partial \sum_{i=1}^m w_i(t)}{\partial t} \leq 0$ for $t \in [t_1,\, t_2]$ (since $w_j(t) \geq m\alpha$ for all $t \in [t_1,\, t_2]$ and equation (6.21) imply $\min_{i \in I_m} w_i(t) > 0$ for all $t \in [t_1,\, t_2]$); thus, we get a contradiction.

Finally, by theorem 6.5.1, $\tilde{H}_V$ is continuously differentiable in $\tilde{S}$. Hence, by invoking the LaSalle invariance principle (see, for instance, [25]), under the descent flow (6.20), generators' positions and weights will converge asymptotically to the *set* of critical points of $\tilde{H}_V$, that is not empty by theorem 6.4.1. □

6.5.3 Obtaining cells similar to regular polygons

In many applications, it is preferable to avoid long and thin subregions. For example, in applications where a mobile agent has to service demands distributed in its own subregion, the maximum travel distance is minimized when the subregion is a circle. Thus, it is of interest to have subregions whose shapes show *circular symmetry*, i.e., that are similar to regular polygons.

Define, now, the distortion function $L_V : (\mathcal{E} \times \mathbb{R})^m \setminus \Gamma_{\text{coinc}} \mapsto \mathbb{R}_{\geq 0}$: $\sum_{i=1}^m \int_{V_i} \|x - g_i\|^2 f(x) dx$ (where V_i is the ith cell in the corresponding power diagram). In [71] it is

shown that, when m is large, for the centroidal Voronoi diagram (i.e., centroidal power diagram with equal weights) that minimizes L_V, all cells are approximately congruent to a *regular hexagon*, i.e., to a polygon with considerable circular symmetry (see Section 6.6 for a more in-depth discussion about circular symmetry).

Indeed, it is possible to obtain a power diagram that is *close* to a centroidal Voronoi diagram by combining control laws (6.17) and (6.20). In particular, we set up the following (spatially-distributed) control law:

$$\begin{aligned} \dot{w}_i =& u_i^{\text{cent,w}} + u_i^{\text{vor,w}}, \\ \dot{g}_i =& u_i^{\text{cent,g}} + u_i^{\text{vor,g}}. \end{aligned} \tag{6.22}$$

Combining the results of theorem 6.5.2 and theorem 6.5.3, we argue that with control law (6.22) it is possible to obtain equitable power diagrams with cells *similar* to regular polygons, i.e. that show circular symmetry.

6.6 Simulations and Discussion

In this section we verify through simulation the effectiveness of the optimization for the secondary objectives. For brevity, we discuss only control law (6.22). We introduce two criteria to judge, respectively, *closeness* to a Voronoi diagram, and circular symmetry of a partition.

6.6.1 Closeness to Voronoi diagrams

In a Voronoi diagram, the intersection between the bisector of two neighboring generators g_i and g_j and the line segment joining g_i and g_j is the midpoint $g_{ij}^{\text{vor}} \doteq (g_i + g_j)/2$. Then, if we define g_{ij}^{pow} as the intersection, in a power diagram, between the bisector of two neighboring generators (g_i, w_i) and (g_j, w_j) and the line segment joining their positions g_i and g_j, a possible way to measure the *distance* η of a power diagram from a Voronoi diagram is the following:

$$\eta \doteq \frac{1}{2N} \sum_{i=1}^{m} \sum_{j \in N_i} \frac{\|g_{ij}^{\text{pow}} - g_{ij}^{\text{vor}}\|}{0.5\, \gamma_{ij}}, \tag{6.23}$$

where N is the number of neighboring relationships and, as before, $\gamma_{ij} = \|g_j - g_i\|$. Clearly, if the power diagram is also a Voronoi diagram (i.e., if all weights are equal), we have $\eta = 0$. We will also refer to η as the *Voronoi defect* of a power diagram.

6.6.2 Circular symmetry of a partition

A quantitative manifestation of circular symmetry is the well-known *isoperimetric inequality* which states that among all planar objects of a given perimeter, the circle encloses the largest area. More precisely, given a planar region V with perimeter p_V and area $|V|$, then $p_V^2 - 4\pi|V| \geq 0$, and equality holds if and only if V is a circle. Then, we can define the *isoperimetric ratio* as follows: $Q_V = \frac{4\pi|V|}{p_V^2}$; by the isoperimetric inequality, $Q_V \leq 1$, with equality only for circles. Interestingly, for a regular n-gon the isoperimetric ratio Q_n is $Q_n = \frac{\pi}{n \tan \frac{\pi}{n}}$, which converges to 1 for $n \to \infty$. Accordingly, given a partition $\{\mathcal{E}_i\}_{i=1}^m$, we define, as a measure for the circular symmetry of the partition, the isoperimetric ratio $Q_{\{\mathcal{E}_i\}_{i=1}^m} \doteq \frac{1}{m} \sum_{i=1}^m Q_{\mathcal{E}_i}$.

Table 6.1: Performance of control law (6.22).

f	$\mathbb{E}[\epsilon]$	$\max \epsilon$	$\mathbb{E}[\eta]$	$\max \eta$	$\mathbb{E}[Q_{\mathcal{V}}]$	$\min Q_{\mathcal{V}}$
unif	$3.8\,10^{-3}$	0.16	0.01	0.03	0.73	0.66
gauss	$8.9\,10^{-2}$	0.15	0.02	0.04	0.75	0.69

6.6.3 Simulation results

We simulate ten agents providing service in the unit square $\mathcal{E}$. Agents' initial positions are independently and uniformly distributed over $\mathcal{E}$, and all weights are initialized to zero. Time is discretized with a step $dt = 0.01$, and each simulation run consists of 800 iterations (thus, the final time is $T = 8$). Define the area error ϵ as $\epsilon \doteq (f_{i_{\max}} - f_{i_{\min}})/(f_{\mathcal{E}}/m)$, evaluated at time $T = 8$; in the definition of ϵ, $f_{i_{\max}}$ is the measure of the region of dominance with maximum measure and $f_{i_{\min}}$ is the measure of the region of dominance with minimum measure.

First, we consider a measure f *uniform* over $\mathcal{E}$, in particular $f \equiv 1$. Therefore, we have $f_{\mathcal{E}} = 1$ and agents should reach a partition in which each region of dominance has measure equal to 0.1. For this case, we run 50 simulations.

Then, we consider a measure f that follows a gaussian distribution, namely $f(x, y) = e^{-5((x-0.8)^2+(y-0.8)^2)}$, $(x, y) \in \mathcal{E}$, whose peak is at the north-east corner of the unit square. Therefore, we have $f_{\mathcal{E}} \approx 0.336$, and agents should reach a partition in which each region of dominance has measure equal to 0.0336. For this case, we run 20 simulations.

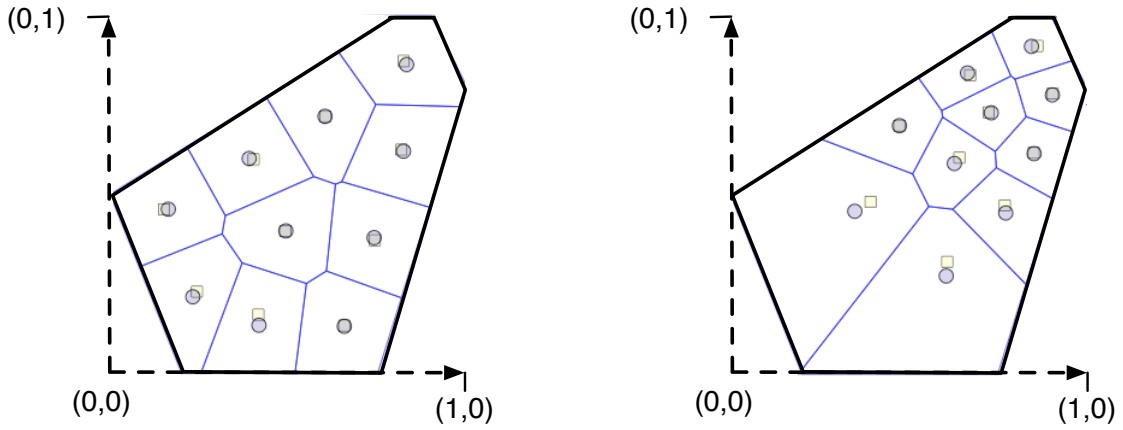
Table 6.1 summarizes simulation results for the uniform f (f=unif) and the gaussian f (f=gauss). Expectation and worst case values of area error ϵ, Voronoi defect η, and isoperimetric ratio $Q_{\mathcal{V}}$ are with respect to 50 runs for uniform f, and 20 runs for gaussian f. Notice that for both measures, after 800 iterations, (i) the worst case area error is no more than 16%, (ii) the worst case η is very close to 0, and, finally, (iii) cells have, approximately, the circular symmetry of squares (since $Q_4 \approx 0.78$). Therefore, convergence to a convex equitable partition with the desired properties (i.e., closeness to Voronoi diagrams and circular symmetry) seems to be robust. Figure 6-5 shows the typical equitable partitions that are achieved with control law (6.22) with 10 agents.

6.7 Conclusion

In this chapter we presented provably correct, spatially-distributed algorithms that allow a team of agents to achieve a convex and equitable partition of a convex environment. These algorithms could find applications in many problems, including dynamic vehicle routing (as we will show in the next chapter), and wireless networks. All the algorithms that we proposed are synchronous: a possible line of future research would be to devise similar algorithms that are amenable to asynchronous implementation. Implementation on a real robotic network would also be important to assess closed-loop robustness and feasibility.

Appendix

Proof of theorem 6.4.15. The proof mainly relies on [22]. Let v be the unit vector considered in the definition of the Unimodal Property. Then, there exist unique values $s_0 < s_1 < \ldots <$



(a) Typical equitable partition of $\mathcal{E}$ for $f(x,y) = 1$.

(b) Typical equitable partition of $\mathcal{E}$ for $f(x,y) = e^{-5((x-0.8)^2+(y-0.8)^2)}$.

Figure 6-5: Typical equitable partitions achieved by using control law (6.22). The yellow squares represent the position of the generators, while the blue circles represent the centroids. Notice how each bisector intersects the line segment joining the two corresponding power neighbors almost at the midpoint; hence both partitions are very close to Voronoi partitions. Compare with Figure 6-1.

s_m such that $s_0 = \inf\{s; \mathcal{E}^s \neq \emptyset\}$, $s_m = \sup\{s; \mathcal{E}^s \neq \emptyset\}$, and

$$f_{\{x\in\mathcal{E};\, v\cdot x\leq s_k\}} = \frac{k}{m} f_{\mathcal{E}}, \quad k = 1, \ldots, m-1. \tag{6.24}$$

Consider the intervals $I_i \doteq [s_{i-1}, s_i]$, $i \in \{1, \ldots, m\}$. We claim that one can choose points $g_i = t_i v \in \mathbb{R}^d$, $i \in \{1, \ldots, m\}$, such that $t_i \in I_i$ and the corresponding Voronoi diagram is

$$\begin{aligned} V_i &= \{x \in \mathcal{E}; \quad \|x - g_i\| = \min_k \|x - g_k\|\} \\ &= \{x \in \mathcal{E}; \quad v \cdot x \in [s_{i-1}, s_i]\}. \end{aligned} \tag{6.25}$$

Together, equation (6.24) and equation (6.25) yield the desired result.

Since, by assumption, $\mathcal{E}$ enjoys the Unimodal Property, there exists an index $\kappa \in \{1, \ldots, m\}$ such that the length of the intervals $I_i = [s_{i-1}, s_i]$ decreases as i ranges from 1 to κ, then increases as i ranges from κ to m. Let $I_\kappa = [s_{\kappa-1}, s_\kappa]$ be the smallest of these intervals, and define

$$t_\kappa \doteq \frac{s_{\kappa-1} + s_\kappa}{2} \in I_\kappa.$$

By induction, for i increasing from κ to $m-1$, define t_{i+1} be the symmetric to t_i with respect to s_i, so that

$$t_{i+1} = 2s_i - t_i \quad i = \kappa, \kappa+1, \ldots, m-1.$$

Since the length of I_{i+1} is larger than the length of I_i, we have

$$t_i \in I_i \Rightarrow t_{i+1} \in I_{i+1}. \tag{6.26}$$

Similarly, for i decreasing from κ to 1, we define

$$t_{i-1} = 2s_{i-1} - t_i, \quad i = \kappa, \kappa - 1, \ldots, 2.$$

Since the interval I_{i-1} is now larger than the interval I_i, we have

$$t_i \in I_i \Rightarrow t_{i-1} \in I_{i-1}. \tag{6.27}$$

Equations (6.26)-(6.27) imply $t_i \in I_i$ for all $i = 1, \ldots, m$. Hence the second equality in equation (6.25) holds. □

We now specialize the theorem to the case when $\mathcal{E}$ is convex.

Corollary 6.7.1. *Let $\mathcal{E} \subset \mathbb{R}^d$ be a compact, convex set, and f be constant on $\mathcal{E}$. Then for every $m \geq 1$ there exist points $g_1, \ldots, g_m$ all in the interior of $\mathcal{E}$, such that the corresponding Voronoi diagram is equitable.*

Proof. Notice that every compact convex set enjoys the Unimodal Property, with an arbitrary choice of the unit vector v. By compactness, there exist points $a, b \in \mathcal{E}$ such that $\|b - a\| = \max_{z,z' \in \mathcal{E}} \|z - z'\|$. By a translation of coordinates, we can assume $a = 0$. Choose $v \doteq b/\|b\|$. Then the previous construction yields an equitable Voronoi diagram generated by m points $g_i = t_i v$ all in the interior of $\mathcal{E}$. □

Proof of Lemma 6.4.14. By theorem 6.4.9 and by its very definition $W^*(G)$ is the zero of the vector field $-\frac{\partial H_V}{\partial w_i}(W(t))$. Now let us denote with

$$K(W, G) \doteq -\frac{\partial H_V}{\partial w_i}(W),$$

the corresponding continuous function, viewed as a function of two independent set of variables, namely the weights $(w_1, \ldots, w_n) = W$ and the non-degenerate vector of generators' positions G. In order to prove that the assignment $G \mapsto W^*(G)$ is continuous, notice that by theorem 6.4.6 the function $K(W, G)$ is identically zero when restricted to the graph of W^*, namely $K(W^*(G), G) = 0$. The function W^* is continuous iff it is continuous in each of its argument. Fix, first, a generator $g_i \notin \partial\Gamma$ and consider for any $v \in \mathbb{R}^2$, the variation $(g_1, \ldots, g_{i-1}, g_i + hv, g_{i+1}, \ldots, g_m)$. Since $g_i \notin \partial\Gamma$, there always exists an $\epsilon > 0$, depending on g_i and v, such that for any h with $0 \leq h < \epsilon$, $(g_1, \ldots, g_{i-1}, g_i + hv, g_{i+1}, \ldots, g_m) \in \Gamma$. Now $K(W^*(g_1, \ldots, g_{i-1}, g_i + hv, g_{i+1}, \ldots, g_m), (g_1, \ldots, g_{i-1}, g_i + hv, g_{i+1}, \ldots, g_m)) = 0$ for any $0 \leq h < \epsilon$ by definition. Therefore, taking the limit for $h \to 0^+$, we still get zero. On the other hand, since K is continuous, we can take the limit inside K and we get

$$K(\lim_{h \to 0^+} W^*(g_1, \ldots, g_{i-1}, g_i + hv, g_{i+1}, \ldots), (g_1, \ldots, g_{i-1}, g_i, g_{i+1}, \ldots)) = 0.$$

Therefore, we have that

$$\lim_{h \to 0^+} W^*(g_1, \ldots, g_{i-1}, g_i + hv, g_{i+1}, \ldots, g_m) = W^*(g_1, \ldots, g_{i-1}, g_i, g_{i+1}, \ldots, g_m),$$

by the uniqueness in Ω of the value of W^* for which, given G, the function K vanishes. □

Chapter 7

Adaptive and Distributed Algorithms for DVR

In this chapter we leverage on the spatially-distributed algorithms developed in chapter 6 to obtain adaptive and distributed algorithms for DVR problems, in particular for the m-vehicle Dynamic Traveling Repairman Problem. In fact, to the best of our knowledge, all known control policies for the m-DTRP rely on centralized task assignment and are not robust against changes in the environment. This is clearly undesirable for applications involving large-scale robotic networks.

In many applications it is desirable to provide spatially-unbiased service, i.e., the same quality of service to all outstanding demands independently of their locations; e.g., in the UAV example discussed in the introduction, outstanding "threats" at the periphery of $\mathcal{E}$ should receive the same quality of service as the outstanding "threats" in the middle of $\mathcal{E}$. However, in light load, when most of the time the vehicles are idling, it is more reasonable to consider spatially-biased policies that aim at positioning the vehicles in the best a priori locations, i.e., at the locations that minimize the expected distance to the next demand. Accordingly, in this chapter we focus on policies for the m-DTRP that are constrained to provide spatially-unbiased service in heavy load, but can provide spatially-biased service in light load. With a slight abuse, we refer to policies satisfying this constraint as *unbiased policies*. The objective then is to devise *adaptive*, *spatially-distributed*, *scalable* unbiased policies with *provable performance guarantees*, that rely only on local exchange of information between neighboring vehicles, and do not require explicit knowledge of the global structure of the network.

Specifically, the contributions of this chapter are as follows. First, we present a new class of policies for the 1-DTRP that are adaptive (e.g., to the load conditions) and satisfy the aforementioned constraint about unbiased service. In particular, we propose the Divide & Conquer (DC) policy, whose performance depends on a design parameter $r \in \mathbb{N}$. If $r \to +\infty$, the policy is (i) provably optimal both in light- and in heavy-load conditions, and (ii) adaptive with respect to changes in the load conditions and in the statistics of the on-site service requirement; if, instead, $r = 1$, the policy is (i) provably optimal in light-load conditions and within a factor 2 of the optimal performance in heavy-load conditions, and (ii) adaptive with respect to all problem data, in particular, and perhaps surprisingly, it does not require *any* knowledge about the demand generation process. Moreover, by applying ideas of receding-horizon control to dynamic vehicle routing problems, we introduce the Receding Horizon (RH) policy, that also does not require any knowledge about the

Table 7.1: Adaptive policies for the 1-DTRP

Properties	**DC Policy,** $r \to \infty$	**DC Policy** $r = 1$	**RH Policy**	**UTSP Policy,** $r \to \infty$
Light-load performance	optimal	optimal	optimal	not optimal
Heavy-load performance	optimal	within 100% of the optimum	within 100% of the optimum*	optimal
Adaptive	yes, except for f	yes	yes	no

Table 7.2: Distributed and adaptive policies for the m-DTRP

Properties	m**-DC Policy,** $r \to \infty$	m**-RH Policy**	**UTSP Policy,** $r \to \infty$
Light-load performance	"locally" optimal	"locally" optimal	not optimal
Heavy-load performance	optimal for uniform f, optimal for any f when $\bar{s} = 0$, within m of optimal unbiased in general†	within 100% of optimal unbiased for uniform $f^{*,\dagger}$, within 100% of optimal unbiased for any f when $\bar{s} = 0^{*,\dagger}$, within $2m$ of optimal unbiased in general*,†	optimal
Adaptive	yes, except for f	yes, except for f	no
Distributed	yes	yes	no

demand generation process; we show that the RH policy is optimal in light load and stable in any load condition, and we heuristically argue that its performance is close to optimal in heavy-load conditions (in particular, we heuristically argue that the RH policy is the best available unbiased and adaptive policy for the 1-DTRP). Second, we show that specific partitioning policies, whereby the environment is partitioned among the vehicles and each vehicle follows a certain set of rules within its own region, are optimal in heavy-load conditions. Finally, by combining the DC policy with the spatially-distributed algorithms for environment partitioning developed in chapter 6, we design a routing policy for the m-DTRP (called m-DC policy) that (i) is spatially distributed, scalable to large networks, and adaptive to network changes, (ii) is within a constant factor of the optimal unbiased performance in heavy-load conditions (in particular, it is optimal when demands are uniformly dispersed over the environment or when the average on-site service time $\bar{s}$ is zero) and stabilizes the system in *any* load condition. Here, by network changes we mean changes in the number of vehicles, in the arrival rate of demands, and in the characterization of the on-site service requirement. Tables 7.1 and 7.2 provide a synoptic view about the main properties of the routing policies that we propose and analyze in this chapter; our policies are compared with the best spatially-unbiased policy available in the literature, i.e., the UTSP policy with $r \to \infty$ [16]. (In Tables 7.1 and 7.2 an asterisk $*$ signals that the result is heuristic, a dagger $\dagger$ means that the spatially-unbiased constraint might be violated in heavy load, f denotes the spatial probability density for the demands' locations, and m is the number of vehicles.)

This chapter is structured as follows. In section 7.1 we discuss our strategy to devise adaptive and distributed policies for the m-DTRP. In sections 7.2 and 7.3 we present and analyze, respectively, the Divide & Conquer policy and the Receding Horizon policy. In section 7.4, building upon the two previous policies for the 1-DTRP, we discuss partitioning policies and we design spatially-distributed algorithms for the m-DTRP. In section 7.5 we discuss the particular case when the on-site service time is zero, and in section 7.6 we present results from numerical experiments. Finally, in section 7.7, we draw some conclusions; in particular, we argue that the results of this chapter extend to a variety of DVR problems with complex demand models.

The work in this chapter is based on the journal article [81] and the preliminary conference paper [79].

7.1 Toward Adaptive, Distributed, Scalable Control Policies for the m-DTRP

A candidate adaptive, distributed, scalable control policy for the m-DTRP is the simple Nearest Neighbor (NN) policy: at each service completion epoch, each vehicle chooses to visit next the closest unserviced demand, if any, otherwise it stops at the current position. Because of the dependencies among the travel inter-demand distances, the analysis of the NN policy is difficult and no rigorous results have been obtained so far [14]; in particular, there are no rigorous results about its stability properties. Simulation experiments show that the NN policy performs like a biased policy in any load, and is not optimal *neither* in the light-load case *nor* in the heavy-load case. In particular, when f is uniform, in light load the steady-state system time under the NN policy is 40% larger than the steady-state system time under the (optimal) SQM policy, while in heavy load the steady-state system time under the NN policy is 50% larger than the steady-state system time under the (optimal) Biased TSP policy [14, 16]. Therefore, the NN policy *lacks* provable performance guarantees (in particular about stability), is *biased* in any load, and does *not* seem to achieve optimal performance neither in light load nor in heavy load.

The key idea that we will pursue in this chapter is that of partitioning policies, which were already introduced in chapter 6. The following definition of partitioning policy is conceptually identical to the one in section 6.2, but it emphasizes which single-vehicle policy is used in each subregion.

Definition 7.1.0.1. *Given a policy π for the 1-DTRP and m vehicles, a π-partitioning policy is a family of multi-vehicle policies such that*

1. *the environment $\mathcal{E}$ is partitioned into m openly disjoint subregions $\mathcal{E}_k$, $k \in \{1, \ldots, m\}$, whose union is $\mathcal{E}$,*

2. *one vehicle is assigned to each subregion (thus, there is a one-to-one correspondence between vehicles and subregions),*

3. *each vehicle executes the single-vehicle policy π to service demands that fall within its own subregion.*

Note that definition 7.1.0.1 does not specify how the environment is actually partitioned, therefore definition 7.1.0.1 describes a *family* of policies (one for each partitioning strategy) for the m-DTRP. The SQM policy, which is optimal *in light load*, is indeed a partitioning policy whereby $\mathcal{E}$ is partitioned according to a median Voronoi diagram that "minimizes" H_m, and each vehicle executes inside its own Voronoi cell the single-vehicle policy "service FCFS and return to the assigned median after each service completion". Moreover, in section 7.4, we will prove that *in heavy load*, given a single-vehicle, optimal unbiased policy π^* and m vehicles, there exists an unbiased π^*-partitioning policy that is optimal.

The above discussion leads to the following strategy: First, for the 1-DTRP, we design unbiased and adaptive control policies with provable performance guarantees. Then, by using the spatially-distributed algorithms for environment partitioning developed in chapter 6, we extend these single-vehicle policies to spatially-distributed multi-vehicle policies.

7.2 The Single-Vehicle Divide & Conquer Policy

In this section we introduce an unbiased and adaptive control policy for the 1-DTRP, which we call the Divide & Conquer (DC) policy (see section 2.5 for the definition of the DTRP problem). We describe the Divide & Conquer (DC) policy in Algorithm 5, where Q is the set of outstanding demands waiting for service and p is the current position of the vehicle. An

Algorithm 5: **Divide & Conquer (DC) Policy**

Assumes: An r-partition $\{\mathcal{E}_k\}_{k=1}^r$ of $\mathcal{E}$ that is simultaneously equitable with respect to f and $f^{1/2}$.

if $Q = \emptyset$ **then**
  Let $\tilde{G}_1^*$ be the point minimizing the sum of distances to demands serviced in the past (the initial condition for $\tilde{G}_1^*$ is a random point in $\mathcal{E}$); if $p \neq \tilde{G}_1^*$, move to $\tilde{G}_1^*$, otherwise stop.
else
  $k \leftarrow k_o$ (k_0 is uniformly randomly chosen in the set $\{1, \ldots, r\}$).
  repeat
    Move to the median of subregion $\mathcal{E}_k$.
    Compute a TSP tour through all demands in subregion $\mathcal{E}_k$. Service all demands by following the TSP tour, starting at a random demand on the TSP tour.
    $k \leftarrow k + 1$ modulo r.
  until $Q = \emptyset$
Repeat.

r-partition that is simultaneously equitable with respect to f and $f^{1/2}$ is indeed guaranteed to exist (see discussion on equitable partitions in section 2.3.2). The number of subregions r is a design parameter whose choice will be discussed after the analysis of the policy. One should note that $\tilde{G}_1^*$ (which is defined in Algorithm 5) is indeed the empirical median. Next, we characterize the DC policy. One can easily show that the DC policy is *unbiased*.

7.2.1 Analysis of the DC policy in light load

We first study the light-load case (i.e., $\varrho \to 0^+$).

Theorem 7.2.1 (Performance of DC policy in light load)**.** *As $\varrho \to 0^+$, for every r, the DC policy is asymptotically optimal, that is,*

$$\overline{T}_{\mathrm{DC}}(r) \to \overline{T}^*, \qquad \text{as } \varrho \to 0^+.$$

Proof. Using arguments similar to those in [73], consider generic initial conditions in $\mathcal{E}$ for the vehicle's position and for the positions of the outstanding demand (denote the initial number of demands with n_0). Under the DC policy, an upper bound to the time C_0 needed to service *all* of the initial demands and move to the empirical median $\tilde{G}_1^*$ is: $C_0 \leq (n_0 + 1)\operatorname{diam}(\mathcal{E})/v + \sum_{j=1}^{n_0} s_j$. Thus, we have $\mathbb{E}[C_0] \leq \mathbb{E}[n_0](\operatorname{diam}(\mathcal{E})/v + \bar{s}) + \operatorname{diam}(\mathcal{E})/v$.

Given the time interval $C_0 = t$, $t \in \mathbb{R}_{\geq 0}$, the probability that no *new* demand arrives during such time interval is $\mathbb{P}[N(C_0) = 0 | C_0 = t] = \exp(-\lambda t)$, where $N(t)$ is the counting

variable associated to a Poisson process with rate λ. Hence, by applying the law of total probability and Jensen's inequality for convex functions

$$\begin{aligned}\mathbb{P}\left[N(C_0)=0\right] &= \int_0^\infty \mathbb{P}\left[N(C_0)=0|\, C_0=t\right] f_{C_0}(t)\, dt \\ &= \int_0^\infty \exp(-\lambda t) f_{C_0}(t)\, dt \geq \exp\Big(-\lambda \int_0^\infty t f_{C_0}(t)\, dt\Big) \\ &= \exp\Big(-\lambda \mathbb{E}\left[C_0\right]\Big) \geq \exp\Big(-\lambda \mathbb{E}\left[n_0\right](\operatorname{diam}(\mathcal{E})/v + \bar{s}) - \lambda \operatorname{diam}(\mathcal{E})/v\Big),\end{aligned} \tag{7.1}$$

where f_{C_0} is the density function of random variable C_0. Since, by assumption, $\mathbb{E}\left[n_0\right] < \infty$ and $\varrho \to 0^+$ (i.e., $\lambda \to 0^+$), we obtain $\mathbb{P}\left[N(C_0)=0\right] \to 1^-$ as $\varrho \to 0^+$. As a consequence, after an initial transient, all demands will be generated with the vehicle at the empirical median $\tilde{G}_1^*$ and with an empty demand queue, with probability 1. Then, in light load, the DC policy becomes identical to the light-load routing policy presented in [5]; since such routing policy is optimal, in light load, for the 1-DTRP [5], we obtain the desired result (the proof in [5] basically shows that $\tilde{G}_1^* \to G_1^*(\mathcal{E})$). □

7.2.2 Analysis of the DC policy in heavy load

Next, we consider the DC policy in heavy load, i.e., when $\varrho \to 1^-$. The performance of the DC policy in heavy load is characterized by the following theorem.

Theorem 7.2.2 (Performance of DC policy in heavy load)**.** *As $\varrho \to 1^-$, the system time for the DC policy satisfies*

$$\overline{T}_{\mathrm{DC}}(r) \leq \left(1+\frac{1}{r}\right) \frac{\beta_{\mathrm{TSP},2}^2}{2} \frac{\lambda\Big[\int_{\mathcal{E}} f^{1/2}(x)\, dx\Big]^2}{v^2\,(1-\varrho)^2}, \tag{7.2}$$

where r is the number of subregions.

Contrary to the UTSP policy (see section 2.5.3 for its description), the DC policy does not define a $\sum GI/G/1$ queue (a queue is denoted $\sum GI/G/1$ if its input process is the superposition of r independent renewal processes). Therefore, to prove theorem 7.2.2 we use non-standard techniques. The proof of theorem 7.2.2 builds on a number of intermediate results; we start with the following lemma, similar to Lemma 1 in [73] and lemma 4.3.1, characterizing the number of outstanding demands in heavy load.

Lemma 7.2.3 (Number of outstanding demands in heavy load for DC policy)**.** *In heavy load (i.e., $\varrho \to 1^-$), after a transient, the number of demands serviced in a single tour of the vehicle in the DC policy is very large with high probability (i.e., the number of demands tends to $+\infty$ with probability that tends to 1, as ϱ approaches 1^-).*

Proof. Consider the case where the vehicle moves with infinite velocity (i.e., $v \to +\infty$); then the system is reduced to an M/G/1 queue (i.e., a queue with exponentially distributed inter-arrival times, generally distributed service times, and a single server). The infinite-velocity system has fewer demands waiting in queue. A known result on M/G/1 queues [106] states that, after transients, the total number of demands, as $\varrho \to 1^-$, is very large with high probability. Due to the way $\mathcal{E}$ is partitioned, the number of outstanding demands in each subregion is also very large with high probability. In particular, after transients,

the number of demands is very large with high probability at the time instants when the vehicle starts a new TSP tour. □

Lemma 7.2.3 has two implications. First, since the number of demands is very large at the time instants when the vehicle starts a new TSP tour, we can apply equation (2.7) to estimate the average length of a TSP tour. Second, since $Q \neq \emptyset$ with high probability, the DC policy, in heavy load, *reduces* to lines 5-9 of Algorithm 5 with the condition $Q = \emptyset$ *always* false.

We refer to the time instant t_i, $i \geq 0$, in which the vehicle starts a new TSP tour in subregion 1 as the epoch i of the policy; we refer to the time interval between epoch i and epoch $i+1$ as the ith iteration. Let n_i^k, $k \in \{1, \ldots, r\}$, be the number of outstanding demands serviced in subregion k during iteration i. Finally, let C_i^k be the time interval between the time instant the vehicle starts to service demands in subregion k during iteration i and the time instant the vehicle starts to service demands in the same subregion k during next iteration $i+1$. Demands arrive in subregion $\mathcal{E}_k$ according to a Poisson process with rate $\hat{\lambda} \doteq \lambda / r$, where we use the fact that the partition $\{\mathcal{E}_k\}_{k=1}^r$ is equitable with respect to f; then, we have $\mathbb{E}\left[n_{i+1}^k\right] = \hat{\lambda}\, \mathbb{E}\left[C_i^k\right]$. The time interval C_i^k is the sum of three components:

1. the time for the r travels from the end of one TSP tour to the start of next one;
2. the time to perform r TSP tours;
3. the time to provide on-site service while performing the r TSP tours.

The first component is trivially upper bounded by $2\,\mathrm{diam}(\mathcal{E})/v \cdot r$. Given that a demand falls in subregion $\mathcal{E}_k$, the conditional density for its location (whose support is $\mathcal{E}_k$) is $f(x) / \left(\int_{\mathcal{E}_k} f(x)\, dx \right)$. By equation (2.7), we have that the expected length of a TSP tour though n demands distributed according to the density $f(x) / \left(\int_{\mathcal{E}_k} f(x)\, dx \right)$ satisfies (with a slight abuse of notation, we call such length $\mathrm{TSP}(n)$)

$$\lim_{n \to +\infty} \frac{\mathbb{E}\left[\mathrm{TSP}(n)\right]}{\sqrt{n}} = \beta_{\mathrm{TSP}} \left(\int_{\mathcal{E}_k} \sqrt{\frac{f(x)}{\int_{\mathcal{E}_k} f(x)\, dx}}\, dx \right) = \beta_{\mathrm{TSP}} \sqrt{r}\, \frac{\int_{\mathcal{E}} f^{1/2}(x)\, dx}{r} \doteq \beta, \tag{7.3}$$

where we have exploited the fact that, by definition of the DC policy, the partition $\{\mathcal{E}_k\}_{k=1}^r$ is simultaneously equitable with respect to f and $f^{1/2}$. Pick now an arbitrarily small $\zeta > 0$. By equation (7.3), there exists an $\tilde{n} \in \mathbb{N}$ such that for all $n > \tilde{n}$ it holds $\mathbb{E}\left[\mathrm{TSP}(n)\right] \leq (\beta + \zeta)\sqrt{n}$. Then, the expected length of a TSP tour through n_i^k demands (with a slight abuse of notation, we call such length $\mathrm{TSP}(n_i^k)$) can be upper bounded as

$$\begin{aligned}
\mathbb{E}\left[\mathrm{TSP}(n_i^k)\right] &= \sum_{n=0}^{+\infty} \mathbb{E}\left[\mathrm{TSP}(n_i^k) \,|\, n_i^k = n\right] \mathbb{P}\left[n_i^k = n\right] \\
&\leq \sum_{n=\tilde{n}+1}^{+\infty} (\beta + \zeta)\sqrt{n}\, \mathbb{P}\left[n_i^k = n\right] + \sum_{n=0}^{\tilde{n}} \tilde{n}\ \mathrm{diam}(\mathcal{E})\, \mathbb{P}\left[n_i^k = n\right] \\
&\leq (\beta + \zeta)\, \mathbb{E}\left[\sqrt{n_i^k}\right] + \tilde{n}\ \mathrm{diam}(\mathcal{E})\, \mathbb{P}\left[n_i^k \leq \tilde{n}\right] \\
&\leq (\beta + \zeta) \sqrt{\mathbb{E}\left[n_i^k\right]} + \tilde{n}\ \mathrm{diam}(\mathcal{E})\, \mathbb{P}\left[n_i^k \leq \tilde{n}\right],
\end{aligned} \tag{7.4}$$

where in the first equality we have used the law of total expectation, and in the last inequality we have applied Jensen's inequality for concave functions in the form $\mathbb{E}\left[\sqrt{X}\right] \leq \sqrt{\mathbb{E}[X]}$. By lemma 7.2.3, there exists $\bar{\varrho} \in (0, 1)$ such that for all $\varrho \in [\bar{\varrho}, 1)$ there exists an integer $\bar{i}(\varrho)$ (possibly dependent on ϱ) such that $\mathbb{P}\left[n_i^k \leq \tilde{n}\right] < 1/\tilde{n}$. Consider any $\varrho \in [\bar{\varrho}, 1)$ and assume that $i \geq \bar{i}(\varrho)$, then

$$\mathbb{E}\left[\mathrm{TSP}(n_i^k)\right] \leq (\beta + \zeta)\sqrt{\mathbb{E}\left[n_i^k\right]} + \mathrm{diam}(\mathcal{E}), \quad \text{for } i \geq \bar{i}(\varrho) \text{ and } \varrho \in [\bar{\varrho}, 1).$$

Finally, since on-site service times are independent of the number of outstanding demands, the expected on-site service requirement for n_i^k demands is $\bar{s}\,\mathbb{E}\left[n_i^k\right]$.

Then, we obtain the following recurrence relation (where we define $\bar{n}_i^k \doteq \mathbb{E}\left[n_i^k\right]$):

$$\begin{aligned}\bar{n}_{i+1}^k &= \hat{\lambda}\mathbb{E}\left[C_i^k\right] \\ &\leq \hat{\lambda}\Big(\mathrm{diam}(\mathcal{E})\frac{3\,r}{v} + \frac{\beta+\zeta}{v}\sum_{j=k}^{r}\sqrt{\bar{n}_i^j} + \frac{\beta+\zeta}{v}\sum_{j=1}^{k-1}\sqrt{\bar{n}_{i+1}^j} + \bar{s}\sum_{j=k}^{r}\bar{n}_i^j + \bar{s}\sum_{j=1}^{k-1}\bar{n}_{i+1}^j\Big),\end{aligned} \tag{7.5}$$

where $k \in \{1, \ldots, r\}$, $i \geq \bar{i}(\varrho)$, and $\varrho \in [\bar{\varrho}, 1)$.

The r inequalities above describe a system of recurrence relations that allows us to find an upper bound on $\limsup_{i\to+\infty} \bar{n}_i^k$. The following lemma bounds the values to which the limits $\limsup_{i\to+\infty} \bar{n}_i^k$ converge.

Lemma 7.2.4 (Steady state number of demands for DC policy)**.** *In heavy load, for every set of initial conditions $\{\bar{n}_1^k\}_{k\in\{1,\ldots,r\}}$, the trajectories $i \mapsto \bar{n}_i^k$, $k \in \{1, \ldots, r\}$, satisfy*

$$\bar{n}^k \doteq \limsup_{i\to+\infty} \bar{n}_i^k \leq \beta_{\mathrm{TSP}}^2 \frac{\lambda^2\left[\int_{\mathcal{E}} f^{1/2}(x)\,dx\right]^2}{r\,v^2\,(1-\varrho)^2}.$$

Proof. Define the *constants*: $\tilde{\beta} \doteq \beta + \zeta$ and $a \doteq \mathrm{diam}(\mathcal{E})\,3\,r/v$. Henceforth we assume that $\varrho \in [\bar{\varrho}, 1)$ and that $i \geq \bar{i}(\varrho)$, and we simply denote $\bar{i}(\varrho)$ as $\bar{i}$. Next we define two auxiliary systems, System-Y and System-Z, whose trajectories will be used to bound the trajectories $i \mapsto \bar{n}_i^k$. We define System-Y (with state $y(i) \in \mathbb{R}^r$) as

$$\begin{aligned}&y_k(i+1) = \hat{\lambda}\,a + \hat{\lambda}\left(\sum_{j=k}^{r}\Big(\bar{s}\,y_j(i) + \frac{\tilde{\beta}}{v}\sqrt{y_j(i)}\Big) + \sum_{j=1}^{k-1}\Big(\bar{s}y_j(i+1) + \frac{\tilde{\beta}}{v}\sqrt{y_j(i+1)}\Big)\right), \text{ for } i \geq \bar{i}, \\ &y_k(\bar{i}) = \bar{n}_{\bar{i}}^k,\end{aligned} \tag{7.6}$$

where $k \in \{1, \ldots, r\}$. System-Y is obtained by replacing the inequality in equation (7.5) with an equality. Pick, now, any $\varepsilon > 0$. From Young's inequality

$$\sqrt{w} \leq \frac{1}{4\varepsilon} + \varepsilon w, \quad \text{for all } w \in \mathbb{R}_{\geq 0}.$$

By applying Young's inequality in equation (7.6) we obtain

$$y_k(i+1) \leq \hat{\lambda}\left(\bar{s}+\varepsilon\frac{\tilde{\beta}}{v}\right)\left(\sum_{j=k}^{r} y_j(i)+\sum_{j=1}^{k-1} y_j(i+1)\right)+\hat{\lambda}\left(\frac{r\,\tilde{\beta}}{4v\varepsilon}+a\right), \qquad k\in\{1,\ldots,r\}. \tag{7.7}$$

Next, define System-Z (with state $z(i)\in\mathbb{R}^r$) as

$$\begin{aligned} z_k(i+1) &= \hat{\lambda}\left(\bar{s}+\varepsilon\frac{\tilde{\beta}}{v}\right)\left(\sum_{j=k}^{r} z_j(i)+\sum_{j=1}^{k-1} z_j(i+1)\right)+\hat{\lambda}\left(\frac{r\,\tilde{\beta}}{4v\varepsilon}+a\right), \qquad \text{for } i\geq\bar{i}, \\ z_k(\bar{i}) &= \bar{n}_{\bar{i}}^{k}, \end{aligned} \tag{7.8}$$

where $k\in\{1,\ldots,r\}$. The proof now proceeds as follows. First, we show that the condition $\bar{n}_{\bar{i}}^{k}=y_k(\bar{i})=z_k(\bar{i})$ for all k implies that

$$\bar{n}_i^k \leq y_k(i)\leq z_k(i), \quad \text{for all } k \text{ and for all } i\geq\bar{i}. \tag{7.9}$$

Second, we show that the trajectories of System-Z are bounded; this fact, together with equation (7.9), implies that also trajectories of variables $\bar{n}_i^k$ and $y_k(i)$ are bounded. Third, and last, we will compute $\limsup_{i\to+\infty} y_k(i)$; this quantity, together with equation (7.9), will yield the desired result.

We prove the first fact by induction on i, i.e., we prove that if $\bar{n}_i^k\leq y_k(i)\leq z_k(i)$ for all $k\in\{1,\ldots,r\}$, then $\bar{n}_{i+1}^k\leq y_k(i+1)\leq z_k(i+1)$ for all $k\in\{1,\ldots,r\}$, where $i\geq\bar{i}$. Indeed, it is immediate to show that, under the assumptions, it holds: $\bar{n}_{i+1}^1\leq y_1(i+1)\leq z_1(i+1)$; moreover, it is easy to show that, under the assumptions, and if $\bar{n}_{i+1}^j\leq y_j(i+1)\leq z_j(i+1)$ for all $j\leq k<r$, then $\bar{n}_{i+1}^{k+1}\leq y_{k+1}(i+1)\leq z_{k+1}(i+1)$. The inductive step then follows immediately.

We now turn our attention to the second issue, namely boundedness of trajectories for System-Z (described in equation (7.8)). Define $\delta\doteq\hat{\lambda}(\bar{s}+\varepsilon\,\tilde{\beta}/v)$. System-Z is a discrete-time system and can be rewritten (by adding and subtracting $z_{k-1}(i)$ in the second term between parentheses in equation (7.8), when $k>1$) as

$$z_k(i+1)=\begin{cases} \delta\sum_{j=1}^{r} z_j(i)+\hat{\lambda}\left(\frac{r\,\tilde{\beta}}{4v\varepsilon}+a\right), & \text{for } k=1, \\ \Big(1+\delta\Big)\,z_{k-1}(i+1)-\delta\,z_{k-1}(i), & \text{for } k\in\{2,\ldots,r\}. \end{cases}$$

By using the above equation and by simple induction arguments, it is possible to show that System-Z can be written in canonical form as

$$z_k(i+1)=(1+\delta)^{k-1}\left(\delta\sum_{j=1}^{r} z_j(i)+\hat{\lambda}\left(\frac{r\,\tilde{\beta}}{4v\varepsilon}+a\right)\right)-\sum_{j=1}^{k-1}\delta(1+\delta)^{k-j-1}z_j(i), \qquad k\in\{1,\ldots,r\}.$$

Hence, System-Z can be written in compact form as

$$z(i+1)=\Big(A+B(\varepsilon)\Big)z(i)+c(\varepsilon),$$

where A is a matrix in $\mathbb{R}^{r\times r}$ and is independent of ε, $B(\varepsilon)\in\mathbb{R}^{r\times r}$ and each element of $B(\varepsilon)$, say $b(\varepsilon)_{kj}$, has the property $\lim_{\varepsilon\to0^+} b(\varepsilon)_{kj}=0$ for all $k,j\in\{1,\ldots,r\}$, and $c(\varepsilon)\in\mathbb{R}^{r\times1}$

(notice that $c(\varepsilon)$ is well defined for every $\varepsilon > 0$). It is easy to see that the entries of matrix A are

$$a_{kj} = \begin{cases} \hat{\varrho}\Big[(1+\hat{\varrho})^{k-1} - (1+\hat{\varrho})^{k-j-1}\Big], & \text{for } 1 \le j \le k-1, \\ \hat{\varrho}(1+\hat{\varrho})^{k-1}, & \text{otherwise,} \end{cases}$$

where $\hat{\varrho} \doteq \hat{\lambda}\bar{s}$. Notice that, since $\hat{\varrho} > 0$, we have $|a_{kj}| = a_{kj}$ for all k and j. Then, for each $k \in \{1, \dots, r\}$, we have

$$\sum_{j=1}^{r} |a_{kj}| = \sum_{j=1}^{r} a_{kj} = r\,\hat{\varrho}\,(1+\hat{\varrho})^{k-1} - \hat{\varrho} \sum_{j=1}^{k-1} (1+\hat{\varrho})^{k-j-1} = (1+\hat{\varrho})^{k-1} \underbrace{(r\hat{\varrho} - 1)}_{<0} + 1 < 1,$$

since $\hat{\varrho} = \lambda\bar{s}/r$ and $\varrho = \lambda\bar{s} < 1$ by assumption. Therefore, the L^∞-induced norm of matrix A satisfies: $\|A\|_\infty = \max_k \sum_{j=1}^{r} |a_{kj}| < 1$. Since any induced norm $\|\cdot\|$ satisfies the inequality $\rho(A) \le \|A\|$, where $\rho(A)$ is the spectral radius of A, we conclude that $A \in \mathbb{R}^{r\times r}$ has eigenvalues strictly inside the unit disk (i.e., A is a stable matrix). Since the eigenvalues of a matrix depend continuously on the matrix entries, there exists a sufficiently small $\varepsilon > 0$ such that the matrix $A + B(\varepsilon)$ has eigenvalues strictly inside the unit disk. Accordingly, having selected a sufficiently small ε, the solution $i \mapsto z(i) \in \mathbb{R}^r_{\ge 0}$ of System-Z converges exponentially fast to the unique equilibrium point

$$z^*(\varepsilon) = \Big(I_r - A - B(\varepsilon)\Big)^{-1} c(\varepsilon), \tag{7.10}$$

where I_r is the identity matrix of size r. Combining equation (7.9) with the previous statement, we see that the solutions $i \mapsto \bar{n}(i)$ (where $\bar{n}(i) \doteq (\bar{n}_i^1, \dots, \bar{n}_i^r)$) and $i \mapsto y(i)$ are bounded. Thus

$$\limsup_{i\to+\infty} \bar{n}(i) \le \limsup_{i\to+\infty} y(i) < +\infty. \tag{7.11}$$

Finally, we turn our attention to the third issue, namely the computation of $y \doteq \limsup_{i\to+\infty} y(i)$. Taking the lim sup of the left- and right-hand sides of equation (7.6), and noting that

$$\limsup_{i\to+\infty} \sqrt{y_j(i)} = \sqrt{\limsup_{i\to+\infty} y_j(i)}, \quad \text{for } j \in \{1, \dots, r\},$$

since $\sqrt{\cdot}$ is continuous and strictly monotone increasing on $\mathbb{R}_{>0}$, we obtain that

$$y_k = \hat{\lambda}\,a + \hat{\lambda} \sum_{j=1}^{r} \Big(\bar{s}\,y_j + \frac{\tilde{\beta}}{v}\sqrt{y_j}\Big); \tag{7.12}$$

therefore we have $y_k = y_j$ for all $k, j \in \{1, \dots, r\}$. Substituting in equation (7.12) and solving for y_k we obtain

$$y_k = \frac{1}{4}\left(\frac{\lambda\tilde{\beta}}{v\,(1-\varrho)} + \sqrt{\frac{\lambda^2\tilde{\beta}^2}{v^2\,(1-\varrho)^2} + \frac{4\hat{\lambda}a}{1-\varrho}}\right)^2.$$

Recalling that this analysis holds for every $\varrho \in [\bar{\varrho}, 1)$, by taking ϱ arbitrarily close to 1 we

obtain

$$y_k = \frac{\lambda^2 \tilde{\beta}^2}{v^2 (1-\varrho)^2} \quad \text{as } \varrho \to 1^-. \tag{7.13}$$

(Formally, we should have written $\lim_{\varrho \to 1^-} y_k (1-\varrho)^2 = \lambda^2 \tilde{\beta}^2 / v^2$, however, as discussed in Remark 2.5.1, we prefer to adhere to the less formal but more transparent style of equation (7.13)).

Noting that from equation (7.11) $\limsup_{i \to +\infty} \bar{n}_i^k \leq y_k$, and recalling that

$$\tilde{\beta} = \beta_{\text{TSP}} \int_{\mathcal{E}} f^{1/2}(x)\, dx / \sqrt{r} + \zeta$$

where ζ is an arbitrarily small constant, we obtain the desired result. □

We are now in a position to prove theorem 7.2.2

Proof of theorem 7.2.2. Define $\bar{C}^k \doteq \limsup_{i \to \infty} \mathbb{E}\left[C_i^k\right]$; then we have, by using the upper bound on $\mathbb{E}\left[C_i^k\right]$ in equation (7.5) and neglecting constant terms,

$$\bar{C}^k \doteq \limsup_{i \to \infty} \mathbb{E}\left[C_i^k\right] \leq \frac{\tilde{\beta}}{v} \sum_{j=1}^{r} \sqrt{\bar{n}^j} + \bar{s} \sum_{j=1}^{r} \bar{n}^j \leq \frac{r\, \lambda\, \tilde{\beta}\, \beta}{v^2 (1-\varrho)} + \frac{r\, \varrho\, \lambda\, \beta^2}{v^2 (1-\varrho)^2}.$$

Hence, in the limit $\varrho \to 1^-$, we have

$$\bar{C}^k \leq \frac{r\, \lambda\, \beta^2}{v^2 (1-\varrho)^2} = \frac{\lambda \beta_{\text{TSP}}^2 \left[\int_{\mathcal{E}} f^{1/2}(x)\, dx\right]^2}{v^2 (1-\varrho)^2}.$$

Consider a random demand that arrives in subregion k. Its expected steady-state system time, $\overline{T}^k$, will be upper bounded, as $\varrho \to 1^-$, by

$$\overline{T}^k \leq \frac{1}{2} \bar{C}^k + \frac{1}{2} \bar{s}\, \bar{n}^k \leq \frac{1}{2} \bar{C}^k + \frac{1}{2} \frac{\lambda \beta^2}{v^2 (1-\varrho)^2},$$

where we used the fact that, as $\varrho \to 1^-$, the travel time along a TSP tour is negligible compared to the on-site service time requirement. Unconditioning, we obtain the claim

$$\overline{T}_{\text{DC}}(r) \leq \left(1 + \frac{1}{r}\right) \frac{\lambda \beta_{\text{TSP}}^2 \left[\int_{\mathcal{E}} f^{1/2}(x)\, dx\right]^2}{2 v^2 (1-\varrho)^2}.$$

□

7.2.3 Discussion

The DC policy is optimal in light load (theorem 7.2.1); moreover, if we let $r \to \infty$, the DC policy achieves the heavy-load lower bound (2.11) for unbiased policies (theorem 7.2.2). Therefore the DC policy is *both* optimal in light load *and* arbitrarily close to optimality in heavy load, and stabilizes the system in every load condition (since it stabilizes the system in the limit $\varrho \to 1^-$). Notice that with $r = 10$ the DC policy is already guaranteed to be within 10% of the optimal (for unbiased policies) performance in heavy load.

If one chooses $r > 1$, the computation of a simultaneously equitable r-partition of $\mathcal{E}$ requires the knowledge of f. Hence, if $r \to +\infty$, the policy is (i) provably optimal in both light and heavy load, and (ii) adaptive with respect to arrival rate λ, expected on-site service $\bar{s}$, and vehicle's velocity v. If, instead, $r = 1$, the policy is (i) provably optimal in light load and within a factor 2 of the optimal performance in heavy load, and (ii) adaptive with respect to arrival rate λ, expected on-site service $\bar{s}$, vehicle's velocity v, and spatial density f; in other words, when $r = 1$, the DC policy adapts to *all* problem data. (These results should be compared with the discussion in section 2.5.3 about the non-adaptive nature of the UTSP policy, which, for example, might become unstable if λ increases.)

An approximate algorithm to compute, with *arbitrarily small* error, an r-partition that is simultaneously equitable with respect to two measures can be found in [17] (specifically, see discussion on page 621); in the particular case when the density f is uniform, the problem of finding an r-partition that is simultaneously equitable with respect to f and $f^{1/2}$ becomes trivial. Given a simultaneously equitable r-partition of $\mathcal{E}$, the subregions can be indexed using the following procedure: (i) a TSP tour through the medians of the subregions is computed, (ii) subregions are indexed according to the order induced by this TSP tour. In practice, the DC policy would be implemented by allowing the vehicle to skip subregions where no demand is present (the "shortcuts" were not included in the presentation of the DC policy to make its analysis easier).

Note that if we assume that f is known (as it must be the case for the implementation of the DC policy with $r > 1$), then instead of using the empirical median $\tilde{G}_1^*$ one should directly use the exact median $G_1^*(\mathcal{E})$ (which is the solution of a strictly convex problem).

The DC policy requires on-line solutions of possibly large TSPs and, therefore, practical implementations of this policy should rely on heuristics for TSPs, such as Lin-Kernighan's or Christofides' (see section 2.4.3). We will further discuss computation times in section 7.6.

Finally, the DC policy with $r = 1$ is similar to the generation policy presented in [73]; in particular, the generation policy has the same performance guarantees of the DC policy with $r = 1$. However, the generation policy is analyzed only for the case of *uniform* spatial density f, and its implementation *does* require the knowledge of f (while, as discussed before, the DC policy with $r = 1$ adapts to all problem data, including f). The part of Algorithm 5 in lines 5-9 is similar to the PART-TSP policy presented in [108]; however, the PART-TSP policy is only *informally* analyzed by assuming steady-state conditions, in particular no proof of stability and convergence is provided.

In the next section we present and analyze another single-vehicle policy, the Receding Horizon (RH) policy, that applies ideas of receding-horizon control to dynamic vehicle routing problems.

7.3 The Single-Vehicle Receding Horizon Policy

We describe the Receding Horizon (RH) policy in Algorithm 6, where Q is the set of outstanding demands waiting for service and p is the current position of the vehicle; moreover, given a tour T of Q, a fragment of T is a connected subgraph of T. If $Q \neq \emptyset$, the vehicle selects a random η-fragment of the TSP tour through the demands in Q (i.e., a fragment of length η TSP(Q) of such tour); note that the horizon is not fixed, but it is adjusted according to the cost of servicing the outstanding demands. In general, the performance of the system will depend on the choice of the parameter η, which will be discussed after the

Algorithm 6: **Receding Horizon (RH) Policy**

input: Scalar $\eta \in (0,\, 1)$

1 **if** $Q = \emptyset$ **then**

2 Let $\tilde{G}_1^*$ be the point minimizing the sum of distances to demands serviced in the past (the initial condition for $\tilde{G}_1^*$ is a random point in $\mathcal{E}$); if $p \neq \tilde{G}_1^*$, move to $\tilde{G}_1^*$, otherwise stop.

3 **else**

4 **repeat**

5 Compute the TSP tour through all demands in Q.

6 Uniformly randomly choose, independently from the past, a fragment of length η TSP(Q) of such TSP tour.

7 Service the selected fragment starting from the endpoint closest to the current position.

8 **until** $Q = \emptyset$

9 Repeat.

analysis of the policy. One can easily show that the RH policy is *unbiased*.

7.3.1 Stability and performance of the RH policy

The RH policy is attractive since it can be implemented without any knowledge of the underlying demand generation process (in particular, without any knowledge of f). On the other hand, an important caveat of receding horizon control is that closed-loop stability is not guaranteed in general. In this section, we study the stability of the RH policy and we discuss its performance.

Stability of the RH policy

In general, the stability properties of receding horizon controllers deteriorate as the horizon becomes shorter; remarkably, the RH policy is stable in every load for every $0 < \eta < 1$.

Theorem 7.3.1. *As $\varrho \to 1^-$, the system time for the RH policy satisfies*

$$\overline{T}_{\mathrm{RH}}(\eta) \leq \frac{\lambda \beta_{\mathcal{E},2}^2 |\mathcal{E}|}{v^2\,(1-\varrho)^2}, \qquad \textit{for all } \eta \in (0,\, 1),$$

where $|\mathcal{E}|$ is the area of $\mathcal{E}$ and $\beta_{\mathcal{E},2}$ is the constant appearing in equation (2.8).

Proof. By following the same arguments as in lemma 7.2.3, it is easy to show that under the RH policy, after a transient, the number of demands at the instants when a new TSP tour is computed is very large with high probability. Then, in heavy load, we have $Q \neq \emptyset$ with high probability, and the RH policy *reduces* to lines 4-8 with the condition $Q = \emptyset$ *always* false.

We refer to the time instant t_i, $i \geq 0$, in which the vehicle computes a new TSP tour as the epoch i of the policy; we refer to the time interval between epoch i and epoch $i+1$ as the ith iteration and we refer to its length as the cost C_i. Let n_i be the number of outstanding demands at epoch i; with a slight abuse of notation, we denote by TSP(n_i) the length of the TSP tour through the outstanding demands at epoch i.

Since an η-fragment is randomly chosen, the expected number of demands left unserviced during iteration i is $(1-\eta)\,\mathbb{E}\,[n_i]$; then,

$$\mathbb{E}\,[n_{i+1}] = \underbrace{\lambda\,\mathbb{E}\,[C_i]}_{\text{newly arrived demands}} + \underbrace{(1-\eta)\,\mathbb{E}\,[n_i]}_{\text{demands left unserviced during iteration } i}.$$

The expected number of demands that receive service during iteration i is $\eta\,\mathbb{E}\,[n_i]$. Moreover, the time to reach the starting endpoint of the selected fragment is bounded by $\mathrm{diam}(\mathcal{E})/v$. Then, the expected time length of iteration i can be upper bounded as

$$\mathbb{E}\,[C_i] \leq \mathrm{diam}(\mathcal{E})/v + \frac{\eta}{v}\,\mathbb{E}\,[\mathrm{TSP}(n_i)] + \bar{s}\,\eta\mathbb{E}\,[n_i] \leq \mathrm{diam}(\mathcal{E})/v + \frac{\eta}{v}\,\beta_{\mathcal{E},2}\sqrt{|\mathcal{E}|}\,\sqrt{\mathbb{E}\,[n_i]} + \bar{s}\,\eta\mathbb{E}\,[n_i], \tag{7.14}$$

where we have applied the deterministic bound in equation (2.8) and Jensen's inequality for concave functions. Therefore, we obtain

$$\mathbb{E}\,[n_{i+1}] \leq \lambda\,\mathrm{diam}(\mathcal{E})/v + \lambda\,\frac{\eta}{v}\,\beta_{\mathcal{E},2}\sqrt{|\mathcal{E}|}\,\sqrt{\mathbb{E}\,[n_i]} + \varrho\,\eta\,\mathbb{E}\,[n_i] + (1-\eta)\,\mathbb{E}\,[n_i].$$

By using the same techniques as in the proof of lemma 7.2.4 (in particular, the eigenvalue of the "bounding" discrete-time linear system is $\mu = 1-\eta\,(1-\varrho)$, whose magnitude is strictly less than one for all $\eta \in (0,\,1)$), it is possible to show that

$$\bar{n} \doteq \limsup_{i\to\infty}\,\mathbb{E}\,[n_i] \leq \frac{1}{4}\left(\frac{\lambda\,\beta_{\mathcal{E},2}\,\sqrt{|\mathcal{E}|}}{v\,(1-\varrho)} + \sqrt{\frac{\lambda^2\beta_{\mathcal{E},2}^2|\mathcal{E}|}{v^2\,(1-\varrho)^2} + \frac{4\,\lambda\,\mathrm{diam}(\mathcal{E})}{v\,\eta\,(1-\varrho)}}\right)^2, \quad \text{for all} \quad \eta \in (0,\,1).$$

Define $\bar{C} \doteq \limsup_{i\to+\infty}\mathbb{E}\,[C_i]$. Consider a random demand in steady state; the expected number of iterations before such demand is scheduled for service is given by $\sum_{p=0}^{+\infty}\,p\,\eta(1-\eta)^p$. Hence, the expected steady-state system time of such demand, $\overline{T}$, will be upper bounded for all $\eta \in (0,\,1)$ by

$$\overline{T} \leq \frac{1}{2}\bar{C} + \frac{1}{2}\,\bar{s}\,\eta\,\bar{n} + \bar{C}\sum_{p=0}^{+\infty}\,p\,\eta(1-\eta)^p \leq \frac{\lambda\beta_{\mathcal{E},2}^2|\mathcal{E}|}{v^2\,(1-\varrho)^2}, \qquad \text{as } \varrho\to 1^-,$$

where we used the fact that, as $\varrho \to 1^-$, the travel time along an η-fragment is negligible compared to the on-site service time requirement. This concludes the proof.

□

Since, by theorem 7.3.1, the RH policy stabilizes the system in the limit $\varrho \to 1^-$, it also stabilizes the system for *every* load factor ϱ.

Performance of the RH policy

In light load, the RH policy becomes identical to the DC policy, and therefore it is *optimal* (see theorem 7.2.1). For the case $\varrho \to 1^-$, because of the dependencies that arise among demands' locations, we were unable to obtain rigorous upper bounds on the system time that are well matched by numerical experiments. However, we now present an heuristic analysis of the RH policy that provides interesting insights on its behavior and suggests bounds on its heavy-load system time that are well matched by simulation results.

The service of an η-fragment introduces *dependencies* among the locations of the demands that are left unserviced; therefore, even though the number of demands, as $\varrho \to 1^-$, becomes large with probability 1, the result in equation (2.7) (that requires a set of *independently* distributed demands) formally does *not* hold. However, due to the randomized selection of the η-fragments, it is quite natural to conjecture that equation (2.7) still provides a good estimate on the average lengths of the TSP tours that are computed. The rationale behind this conjecture is the following: When η is close to 1, most of the outstanding demands are serviced during one iteration, and the next TSP tour is mostly through newly arrived demands, that are independent by the assumptions on the demand generation process; when, instead, η is close to zero, the randomized selection of *short* fragments only introduce "negligible" correlations.

The above observations motivate us to use $\beta_{\text{TSP}} \int_{\mathcal{E}} f^{1/2}(x)\,dx$ (as it would be dictated by equation (2.7)) in equation (7.14), instead of $\beta_{\mathcal{E},2}\sqrt{|\mathcal{E}|}$ (as it is dictated by equation (2.8)). Then, by repeating the same steps as in the proof of theorem 7.2.2, lemma 7.2.4, and theorem 7.3.1, we would obtain the following result for all $\eta \in (0,\, 1)$

$$\overline{T}_{\text{RH}}(\eta) \leq \frac{\lambda \beta_{\text{TSP}}^2 \Big[\int_{\mathcal{E}} f^{1/2}(x)\,dx\Big]^2}{v^2\,(1-\varrho)^2}, \qquad \text{as } \varrho \to 1^-. \tag{7.15}$$

The upper bound in equation (7.15) is generally tighter[1] than the upper bound in theorem 7.3.1, but unlike the upper bound in theorem 7.3.1 it has not been established formally. Simulation results (see section 7.6) indeed confirm the upper bound in equation (7.15).

Finally, note that when η is close to zero, the RH policy is conceptually similar to the DC policy with $r \to \infty$, so we also speculate that in heavy load the performance of the RH policy improves as the horizon becomes shorter. Simulation results (see section 7.6) indeed confirm this behavior.

7.3.2 Discussion

The RH policy is optimal in light load and stabilizes the system in every load condition (these two statements have been established rigorously). The implementation of the RH policy does *not* require the knowledge of $\lambda, \bar{s}, v$ and f. Therefore, the RH policy *adapts* to arrival rate λ, expected on-site service $\bar{s}$, vehicle's velocity v, and spatial density f; in other words, the RH policy adapts to *all* problem data.

In section 7.2 we saw that also the DC policy with $r = 1$ adapts to all problem data. Simulation results (see section 7.6) show that in heavy load the RH policy with a short horizon η (say, $\eta \approx 0.2$) performs better than the DC policy with $r = 1$ (the light-load behavior is clearly the same). Therefore, to date, the best available policy for the 1-DTRP that is unbiased and adapts to all problem data seems to be the RH policy with $\eta \approx 0.2$.

It is natural to wonder how the RH policy performs if the η-fragment is not selected randomly, but it is instead selected according to the rule: Find the fragment of length $\eta\,\text{TSP}(Q)$ that maximizes the number of reached demands. In other words, the vehicle now looks for a maximum-reward η-fragment. Clearly, theorem 7.3.1 still applies. Simulation results (see [79]) show that this spatially-biased variant of the RH policy appears to perform better than an optimal unbiased policy but not as well as an optimal biased policy.

[1]Notice that by Jensen's inequality for concave functions $\int_{\mathcal{E}} f^{1/2}(x)\,dx \leq \sqrt{|\mathcal{E}|}\Big[\int_{\mathcal{E}} f(x)\,dx\Big]^{1/2} = \sqrt{|\mathcal{E}|}$

As for the DC policy, the RH policy requires on-line solutions of possibly large TSPs and, therefore, practical implementations of this policy should also rely on heuristics for TSPs, such as Lin-Kernighan's or Christofides'. We will further discuss computation times in section 7.6.

7.4 Adaptive and Distributed Policies for the m-DTRP

In this section we extend the previous single-vehicle policies to the multi-vehicle case. First, we prove that in heavy load, given a single-vehicle, optimal unbiased policy π^* and m vehicles, there exists an unbiased π^*-partitioning policy that is optimal. This result and the optimality of the SQM policy (which is a partitioning policy that is optimal in light load) lead us to propose, for the multi-vehicle case, *partitioning policies* that use the DC policy or the RH policy as single-vehicle policies. By combining the spatially-distributed partitioning algorithms devised in chapter 6 with the results of this chapter we readily obtain adaptive routing policies amenable to distributed implementation.

7.4.1 Optimality of partitioning policies in heavy load

The following theorem shows the optimality of a specific type of partitioning policy.

Theorem 7.4.1. *As $\varrho \to 1^-$, given a single-vehicle, optimal unbiased policy π^* and m vehicles, a π^*-partitioning policy using an m-partition which is simultaneously equitable with respect to f and $f^{1/2}$ is an optimal unbiased policy for the m-DTRP.*

Proof. Let π^* be an optimal unbiased policy (e.g., the DC policy with $r \to \infty$). Construct an m-partition $\{\mathcal{E}_k\}_{k=1}^m$ of $\mathcal{E}$ that is simultaneously equitable with respect to f and $f^{1/2}$ (such partition is guaranteed to exist as discussed in section 2.3.2); assign one vehicle to each subregion. Each vehicle executes the single-vehicle unbiased policy π^* to service demands that fall within its own subregion. The probability that a demand falls in subregion $\mathcal{E}_k$ is equal to $\int_{\mathcal{E}_k} f(x)\,dx = 1/m$. Notice that the arrival rate λ_k in subregion $\mathcal{E}_k$ is reduced by a factor $\int_{\mathcal{E}_k} f(x)\,dx$, i.e., $\lambda_k = \lambda \int_{\mathcal{E}_k} f(x)\,dx = \lambda/m$; therefore, the load factor in subregion $\mathcal{E}_k$ (where only one vehicle provides service) is $\varrho_k = \lambda_k \bar{s} = \lambda \bar{s}/m = \varrho < 1$, in particular the necessary condition for stability *is satisfied* in every subregion. Finally, given that a demand falls in subregion $\mathcal{E}_k$, the conditional density for its location (whose support is $\mathcal{E}_k$) is $f(x)/\left(\int_{\mathcal{E}_k} f(x)\,dx\right) = m\,f(x)$. Then, by the law of total expectation we have

$$\begin{aligned}
\overline{T} &= \sum_{k=1}^{m} \left(\int_{\mathcal{E}_k} f(x)\,dx \;\; \frac{\beta_{\mathrm{TSP},2}^2}{2} \, \frac{\lambda_k}{v^2\,(1-\varrho_k)^2} \;\; \left[\int_{\mathcal{E}_k} \sqrt{\frac{f(x)}{\int_{\mathcal{E}_k} f(x)\,dx}}\,dx \right]^2 \right) \\
&= \sum_{k=1}^{m} \left(\frac{1}{m} \, \frac{\beta_{\mathrm{TSP},2}^2}{2} \, \frac{\lambda}{v^2\,(1-\varrho)^2} \;\; \left[\int_{\mathcal{E}_k} f^{1/2}(x)\,dx \right]^2 \right) \qquad (7.16) \\
&= \frac{\beta_{\mathrm{TSP},2}^2}{2} \, \frac{\lambda \left[\int_{\mathcal{E}} f^{1/2}(x)\,dx \right]^2}{m^2\,v^2\,(1-\varrho)^2}, \qquad \text{as } \varrho \to 1^-.
\end{aligned}$$

Thus, the bound in equation (2.11) is achieved. We finally show that such multi-vehicle policy is indeed unbiased. In fact, assume steady-state conditions, and let X be the location

of a randomly chosen demand, T be its system time and $\mathcal{S}$ be an arbitrary subset of $\mathcal{E}$; then, by the law of total expectation we have

$$\mathbb{E}\left[T \,|X \in \mathcal{S}\right] = \mathbb{E}\Big[\mathbb{E}\left[T \,|X \in \mathcal{S} \cap \mathcal{E}_k\right] |X \in \mathcal{S}\Big].$$

Since π^* is unbiased, it must hold $\mathbb{E}\left[T \,|X \in \mathcal{S} \cap \mathcal{E}_k\right] = \mathbb{E}\left[T \,|X \in \mathcal{E}_k\right]$; moreover, from the previous analysis, we have $\mathbb{E}\left[T \,|X \in \mathcal{E}_k\right] = \overline{T}_{\pi^*}$, which is a constant independent of $\mathcal{E}_k$. Hence, we finally obtain $\mathbb{E}\left[T \,|X \in \mathcal{S}\right] = \overline{T}_{\pi^*}$. The claim then follows from the fact that $\mathcal{S}$ was chosen arbitrarily. □

Remark 7.4.2. *Some remarks are in order.*

1. *Theorem 7.4.1 proves Conjecture* 1 *(made only for uniform spatial densities f) in [15].*

2. *A similar result can be proven for biased policies; in this case the m-partition should be simultaneously equitable with respect to f and $f^{2/3}$.*

3. *The proof of theorem 7.4.1 relies on m-partitions that are simultaneously equitable with respect to f and $f^{1/2}$. We next show in theorem 7.4.3 that a π^*-partitioning policy using an m-partition which is equitable with respect to $f^{1/2}$ may be unstable; moreover, we show in theorem 7.4.4 that, in general, a π^*-partitioning policy using an m-partition which is equitable with respect to f does not achieve the optimal unbiased performance, however it is always within a factor m of it.*

Theorem 7.4.3. *Given a single-vehicle policy π and m vehicles, a π-partitioning policy using an m-partition which is equitable with respect to $f^{1/2}$ may be unstable for values of ϱ strictly less than one.*

Proof. The arrival rate in subregion $\mathcal{E}_k$, $k \in \{1, \dots, m\}$, is $\lambda_k = \lambda \int_{\mathcal{E}_k} f(x)\, dx$, and therefore the load factor in subregion $\mathcal{E}_k$ (where only one vehicle provides service) is $\varrho_k = \lambda \int_{\mathcal{E}_k} f(x)\, dx\, \bar{s}$. In general, an m-partition that is equitable with respect to $f^{1/2}$ is *not* equitable with respect to f. If this is the case, there exists $\bar{k} \in \{1, \dots, m\}$ and $\varepsilon > 0$ such that $\int_{\mathcal{E}_{\bar{k}}} f(x)\, dx = 1/m + \varepsilon$; then, $\varrho_{\bar{k}} = \varrho + \varepsilon\lambda\bar{s} = \varrho\,(1 + \varepsilon\, m)$. Hence, for every value of ϱ such that $1/(1 + \varepsilon\, m) \leq \varrho < 1$, such multi-vehicle policy is unstable (since in subregion $\mathcal{E}_{\bar{k}}$ we have $\varrho_{\bar{k}} \geq 1$), even though the load factor ϱ is strictly less than one.

□

Theorem 7.4.4. *Let π be a single-vehicle unbiased policy such that in heavy load $\overline{T}_\pi / \overline{T}^*_U \leq \gamma$. As $\varrho \to 1^-$, given the policy π and m vehicles, a π-partitioning policy using an m-partition which is equitable with respect to f does not achieve, in general, the optimal unbiased performance, however it is always within a factor $\gamma\, m$ of it.*

Proof. Let π be an unbiased policy. The environment $\mathcal{E}$ in partitioned into m subregions $\mathcal{E}_k$, $k \in \{1, \dots, m\}$, such that $\int_{\mathcal{E}_k} f(x)\, dx = 1/m$ for all k, and one vehicle is assigned to each subregion. Each vehicle executes the single-vehicle policy π to service demands that fall within its own subregion. Similarly to the proof of theorem 7.4.1 we have, by the law

of total expectation,

$$\begin{aligned}\overline{T} &\geq \sum_{k=1}^{m} \left(\int_{\mathcal{E}_k} f(x)\,dx \;\; \frac{\beta_{\mathrm{TSP},2}^2}{2} \, \frac{\lambda_k}{v^2\,(1-\varrho_k)^2} \;\; \left[\int_{\mathcal{E}_k} \sqrt{\frac{f(x)}{\int_{\mathcal{E}_k} f(x)\,dx}} dx \right]^2 \right) \\ &= \sum_{k=1}^{m} \left(\frac{1}{m} \, \frac{\beta_{\mathrm{TSP},2}^2}{2} \, \frac{\lambda}{v^2\,(1-\varrho)^2} \;\; \left[\int_{\mathcal{E}_k} f^{1/2}(x)\,dx \right]^2 \right) \qquad \text{as } \varrho \to 1^-. \end{aligned} \tag{7.17}$$

Define the vector

$$z = \left(\int_{\mathcal{E}_1} f^{1/2}(x)\,dx, \int_{\mathcal{E}_2} f^{1/2}(x)\,dx, \ldots, \int_{\mathcal{E}_m} f^{1/2}(x)\,dx \right) \in \mathbb{R}^m;$$

we will denote by $\|z\|_2$ and $\|z\|_1$ the 2-norm and the 1-norm, respectively, of vector z. Notice that $\|z\|_1 = \int_{\mathcal{E}} f^{1/2}(x)\,dx$. Since $\|z\|_1 \leq \sqrt{m}\|z\|_2$ for any vector $z \in \mathbb{R}^m$, equation (7.17) becomes

$$\begin{aligned}\overline{T} &\geq \frac{1}{m} \, \frac{\beta_{\mathrm{TSP},2}^2}{2} \, \frac{\lambda}{v^2\,(1-\varrho)^2} \, \|z\|_2^2 \geq \frac{1}{m} \, \frac{\beta_{\mathrm{TSP},2}^2}{2} \, \frac{\lambda}{v^2\,(1-\varrho)^2} \, \frac{\|z\|_1^2}{m} \\ &= \frac{\beta_{\mathrm{TSP},2}^2}{2} \, \frac{\lambda \left[\int_{\mathcal{E}} f^{1/2}(x)\,dx \right]^2}{m^2\,v^2\,(1-\varrho)^2} = \overline{T}_{\mathrm{U}}^*, \qquad \text{as } \varrho \to 1^-, \end{aligned}$$

where $\overline{T}_{\mathrm{U}}^*$ is the optimal unbiased performance. In general, an m-partition that is equitable with respect to f is *not* equitable with respect to $f^{1/2}$; therefore, in general, the inequality $\|z\|_1 \leq \sqrt{m}\|z\|_2$ is strict and $\overline{T}$ is larger than the optimal unbiased performance.

On the other hand, $\|z\|_2 \leq \|z\|_1$ for any vector $z \in \mathbb{R}^m$, therefore by using the same reasoning as in equation (7.17) we obtain

$$\begin{aligned}\overline{T} &\leq \frac{1}{m} \, \gamma \, \frac{\beta_{\mathrm{TSP},2}^2}{2} \, \frac{\lambda}{v^2\,(1-\varrho)^2} \, \|z\|_2^2 \leq \frac{1}{m} \, \gamma \, \frac{\beta_{\mathrm{TSP},2}^2}{2} \, \frac{\lambda}{v^2\,(1-\varrho)^2} \, \|z\|_1^2 \\ &= \gamma \, \frac{\beta_{\mathrm{TSP},2}^2}{2} \, \frac{\lambda \left[\int_{\mathcal{E}} f^{1/2}(x)\,dx \right]^2}{m\,v^2\,(1-\varrho)^2} = \gamma\, m\, T_{\mathrm{U}}^*, \qquad \text{as } \varrho \to 1^-, \end{aligned}$$

where, again, $\overline{T}_{\mathrm{U}}^*$ is the optimal unbiased performance. □

Note that in this case the resulting multi-vehicle policy might be biased.

Remark 7.4.5. *A similar result holds for biased policies: in this case the* 3*-norm is the relevant norm to use.*

7.4.2 Distributed policies for the m-DTRP and discussion

The optimality of the SQM policy and theorem 7.4.1 *suggest* the following distributed multi-vehicle version of the DC policy:

1. the vehicles compute in a distributed way an m-median of $\mathcal{E}$ that induces a Voronoi partition that is equitable with respect to f and $f^{1/2}$,

2. the vehicles assign themselves to the subregions (thus, there is a one-to-one correspondence between vehicles and subregions),

3. each vehicle executes the single-vehicle DC policy to service demands that fall within its own subregion, by using the median of the subregion instead of $\tilde{G}_1^*$.

(In an identical way, we can obtain a multi-vehicle version of the RH policy.)

If there exists, given $\mathcal{E}$ and f, an m-median of $\mathcal{E}$ that induces a Voronoi partition that is equitable with respect to f and $f^{1/2}$, then the above policy is optimal both in light load and arbitrarily close to optimality in heavy load, and stabilizes the system in every load condition. There are two main issues with the above policy, namely (i) existence of an m-median of $\mathcal{E}$ that induces a Voronoi partition that is equitable with respect to f and $f^{1/2}$, and (ii) how to compute it in a distributed way. In section 6.4.5, we showed that for some choices of $\mathcal{E}$ and f a median Voronoi diagram that is equitable with respect to f and $f^{1/2}$ *fails* to exist. On the other hand, in chapter 6, we presented a synchronous, spatially-distributed partitioning algorithm that, for any possible choice of $\mathcal{E}$ and f, provides a partition of $\mathcal{E}$ that is equitable with respect to f and represents a "good" approximation of a median Voronoi diagram (see section 6.5.3; in particular, the motion toward the centroid should be replaced with a motion toward the median). Furthermore, if an m-median of $\mathcal{E}$ that induces a Voronoi partition that is equitable with respect to f exists, the algorithm will locally converge to it. Accordingly, we define the multi-vehicle Divide & Conquer policy as follows

Algorithm 7: Multi-Vehicle DC (m-DC) Policy

1 Each vehicle $k \in \{1, \ldots, m\}$ locally controls a power generator (g_k, w_k) (which is an artificial variable, in other words g_k does not coincide in general with the position of the vehicle). Vehicle k is assigned to the corresponding power cell k.
2 The vehicles run the spatially-distributed partitioning algorithm defined as a set of differential equations in (6.22) (replacing the motion toward the centroid with a motion toward the median), by using as input measure f.
3 Simultaneously, each vehicle executes the single-vehicle DC policy inside its own subregion.

According to theorem 7.4.4 the m-DC policy is within a factor $(1+1/r)\,m$ of the optimal unbiased performance in heavy load (since the algorithm in equation (6.22) *always* provides a partition of $\mathcal{E}$ that is equitable with respect to f), and stabilizes the system in every load condition. In general, the m-DC policy is only *suboptimal* in light load; note, however, that the computation of the global minimum of the Weber function H_m (which is non-convex for $m > 1$) is difficult for $m > 1$ (it is NP-hard for the discrete version of the problem); therefore, for $m > 1$, suboptimality has also to be expected from *any* practical implementation of the SQM policy. If an m-median of $\mathcal{E}$ that induces a Voronoi partition that is equitable with respect to f exists, the partitioning algorithm used by the m-DC will locally converge to it, thus we say that the m-DC policy is "locally" optimal in light load.

Note that, when the density is uniform, a partition that is equitable with respect to f is also equitable with respect to $f^{1/2}$; therefore, when the density is uniform the m-DC policy (with $r \to +\infty$) is an optimal unbiased policy in heavy load (see theorem 7.4.1).

The m-DC policy *adapts* to arrival rate λ, expected on-site service $\bar{s}$, and vehicle's velocity v; however, it requires the knowledge of f. The key feature of the partitioning

algorithm in equation (6.22) is that it is *spatially-distributed*; therefore, the m-DC policy is indeed a spatially-distributed control policy.

The multi-vehicle version of the RH policy (which we call m-RH policy) can be defined in an identical way. Its properties are summarized in Table 7.2.

7.5 On the Case with Zero On-Site Service Time

In the formulation of the m-DTRP we assumed that $\bar{s} > 0$ (see section 2.5.1). However, it is of interest to also study the case $\bar{s} = 0$, since in some applications the on-site service time might actually be zero (or negligible). We show that, perhaps surprisingly, when $\bar{s} = 0$ a π^*-partitioning policy using an m-partition which is equitable with respect to $f^{1/2}$ is an optimal unbiased policy for the m-DTRP. Hence, when $\bar{s} = 0$, optimality does *not* require simultaneous equitability, a considerable simplification.

Specifically, when $\bar{s} = 0$, the light-load regime is defined as $\lambda \to 0^+$, while the heavy-load regime is defined as $\lambda \to +\infty$. Then, it is quite straightforward to verify that all the performance bounds in sections 2.5, 7.2, and 7.3 hold by simply substituting $\varrho = 0$, and by formally changing the limits $\varrho \to 0^+$ and $\varrho \to 1^-$ with $\lambda \to 0^+$ and $\lambda \to +\infty$. For example, equation (2.11) reads

$$\overline{T}^*_{\mathrm{U}} \geq \frac{\beta^2_{\mathrm{TSP},2}}{2} \frac{\lambda \left[\int_{\mathcal{E}} f^{1/2}(x) dx\right]^2}{m^2 v^2} \qquad \text{as } \lambda \to +\infty,$$

and equation (7.2) reads

$$\overline{T}_{\mathrm{DC}}(r) \leq \left(1 + \frac{1}{r}\right) \frac{\beta^2_{\mathrm{TSP},2}}{2} \frac{\lambda \left[\int_{\mathcal{E}} f^{1/2}(x)\, dx\right]^2}{v^2} \qquad \text{as } \lambda \to +\infty,$$

where r is the number of subregions. The major difference is about which type of partition one should use in heavy load (i.e., when λ is large). We have the following result (which can be easily generalized to the case where the single-vehicle unbiased policy π is within a factor $\gamma \geq 1$ of the optimal unbiased performance).

Theorem 7.5.1. *Assume $\bar{s} = 0$. As $\lambda \to +\infty$, given a single-vehicle, optimal unbiased policy π^* and m vehicles, a π^*-partitioning policy using an m-partition which is equitable with respect to $f^{1/2}$ is an optimal unbiased policy for the m-DTRP.*

Proof. The proof is very similar to the one of theorem 7.4.1. Let π^* be an optimal unbiased policy (e.g., the DC policy with $r \to \infty$). Construct an m-partition $\{\mathcal{E}_k\}_{k=1}^m$ of $\mathcal{E}$ that is equitable with respect to $f^{1/2}$; thus $\int_{\mathcal{E}_k} f^{1/2}(x)\, dx = \int_{\mathcal{E}} f^{1/2}(x)\, dx/m$. Notice that the arrival rate λ_k in subregion $\mathcal{E}_k$ is reduced by a factor $\int_{\mathcal{E}_k} f(x)\, dx$, i.e., $\lambda_k = \lambda \int_{\mathcal{E}_k} f(x)\, dx$. Moreover, given that a demand falls in subregion $\mathcal{E}_k$, the conditional density for its location

(whose support is $\mathcal{E}_k$) is $f(x)/\Big(\int_{\mathcal{E}_k} f(x)\,dx\Big)$. By the law of total expectation we have

$$\begin{aligned}\overline{T} &= \sum_{k=1}^{m}\left(\int_{\mathcal{E}_k} f(x)\,dx\;\frac{\beta_{\mathrm{TSP},2}^2}{2}\,\frac{\lambda\int_{\mathcal{E}_k} f(x)\,dx}{v^2}\;\left[\int_{\mathcal{E}_k}\sqrt{\frac{f(x)}{\int_{\mathcal{E}_k} f(x)\,dx}}dx\right]^2\right)\\ &= \sum_{k=1}^{m}\left(\int_{\mathcal{E}_k} f(x)\,dx\;\frac{\beta_{\mathrm{TSP},2}^2}{2}\,\frac{\lambda}{v^2}\;\left[\frac{\int_{\mathcal{E}} f^{1/2}(x)\,dx}{m}\right]^2\right) \qquad (7.18)\\ &= \frac{\beta_{\mathrm{TSP},2}^2}{2}\,\frac{\lambda\left[\int_{\mathcal{E}} f^{1/2}(x)\,dx\right]^2}{m^2\,v^2}, \qquad \text{as } \lambda\to+\infty,\end{aligned}$$

where in the last equality we have used the fact that $\sum_{k=1}^{m}\int_{\mathcal{E}_k} f(x)\,dx = 1$. Thus, the bound in equation (2.11) is achieved. We finally show that such multi-vehicle policy is indeed unbiased. In fact, assume steady-state conditions, and let X be the location of a randomly chosen demand, T be its system time and $\mathcal{S}$ be an arbitrary subset of $\mathcal{E}$; then, by the law of total expectation we have

$$\mathbb{E}\left[T\,|X\in\mathcal{S}\right] = \mathbb{E}\Big[\mathbb{E}\left[T\,|X\in\mathcal{S}\cap\mathcal{E}_k\right]|X\in\mathcal{S}\Big].$$

Since π^* is unbiased, it must hold

$$\mathbb{E}\left[T\,|X\in\mathcal{S}\cap\mathcal{E}_k\right] = \mathbb{E}\left[T\,|X\in\mathcal{E}_k\right] = \beta_{\mathrm{TSP},2}^2\,\lambda\left(\int_{\mathcal{E}} f^{1/2}(x)\,dx\right)^2/(2\,m^2\,v^2),$$

which is a constant independent of $\mathcal{E}_k$. The claim then follows from the fact that $\mathcal{S}$ was chosen arbitrarily. □

Theorem 7.5.1 is very important from a practical viewpoint, since it means that when $\bar{s} = 0$ optimality does not require simultaneous equitability. In light of theorem 7.5.1, when $\bar{s} = 0$ the m-DC policy should be defined in the same way of Algorithm 7, with the exception that the partitioning algorithm in equation (6.22) should partition $\mathcal{E}$ according to $f^{1/2}$ (instead of f). Since the partitioning algorithm in equation (6.22) is then guaranteed to converge to a partition that is equitable with respect to $f^{1/2}$, we conclude that when $\bar{s} = 0$ the spatially-distributed m-DC policy (with the modification that equitability should be with respect to $f^{1/2}$, and with $r\to+\infty$) is an optimal unbiased policy in heavy load for *any* density f.

7.6 Simulation Experiments

In this section we discuss, through the use of simulations, the heavy-load behavior of both the DC and the RH policy, and we comment on their relative performance.

All simulations are performed on a machine with a 2.4GHz Intel Core Duo processor and 4GB of RAM. The code is written in Matlab®7.4 with external calls to the program `linkern`, for which we set a running time bound of one second. In all simulations we consider a circular environment $\mathcal{E} = \{(x,y)\in\mathbb{R}^2\,|\,x^2+y^2\le 1/\pi\}$ (hence, the area of $\mathcal{E}$ is 1), a vehicle's velocity $v = 1$, and an on-site service time uniformly distributed in the interval

Table 7.3: Computation times for DC policy

ϱ	$r = 1$		$r = 16$	
	$\lvert$ TSP $\rvert$	Time	$\lvert$ TSP $\rvert$	Time
0.9	$\simeq 200$ points	$\simeq 0.3s$	$\simeq 20$ points	$\simeq 0.03s$
0.93	$\simeq 400$ points	$\simeq 0.8s$	$\simeq 60$ points	$\simeq 0.1s$
0.95	$\simeq 800$ points	$\simeq 1.1s$	$\simeq 70$ points	$\simeq 0.15s$
0.97	$\simeq 2250$ points	$\simeq 1.7s$	$\simeq 200$ points	$\simeq 0.3s$
0.99	$\simeq 21000$ points	$\simeq 5s$	$\simeq 1300$ points	$\simeq 1.3s$

[0, 1] (thus $\bar{s} = 0.5$). The spatial density $f : \mathcal{E} \to \mathbb{R}_{>0}$ used in the simulation experiments is

$$f(x) = \begin{cases} \frac{1-\delta}{\varepsilon}, & x \in \mathcal{E}_1, \\ \frac{\delta}{1-\varepsilon}, & x \in \mathcal{E}_2, \end{cases}$$

where $\mathcal{E}_1 = \{(x, y) \in \mathbb{R}^2 \,|\, x^2 + y^2 \leq \varepsilon/\pi\}$, $\mathcal{E}_2 = \mathcal{E} \setminus \mathcal{E}_1$, and ε, $\delta \in (0, 1)$. If $1 - \delta = \varepsilon$, then the density f is uniform; if, instead, $1 - \delta > \varepsilon$, then the density f has a peak in subregion $\mathcal{E}_1$. In all simulation experiments, we set $\varepsilon = 0.1$.

7.6.1 Heavy-load performance of the DC policy

We first consider a uniform spatial density, i.e., $\delta = 1 - \varepsilon = 0.9$. We consider 5 values of the load factor, namely $\varrho = 0.9, 0.93, 0.95, 0.97, 0.99$. Circles in the left-hand figure of Figure 7-1 represent the ratios between the experimental system time $\overline{T}_{\mathrm{DC}}(1)$ (i.e., $r = 1$ in the DC policy) and $\overline{T}^*_{\mathrm{U}}$ (whose expression is given in equation (2.11)), for the different values of ϱ. We then repeat the simulation process by executing the DC policy with $r = 16$. Squares in the left-hand figure of Figure 7-1 represent the ratios between the experimental system time $\overline{T}_{\mathrm{DC}}(16)$ (i.e., $r = 16$ in the DC policy) and $\overline{T}^*_{\mathrm{U}}$, for the different values of ϱ. It can be observed that for $r = 1$ the ratios $\overline{T}_{\mathrm{DC}}(1)/\overline{T}^*_{\mathrm{U}}$ decrease as ϱ approaches one and tend to 2, in accordance with the upper bound in theorem 7.2.2 (in fact, $1 + 1/r = 2$ in this case). Similarly, for $r = 16$ the ratios $\overline{T}_{\mathrm{DC}}(16)/\overline{T}^*_{\mathrm{U}}$ decrease as ϱ approaches one and tend to $(1 + 1/16) \approx 1.06$, in accordance again with the upper bound in theorem 7.2.2. These results confirm the theoretical analysis in section 7.2 and suggest that the upper bound (7.2) is indeed tight for the various values of r. Note that for all values of ϱ the experimental system time is *larger* than the upper bound; this is due to one or a combination of the following reasons: First, the upper bound formally holds only in the limit $\varrho \to 1^-$. Second, we are using an approximate solution for the optimal TSP.

In Table 7.3 we show for both $r = 1$ and $r = 16$ the average number of points (denoted by $\lvert$ TSP $\rvert$) through which a TSP tour is computed and the average time required for such computation. Specifically, the computation time includes (i) the time to read the points from an input file, (ii) the TSP computation time, and (iii) the time to save the tour on a file. Recall that we implemented the DC policy by using `linkern` as a solver to generate approximations to the optimal TSP tour, and we set a running time bound of one second for this solver. One can observe that the computation times are rather small even for values of ϱ very close to one; moreover, since the experimental results are very close to the theoretical bounds, we argue that `linkern` (with a running time bound of one second) computes TSP tours very close to the optimal ones. Hence, we conclude that the DC policy can be effectively implemented in real-word applications by using `linkern` as a subroutine.

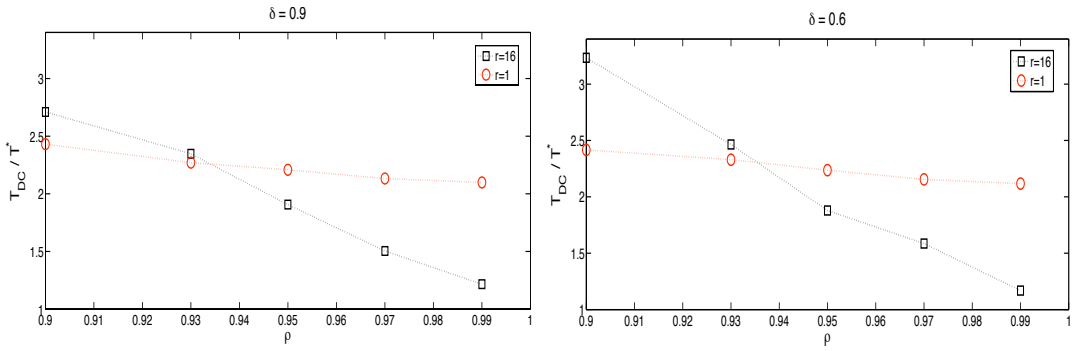


Figure 7-1: Left Figure: Ratio between experimental system time under the DC policy and $\overline{T}^*_{\mathrm{U}}$ (whose expression is given in equation (2.11)) in the case of uniform density (i.e., $\delta = 0.9$). Right Figure: Ratio between experimental system time under the DC policy and $\overline{T}^*_{\mathrm{U}}$ in the case of non-uniform density ($\delta = 0.6$). Circles correspond to the DC policy with $r = 1$, while squares correspond to the DC policy with $r = 16$.

The same set of simulations is performed for a non-uniform spatial density; in particular we consider $\delta = 0.6$, i.e., a peak in the small subregion $\mathcal{E}_1$. Note that in this case a simultaneously equitable partition can be trivially obtained by using radial cuts. Results are shown in the right-hand figure of Figure 7-1. Circles represent the ratios between the experimental system time and $\overline{T}^*_{\mathrm{U}}$ for $r = 1$, while squares represent the ratios between the experimental system time and $\overline{T}^*_{\mathrm{U}}$ for $r = 16$. As before, the results are in accordance with the upper bound in theorem 7.2.2.

Finally, from Figure 7-1 it can be observed that as $\varrho \to 1^-$ the DC policy with $r = 16$ performs better than the DC policy with $r = 1$, precisely by a factor 2 in accordance with the upper bound in theorem 7.2.2. On the other hand, for moderate values of ϱ, say $\varrho \approx 0.9$, the DC policy with $r = 1$ performs better than the DC policy with $r = 16$. This result is expected since for moderate values of ϱ and "large" values of r the number of demands in each subregion is low, and therefore equation (2.7) is no longer applicable (while it is still applicable for $r = 1$). The computation times are very similar to those shown in Table 7.3 and therefore they are omitted.

7.6.2 Heavy-load performance of the RH policy

We first consider a uniform spatial density. We consider 3 values of the load factor, namely $\varrho = 0.95, 0.97, 0.99$, and 7 values of η, namely $\eta = 0.1, 0.2, 0.3, 0.7, 0.8, 0.9, 1$. Formally, the RH policy is not defined for $\eta = 1$, however, by setting $\eta = 1$ in the definition of the RH policy, we simply obtain the DC policy with $r = 1$. The left-hand figure of Figure 7-2 shows the ratios between the experimental system time $\overline{T}_{\mathrm{RH}}(\eta)$ and $\overline{T}^*_{\mathrm{U}}$ (whose expression is given in equation (2.11)) for each pair (ϱ, η). The same set of simulations is performed for a non-uniform spatial density; in particular we consider $\delta = 0.6$. Results are shown in the right-hand figure of Figure 7-2. The results confirm that the system time $\overline{T}_{\mathrm{RH}}$ follows the $1/(1-\varrho)^2$ growth predicted by theorem 7.3.1. In section 7.3 we argued that in heavy load $T_{\mathrm{RH}}(\eta) \leq 2\,\overline{T}^*_{\mathrm{U}}$ (see equation (7.15)) for all $\eta \in (0, 1)$; from Figure 7-2, it can be observed that as ϱ approaches one this asymptotic bound seems to be confirmed, both for a uniform f and a non-uniform f. We also argued in section 7.3 that the performance of the RH policy should improve as the horizon becomes shorter: this is indeed the case, in

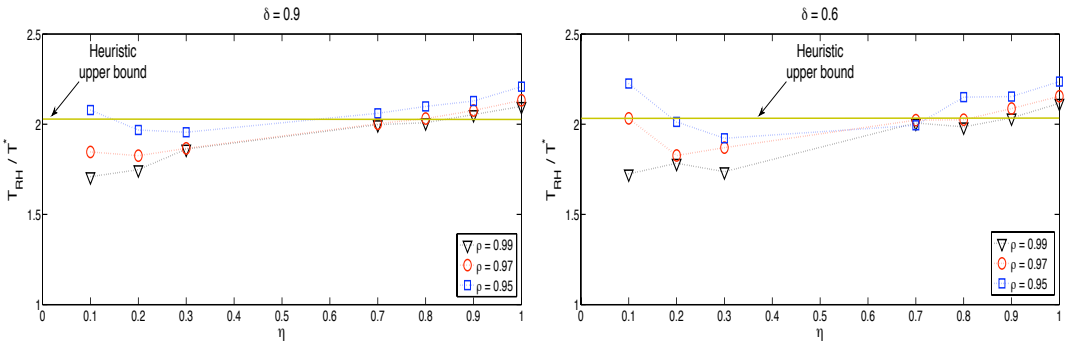


Figure 7-2: Left Figure: Ratio between experimental system time under the RH policy and $\overline{T}^*_{\mathrm{U}}$ (whose expression is given in equation (2.11)) in the case of uniform density (i.e., $\delta = 0.9$). Right Figure: Ratio between experimental system time under the RH policy and $\overline{T}^*_{\mathrm{U}}$ in the case of non-uniform density ($\delta = 0.6$).

particular simulation results show that optimal performance is achieved for $\eta \approx 0.2$. Finally, the computation times for the RH policy with any η are very similar to those of the DC policy with $r = 1$ and therefore they are omitted.

7.6.3 Comparison between DC policy and RH policy

The RH policy should be compared with the version of the DC policy that is also adaptive with respect to f, i.e., with the DC policy with $r = 1$. Indeed, as observed before, the DC policy with $r = 1$ is the same as the RH policy when we set $\eta = 1$. Therefore, we can compare the two policies by looking at Figure 7-2. One should note that the RH policy with $\eta \approx 0.2$ performs consistently better than the DC policy with $r = 1$ (which, again, is the same as the RH policy with $\eta = 1$), in particular it decreases the system time by $\approx 20\%$.

7.6.4 Performance of the multi-vehicle DC policy

In this section we study, through the use of simulations, the behavior of the m-DC policy; we focus, in particular, on the heavy-load scenario. When the density f is uniform, a partition that is equitable with respect to f is also equitable with respect to $f^{1/2}$. By following the same arguments as in equation (7.16) one can show that in heavy load the steady-state system time within the entire environment follows the same behavior of the steady-state system time within each subregion. Hence, when f is uniform and the system is in heavy load the simulation results for the m-DC policy are virtually identical to the simulation results for the 1-DC policy (presented above), and therefore they are omitted.

When the density f is *not* uniform, a partition that is equitable with respect to f is not necessarily equitable with respect to $f^{1/2}$. Theorem 7.4.4, however, guarantees that the m-DC policy in heavy load is within a factor $(1+1/r)\,m$ of the optimal unbiased performance. We verify this result through simulation experiments. We consider a square environment $\mathcal{E} = \{(x, y) \in \mathbb{R}^2 \,|\, 0 \le x \le 1,\, 0 \le y \le 1\}$ (hence, the area of $\mathcal{E}$ is 1), vehicles' velocity $v = 1$, and an on-site service time uniformly distributed in the interval $[0,\, 1]$ (thus $\bar{s} = 0.5$). The spatial density used in the simulations is:

$$f(x) = \begin{cases} 3, & x \in \mathcal{E}_1, \\ 1/3, & x \in \mathcal{E}_2, \end{cases}$$

where $\mathcal{E}_1 = \{(x, y) \in \mathbb{R}^2 \,|\, 0.5 \leq x \leq 1,\, 0.5 \leq y \leq 1\}$ and $\mathcal{E}_2 = \mathcal{E} \setminus \mathcal{E}_1$. We consider a load factor $\varrho = 0.9$ and 4 values for the number m of vehicles, namely $m = 2, 4, 6, 8$. Within each subregion, we simulate the DC policy with $r = 1$, which has a constant factor guarantee equal to 2 (see theorem 7.2.2). The circles in Figure 7-3 represent the ratios between the experimental system time $\overline{T}_{m\text{-DC}}$ and $\overline{T}^*_{\text{U}}$, for the different values of m. The squares in Figure 7-3 represent the upper bounds for such ratios, which, according to theorem 7.4.4, are equal to $2\,m$. The simulation results show that the ratios $\overline{T}_{m\text{-DC}}/\overline{T}^*_{\text{U}}$ seem not to increase linearly with m, or, in other words, that the upper bound in theorem 7.4.4 is conservative.

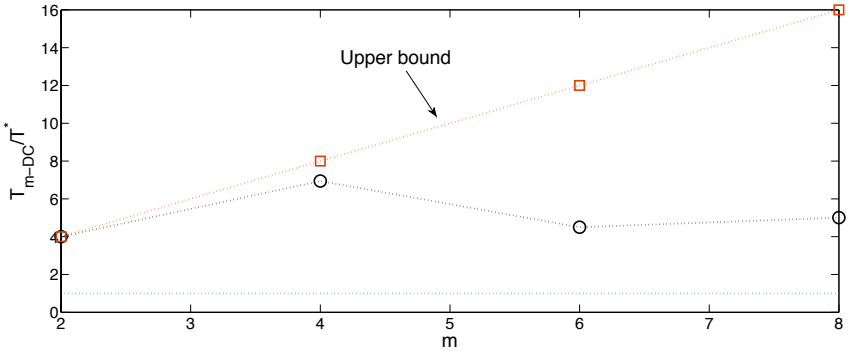


Figure 7-3: The circles represent the ratios between the experimental system times under the m-DC policy (with $r = 1$) and $\overline{T}^*_{\text{U}}$; the squares represent the theoretical upper bounds on these ratios. The load factor is $\varrho = 0.9$.

7.7 Conclusion

The focus of this chapter was on the m-DTRP. Note, however, that in many DVR problems (e.g., DVR with stochastic time constraints and DVR with priorities, see chapters 3 and 4) partitioning policies provide "good" performance. Hence, we argue that combining the spatially-distributed partitioning algorithms of chapter 6 with appropriate partitioning policies and single-vehicle routing policies yields adaptive and distributed algorithms for a wide variety of DVR problems. The approach presented in this chapter is therefore rather general and not specifically tailored to the m-DTRP.

This chapter leaves numerous important extensions open for further research. A first line of research is to devise spatially-distributed algorithms that provide simultaneously equitable partitions; such algorithms would make the m-DC policy (with $r \to +\infty$) optimal for any density f. Second, finding an unbiased policy for the 1-DTRP that is optimal in heavy load and does not rely on the knowledge of f is both practically important and theoretically interesting. Finally, it is also of interest to design optimal or near-optimal policies that are *biased* in heavy load and that are adaptive and distributed (recall that allowing biased service results in a strict reduction of the optimal system time for any non-uniform density f).

Chapter 8

Conclusions

The algorithmic queueing theory approach developed in this dissertation provides a new way of studying robotic systems in dynamically changing environments. This approach consists of three basics steps: 1) queueing model of the robotic system and analysis of its structure; 2) establishment of fundamental limitations on performance, independent of algorithms; and 3) design of algorithms that are either optimal or constant-factor away from optimal. We have argued how algorithmic queueing theory overcomes many limitations of existing approaches to dynamic vehicle routing problems, most notably the online algorithm approach. A key results that we have achieved by adopting the algorithmic queueing theory approach is a systematic method for lifting single-vehicle routing policies to distributed multi-vehicle routing policies with provable performance guarantees.

In the following, we first summarize the results presented in each chapter; collectively, chapters 3 through 7 provide examples of applications of algorithmic queueing theory, and at the same time they contribute to its development. We conclude by presenting some directions for future research.

8.1 Summary

In chapter 3 we studied a dynamic vehicle routing problem where demands have stochastic deadlines on their waiting times. After a careful formulation of the problem, we established a lower bound on the optimal number of vehicles. In deriving such lower bound, we introduced a novel type of facility location problem, and we provided some analysis and algorithms for it. Then, we analyzed and characterized two service policies; in particular, one of the two policies is optimal in light load, while the other one is within a small factor of the optimum in heavy load and under some additional assumptions. Finally, we discussed scaling laws for the number of vehicles.

In chapter 4 we introduced a dynamic vehicle routing problem with priority classes. We captured the priority levels of classes by writing the system time as a convex combination of the system time of each class. We determined a lower bound on the achievable values of the convex combination of the class system times. We then presented the Separate Queues policy and showed that it performs within a constant factor of the lower bound, which depends only on the number of the classes.

In chapter 5 we studied a dynamic pick-up delivery problem with multiple vehicles of unit capacity and we argued that this is a reasonable model for transportation-on-demand and mobility-on-demand systems. We first presented a policy that is optimal in light load;

then we studied the Randomized Batch policy and showed that it performs within a constant factor of the optimal performance.

In chapter 6 we presented provably correct, spatially-distributed control policies that allow a team of vehicles to achieve a convex and equitable partition of a convex environment. We also considered the issue of achieving convex and equitable partitions with special properties (e.g., close to Voronoi diagrams). These algorithms can find applications in many problems, including dynamic vehicle routing (as shown in chapter 7), and wireless networks.

Finally, in chapter 7 our goal was to design unbiased policies for the m-DTRP that (i) are adaptive (in particular do *not* require the knowledge of the arrival rate λ and the statistics of the on-site service time), (ii) enjoy provable performance guarantees (in particular, are provably stable under any load condition), and (iii) are spatially-distributed. To achieve this goal, we introduced two new policies for the 1-DTRP, namely the Divide & Conquer policy and the Receding Horizon policy, which were subsequently extended to the multi-vehicle case through the idea of partitioning policies. The focus of chapter 7 was on the m-DTRP; however, we argued that combining the spatially distributed partitioning algorithms of chapter 6 with appropriate partitioning policies and single-vehicle routing policies yields adaptive and distributed algorithms for a wide variety of DVR problems.

8.2 Future Directions

In this dissertation we have studied dynamic task-assignment and scheduling problems for robotic networks, by using an algorithmic queueing theory approach. This research has provided rigorous answers to many questions in this field, but it has also raised a number of new ones. Some of them have been discussed at the end of each chapter. In this section we outline some additional directions for future research.

DVR in moderate loads Most of the results presented in this dissertation hold in the asymptotic regimes $\varrho \to 0^+$ (where the problem basically becomes one of locational optimization) and $\varrho \to 1^-$ (where the "border effects", e.g., the shape of the environment, become negligible). Specifically, while stability properties have been rigorously established under any load condition (see, e.g., the analysis of the policies in chapter 7), tight performance guarantees only hold in those asymptotic regimes. Addressing optimality of performance in the intermediate regime would be very important both on a theoretical and on a practical level. Unfortunately, the analysis techniques presented in this dissertation might not be sufficient to fulfill this objective. While new analysis tools are being developed, Monte Carlo simulations might provide preliminary insights.

Different performance criterions In this thesis, we have mainly considered as optimality criterion the steady-state expected waiting time. It is also of interest to consider formulations that take into account second-order moments and large-deviation probabilities, as in some applications (e.g., surveillance of protected environments) it might be important to limit variability.

Realistic vehicle dynamics and environments In most of the scenarios we have considered omni-directional vehicles with first order dynamics. More realistic dynamical models will have to be taken into account for practical application to UAVs or other systems. A possibility is to integrate the results presented in this thesis with the

results in [87], where bounds on the Traveling Salesman Problem for vehicles with motion constraints are derived. A preliminary step in this direction can be found in [51]. Also, the environment $\mathcal{E}$ is usually assumed to be a convex set. This assumption might be satisfied in UAV applications, but is highly unrealistic, for instance, in an urban setting. The inclusion of obstacles or other realistic features in the description of $\mathcal{E}$ requires novel methods to integrate queueing theory, computational geometry, and motion planning.

Extensions to other DVR problems There are many other additional key problems in robotic systems that could benefit from being studied under the algorithmic queuing theory framework. Examples include search and rescue missions, force protection, map maintenance, and pursuit-evasion.

DVR and game theory The main approach to devise distributed multi-vehicle routing policies was to combine spatially-distributed partitioning algorithms with appropriate partitioning policies and single-vehicle routing policies. Although this approach is rather powerful, it is not the only possible solution. An alternative solution might be to reformulate the dynamic routing problems studied in this thesis within the theory of *non-cooperative games*, which appears to be an effective framework for the design of distributed motion coordination algorithms. A possible starting point is the work in [6], where a game-theoretical formulation of a static vehicle-target assignment problem is presented.

Convergence rates of distributed partitioning algorithms The distributed partitioning algorithms presented in chapter 6 have guarantees on their steady-state closed-loop performance, but lack a rigorous characterization of their convergence rates. On the other hand, an understanding of their transient behavior is necessary for real-world applications. Hence, it is of interest to develop bounds on their convergence rates. To tackle this challenging problem, one might try to build upon the analysis techniques developed in [40], where bounds on the convergence rate of the Lloyd algorithm are derived.

Partitioning algorithms with minimal information exchange In [43], the authors propose a distributed partitioning algorithm that converges to a multi-median Voronoi diagram and only requires asynchronous pairwise (so-called gossip) communication. An interesting problem would be to devise distributed algorithms for equitable partitioning that provide partitions close to a median Voronoi diagram and that rely on the gossip communication model. In general, an important direction for future research is to devise distributed algorithms for equitable partitioning that require the minimal amount of information exchange.

Partitioning algorithms for structured environments An important extension for the partitioning algorithms presented in chapter 6 is to consider the setting of structured environments (ranging from simple nonconvex polygons to more realistic ground environments). Such extension add several challenges, including connectivity maintenance for the robotic network.

Bibliography

[1] P. K. Agarwal and M. Sharir. Efficient algorithms for geometric optimization. *ACM Computing Surveys*, 30(4):412–458, 1998.

[2] M. Alighanbari and J. P. How. A robust approach to the UAV task assignment problem. *International Journal on Robust and Nonlinear Control*, 18(2):118–134, 2008.

[3] D. Applegate, R. Bixby, V. Chvátal, and W. Cook. On the solution of traveling salesman problems. In *Documenta Mathematica, Journal der Deutschen Mathematiker-Vereinigung*, pages 645–656, Berlin, Germany, August 1998. Proceedings of the International Congress of Mathematicians, Extra Volume ICM III.

[4] S. Arora. Nearly linear time approximation scheme for Euclidean TSP and other geometric problems. In *Proc. 38th IEEE Annual Symposium on Foundations of Computer Science*, pages 554–563, Miami Beach, FL, October 1997.

[5] A. Arsie, K. Savla, and E. Frazzoli. Efficient routing algorithms for multiple vehicles with no explicit communications. *IEEE Transactions on Automatic Control*, 54(10):2302–2317, 2009.

[6] G. Arslan, J. R. Marden, and J. S. Shamma. Autonomous vehicle-target assignment: A game theoretic formulation. *ASME Journal on Dynamic Systems, Measurement, and Control*, 129(5):584–596, 2007.

[7] S. Asmussen. *Applied Probability and Queues*. Springer-Verlag, New York, NY, second edition, 2003.

[8] F. Aurenhammer. Power diagrams: properties, algorithms and applications. *SIAM Journal on Computing*, 16(1):78–96, 1987.

[9] F. Baccelli, P. Boyer, and G. Hebuterne. Single-server queues with impatient customers. *Advances in Applied Probability*, 16(4):887–905, 1984.

[10] O. Baron, O. Berman, D. Krass, and Q. Wang. The equitable location problem on the plane. *European Journal of Operational Research*, 183(2):578–590, 2007.

[11] R. W. Beard, T. W. McLain, M. A. Goodrich, and E. P. Anderson. Coordinated target assignment and intercept for unmanned air vehicles. *IEEE Transactions on Robotics and Automation*, 18(6):911–922, 2002.

[12] G. Berbeglia, J. F. Cordeau, and G. Laporte. Dynamic pickup and delivery problems. *European Journal of Operational Research*, 202(1):8 – 15, 2010.

[13] D. Bertsimas, I. Ch. Paschalidis, and J. N. Tsitsiklis. Optimization of multiclass queueing networks: Polyhedral and nonlinear characterizations of achievable performance. *The Annals of Applied Probability*, 4(1):43–75, 1994.

[14] D. J. Bertsimas and G. J. van Ryzin. A stochastic and dynamic vehicle routing problem in the Euclidean plane. *Operations Research*, 39:601–615, 1991.

[15] 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.

[16] 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.

[17] S. Bespamyatnikh, D. Kirkpatrick, and J. Snoeyink. Generalizing ham sandwich cuts to equitable subdivisions. *Discrete and Computational Geometry*, 24:605–622, 2000.

[18] N. Bisnik, A. Abouzeid, and V. Isler. Stochastic event capture using mobile sensors subject to a quality metric. *IEEE Transactions on Robotics*, 23:676–692, 2007.

[19] N. K. Boots and H. Tijms. A multiserver queueing system with impatient customers. *Management Science*, 45(3):444–448, 1999.

[20] O. Bräysy and M. Gendreau. Vehicle routing problem with time windows, part I: Route construction and local search algorithms. *Transportation Science*, 39(1):104–118, 2005.

[21] O. Bräysy and M. Gendreau. Vehicle routing problem with time windows, part II: Metaheuristics. *Transportation Science*, 39(1):119–139, 2005.

[22] A. Bressan. *Personal Communication*, 2008.

[23] M. Buehler, K. Iagnemma, and S. Singh, editors. *The 2005 DARPA Grand Challenge: the Great Robot Race*. Number 36 in Springer Tracts in Advanced Robotics. Springer, 2007.

[24] M. Buehler, K. Iagnemma, and S. Singh, editors. *The DARPA Urban Challenge: Autonomous Vehicles in City Traffic*. Number 56 in Springer Tracts in Advanced Robotics. Springer, 2009.

[25] F. Bullo, J. Cortés, and S. Martínez. *Distributed Control of Robotic Networks*. Applied Mathematics Series. Princeton University Press, 2009. Available at http://www.coordinationbook.info.

[26] J Carlsson, D Ge, A. Subramaniam, A. Wu, and Y. Ye. Solving min-max multi-depot vehicle routing problem. *Report*, 2007.

[27] J.G. Carlsson, B. Armbruster, and Y. Ye. Finding equitable convex partitions of points in a polygon efficiently. *To appear in The ACM Transactions on Algorithms*, 2008.

[28] I. Chavel. *Eigenvalues in Riemannian Geometry*. Academic Press, New York, NY, 1984.

[29] N. Christofides. Worst-case analysis of a new heuristic for the traveling salesman problem. Technical Report 388, Graduate School of Industrial Administration, Carnegie Mellon University, 1976.

[30] E. G. Coffman Jr. and I. Mitrani. A characterization of waiting time performance realizable by single-server queues. *Operations Research*, 28(3):810–821, 1980.

[31] J. F. Cordeau, G. Laporte, J. Y. Potvin, and M.W.P. Savelsbergh. Transportation on demand. In C. Barnhart and G. Laporte, editors, *Transportation, Handbooks in Operations Research and Management Science*, volume 14, pages 429–466. Elsevier, Amsterdam, The Netherlands, 2007.

[32] J. Cortés, S. Martínez, and F. Bullo. Spatially-distributed coverage optimization and control with limited-range interactions. *ESAIM: Control, Optimisation & Calculus of Variations*, 11:691–719, 2005.

[33] J. Cortés, S. Martínez, T. Karatas, and F. Bullo. Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation*, 20(2):243–255, 2004.

[34] D. J. Daley. General customer impatience in the queue GI/G/1. *Journal of Applied Probability*, 2(1):186–205, 1965.

[35] L. Dechevsky and S. Penev. On shape-preserving wavelet estimators of cumulative distribution functions and densities. *Stochastic Analysis and Applications*, 16(3):423–462, 1998.

[36] J. Desrosiers, Y. Dumas, M. M. Solomon, and F. Soumis. Time constrained routing and scheduling. In M. O. Ball, T. L. Magnanti, C. L. Monma, and G. L. Nemhauser, editors, *Handbooks in Operations Research and Management Science*, chapter 8, pages 35–139. Elsevier Science Publishers, Amsterdam, The Netherlands, 1995.

[37] R. Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer Verlag, New York, second edition, 2000.

[38] V. Dobric and J. E. Yukich. Asymptotics for transportation cost in high dimensions. *Journal of Theoretical Probability*, 8(1):97–118, 1995.

[39] Z. Drezner, editor. *Facility Location: A Survey of Applications and Methods*. Series in Operations Research. Springer, 1995.

[40] Q. Du, M. Emelianenko, and L. Ju. Convergence of the lloyd algorithm for computing centroidal voronoi tessellations. *SIAM Journal on Numerical Analysis*, 44(1):102–119, 2006.

[41] J. Enright, K. Savla, and E. Frazzoli. Coverage control for nonholonomic agents. In *Proc. IEEE Conf. on Decision and Control*, pages 4250–4256, 2008.

[42] A. F. Filippov. *Differential Equations with Discontinuous Righthand Sides*, volume 18 of *Mathematics and its Applications*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1988.

[43] P. Frasca, R. Carli, and F. Bullo. Multiagent coverage algorithms with gossip communication: Control systems on the space of partitions. In *American Control Conference*, pages 2228–2235, St. Louis, MO, June 2009.

[44] R. G. Gallager. *Graph Theory*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996.

[45] 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.

[46] A. Hatcher. *Algebraic Topology*. Cambridge University Press, Cambridge, U.K., 2002.

[47] J. Houdayer, J.H. Boutet de Monvel, and O.C. Martin. Comparing mean field and euclidean matching problems. *The European Physical Journal B - Condensed Matter and Complex Systems*, 6(3):383–393, 1998.

[48] H. Imai, M. Iri, and Kazuo Murota. Voronoi diagram in the Laguerre geometry and its applications. *SIAM Journal on Computing*, 14(1):93–105, 1985.

[49] S. Irani, X. Lu, and A. Regan. On-line algorithms for the dynamic traveling repair problem. *Journal of Scheduling*, 7(3):243–258, 2004.

[50] V. Isler. Placement and distributed deployment of sensor teams for triangulation based localization. In *Proc. IEEE Conf. on Robotics and Automation*, pages 3095–3100, Orlando, FL, USA, 2006.

[51] S. Itani, E. Frazzoli, and M. A. Dahleh. Dynamic travelling repairperson problem for dynamic systems. In *Proc. IEEE Conf. on Decision and Control*, pages 465–470, Cancun, Mexico, 2008.

[52] P. Jaillet and M. R. Wagner. Online routing problems: Value of advanced information and improved competitive ratios. *Transportation Science*, 40(2):200–210, 2006.

[53] L. Kleinrock. *Queueing Systems. Volume II: Computer Applications*. Wiley, New York, 1976.

[54] I. N. Kovalenko. Some queuing problems with restrictions. *Theory of Probability and its Applications*, 6(2):204–208, 1961.

[55] N. N. Krasovskii. *Stability of Motion: Applications of Lyapunov's Second Method to Differential Systems and Equations with Delay*. Stanford University Press, CA, USA, 1963.

[56] S. O. Krumke, W. E. de Paepe, D. Poensgen, and L. Stougie. News from the online traveling repairmain. *Theoretical Computer Science*, 295(1-3):279–294, 2003.

[57] A. Kwok and S. Martinez. Energy-balancing cooperative strategies for sensor deployment. In *Proc. IEEE Conf. on Decision and Control*, pages 6136–6141, New Orleans, LA, December 2007.

[58] R. C. Larson and A. R. Odoni. *Urban Operations Research*. Prentice-Hall, Englewood Cliffs, NJ, 1981.

[59] K. Laventall and J. Cortés. Coverage control by multi-robot networks with limited-range anisotropic sensory. *International Journal of Control*, 82(6):1113–1121, 2009.

[60] R.E. Lillo and M. Martin. Stability in queues with impatient customers. *Stochastic Models*, 17(3):375–389, 2001.

[61] S. Lin and B. W. Kernighan. An effective heuristic algorithm for the traveling-salesman problem. *Operations Research*, 21:498–516, 1973.

[62] B. Liu, Z. Liu, and D. Towsley. On the capacity of hybrid wireless networks. In *IEEE INFOCOM 2003*, pages 1543–1552, San Francisco, CA, April 2003.

[63] S. P. Lloyd. Least-squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

[64] D. G. Luenberger. *Linear and Nonlinear Programming*. Addison-Wesley, 2 edition, 1984.

[65] N. Megiddo and K. J. Supowit. On the complexity of some common geometric location problems. *SIAM Journal on Computing*, 13(1):182–196, 1984.

[66] P. Midgley. The role of smart bike-sharing systems in urban mobility. Available at http://www.gtkp.org/uploads/public/documents/Knowledge/The-Role-of-Smart-Bike.pdf, 2009.

[67] W. J. Mitchell, C. E. Borroni-Bird, and L. D. Burns. *Reinventing the Automobile*. MIT Press, 2010.

[68] S. Mitrovic-Minic and R. Krishnamurti. The multiple TSP with time windows: vehicle bounds based on precedence graphs. *Operations Research Letters*, 34(1):111–120, 2006.

[69] B. J. Moore and K. M. Passino. Distributed task assignment for mobile agents. *IEEE Transactions on Automatic Control*, 52(4):749–753, 2007.

[70] J. Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, 5(1):32–38, March 1957.

[71] D. Newman. The hexagon theorem. *IEEE Transactions on Information Theory*, 28(2):137–139, Mar 1982.

[72] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu. *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*. John Wiley & Sons, New York, NY, 2000.

[73] J. D. Papastavrou. A stochastic and dynamic routing policy using branching processes with state dependent immigration. *European Journal of Operational Research*, 95:167–177, 1996.

[74] S. N. Parragh, K. F. Doerner, and R. F. Hartl. A survey on pickup and delivery problems. *Journal fur Betriebswirtschaft*, 58(2):81–117, 2008.

[75] M. Pavone, A. Arsie, E. Frazzoli, and F. Bullo. Equitable partitioning policies for robotic networks. *IEEE Transactions on Automatic Control*, 2009. Provisionally Accepted, available at `http://arxiv.org/abs/0903.5267`.

[76] M. Pavone, A. Arsie, E. Frazzoli, and F. Bullo. Equitable partitioning policies for robotic networks. In *Proc. IEEE Conf. on Robotics and Automation*, pages 2356–2361, Kobe, Japan, May 2009.

[77] M. Pavone, N. Bisnik, E. Frazzoli, and V. Isler. A stochastic and dynamic vehicle routing problem with time windows and customer impatience. *Mobile Networks and Applications*, 14(3):350–364, 2009.

[78] M. Pavone and E. Frazzoli. Dynamic vehicle routing with stochastic time constraints. In *Proc. IEEE Conf. on Robotics and Automation*, Anchorage, Alaska, May 2010. To Appear.

[79] M. Pavone, E. Frazzoli, and F. Bullo. Decentralized algorithms for stochastic and dynamic vehicle routing with general demand distribution. In *Proc. IEEE Conf. on Decision and Control*, pages 4869–4874, New Orleans, LA, 2007.

[80] M. Pavone, E. Frazzoli, and F. Bullo. Distributed policies for equitable partitioning: Theory and applications. In *Proc. IEEE Conf. on Decision and Control*, pages 4191–4197, Cancun, Mexico, 2008.

[81] 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*, 2009. Provisionally Accepted, `http://arxiv.org/abs/0903.3624`.

[82] M. Pavone, S. L. Smith, F. Bullo, and E. Frazzoli. Dynamic multi-vehicle routing with multiple classes of demands. In *American Control Conference*, pages 604–609, St. Louis, MO, June 2009.

[83] 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.

[84] H. N. Psaraftis. Dynamic programming solution to the single vehicle many-to-many immediate request dial-a-ride problem. *Transportation Science*, 14(2):130–154, 1980.

[85] T. Sakai. Balanced convex partitions of measures in $\mathbb{R}^2$. *Graphs and Combinatorics*, 18:169–192, 2002.

[86] M. W. P. Savelsbergh. Local search in routing problems with time windows. *Annals of Operations Research*, 4(1):285–305, 1985.

[87] K. Savla, E. Frazzoli, and F. Bullo. Traveling salesperson problems for the Dubins vehicle. *IEEE Transactions on Automatic Control*, 53(6):1378 –1391, 2008.

[88] D. D. Sleator and R. E. Tarjan. Amortized efficiency of list update and paging rules. *Communications of the ACM*, 28(2):202–208, 1985.

[89] S. L. Smith and F. Bullo. Monotonic target assignment for robotic networks. *IEEE Transactions on Automatic Control*, 54(9):2042–2057, 2009.

[90] S. L. Smith, M. Pavone, F. Bullo, and E. Frazzoli. Dynamic traveling repairperson with priority demands. In *Proc. IEEE Conf. on Decision and Control*, pages 1206–1211, Cancun, Mexico, 2008.

[91] S. L. Smith, M. Pavone, F. Bullo, and E. Frazzoli. Dynamic vehicle routing with priority classes of stochastic demands. *SIAM Journal on Control and Optimization*, 48(5):3224–3245, 2010.

[92] W. L. Smith. Regenerative stochastic processes. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 232(1188):6–31, 1955.

[93] W. L. Smith. Renewal theory and its ramifications. *Journal of the Royal Statistical Society. Series B (Methodological)*, 20(2):243–302, 1958.

[94] M. M. Solomon. Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints. *Operations Research*, 35(2):254–265, 1987.

[95] R.E. Stanford. Reneging phenomena in single channel queues. *Mathematics Of Operations Research*, 4(2):162–178, 1979.

[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, 1990.

[97] S. Jr. Stidham. Regenerative processes in the theory of queues, with applications to the alternating-priority queue. *Advances in Applied Probability*, 4(3):542–577, 1972.

[98] M. R. Swihart and J. D. Papastavrou. A stochastic and dynamic model for the single-vehicle pick-up and delivery problem. *European Journal of Operational Research*, 114(3):447–464, 1999.

[99] P. Toth and D. Vigo, editors. *The Vehicle Routing Problem*. Monographs on Discrete Mathematics and Applications. SIAM, 2001.

[100] P. Van Hentenryck, R. Bent, and E. Upfal. Online stochastic optimization under time constraints. *Annals of Operations Research*, 2009. To appear.

[101] H.A. Waisanen, D. Shah, and M.A. Dahleh. A dynamic pickup and delivery problem in mobile networks under information constraints. *IEEE Transactions on Automatic Control*, 53(6):1419–1433, 2008.

[102] A.R. Ward and P.W. Glynn. A diffusion approximation for a markovian queue with reneging. *Queueing Syst. Theory Appl.*, 43(1/2):103–128, 2003.

[103] A.R. Ward and P.W. Glynn. A diffusion approximation for a GI/GI/1 queue with balking or reneging. *Queueing Syst. Theory Appl.*, 50(4):371–400, 2005.

[104] A.R. Ward and S. Kumar. Asymptotically optimal admission control of a queue with impatient customers. *Math. Oper. Res.*, 33(1):167–202, 2008.

[105] R. W. Wolff. Poisson arrivals see time averages. *Operations Research*, 30(2):223–231, 1982.

[106] R. W. Wolff. *Stochastic Modeling and the Theory of Queues*. Prentice Hall, 1989.

[107] G. Xing, C. Lu, R. Pless, and J. A. O'Sullivan. Co-grid: an efficient coverage maintenance protocol for distributed sensor networks. In *3rd International Symposium on Information Processing in Sensor Networks (IPSN)*, pages 414–423, New York, NY, USA, 2004. ACM Press.

[108] H. Xu. *Optimal Policies for Stochastic and Dynamic Vehicle Routing Problems*. Dept. of Civil and Environmental Engineering, Massachusetts Institute of Technology, Cambridge, MA, 1995.

[109] S. Zeltyn and A. Mandelbaum. Call centers with impatient customers: Many-server asymptotics of the M/M/n+G queue. *Queueing Systems*, 51(3):361–402, 2005.