Geometric Partitioning Algorithms for Fair Division of Geographic Resources

A THESIS
SUBMITTED TO THE FACULTY OF
UNIVERSITY OF MINNESOTA
BY

Raghuveer Devulapalli

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Advisor: John Gunnar Carlsson

July 2014

Acknowledgements

The journey of my PhD life would not have been possible without the aid and support of several people over the past few years. I must first express my gratitude towards my advisor, Professor John Gunnar Carlsson. I cannot thank him enough for providing me with a research home at a crucial time in my graduate student life. His leadership, support and tremendous hard work have set an example I can only hope to match some day. To work with him has been a real pleasure to me, with heaps of fun and excitement. Above all, John has made me feel like his friend, which I appreciate from my heart. The ideas presented in the current proposal are largely his inspiration; the plural "we" that has been used in the proposal refers to Prof. Carlsson and me.

In addition, I would like to thank the INFORMS interactive session committees for presenting first and third prize awards to two of our publications that were a result of this dissertation. I would also like to thank the INFORMS Computing Society (ICS) prize committee for acknowledging and publicizing the significance of this work.

I have been very privileged to get to know and to collaborate with Professor Yinyu Ye at Stanford University and I thank him for all the support he has provided during my PhD. I would like to thank Professor Ravi Janardhan and his research group for giving me the opportunity to closely interact with them and making me feel like a part of their own group. Sitting in their research group meetings has been of immense learning help and I am truly inspired with the level of perfectness and discipline that Prof. Janardhan brings in to his research and personal life.

I owe the most to my parents and my sister for the support they provided me through my entire life. Without their sacrifices, I would have never been where I am. A warm and special thanks to Pavani for her constant encouragement, understanding and patience during all the good and hard times I faced. Life in Minneapolis would not have been memorable without the presence and support of my long list of friends: Rahul Saladi, Haritha Bellam, Anand, GV, Thaseem, Bala, Jikku, Govind, Palak, Aruna and Divyanshu. The "Legendary" Wednesday nights we have had for years now will be sorely missed. I also would like to thank some of my fellow PhD students: Adhika Lie, Hamid, Fan Jia, Mehdi and Ying Li. They each helped make my time in the PhD program more fun and interesting. To all my friends and family in U.S and back home, I express a warm thank you.

To,
My Parents

Abstract

This dissertation focuses on a fundamental but under-researched problem: how does one divide a piece of territory into smaller pieces in an efficient way? In particular, we are interested in *map segmentation problem* of partitioning a geographic region into smaller subregions for allocating resources or distributing a workload among multiple agents. This work would result in useful solutions for a variety of fundamental problems, ranging from congressional districting, facility location, and supply chain management to air traffic control and vehicle routing. In a typical map segmentation problem, we are given a geographic region R, a probability density function defined on R (representing, say population density, distribution of a natural resource, or locations of clients) and a set of points in R (representing, say service facilities or vehicle depots). We seek a *partition* of R that is a collection of disjoint sub-regions $\{R_1, ..., R_n\}$ such that $\bigcup_i R_i = R$, that optimizes some objective function while satisfying a shape condition. As examples of shape conditions, we may require that all sub-regions be compact, convex, star convex, simply connected (not having holes), connected, or merely measurable.

Such problems are difficult because the search space is infinite-dimensional (since we are designing boundaries between sub-regions) and because the shape conditions are generally difficult to enforce using standard optimization methods. There are also many interesting variants and extensions to this problem. It is often the case that the optimal partition for a problem changes over time as new information about the region is collected. In that case, we have an *online* problem and we must re-draw the sub-region boundaries as time progresses. In addition, we often prefer to construct these sub-regions in a *decentralized* fashion: that is, the sub-region assigned to agent i should be computable using only local information to agent i (such as nearby neighbors or information about its surroundings), and the optimal boundary between two sub-regions should be computable using only knowledge available to those two agents.

This dissertation is an attempt to design geometric algorithms aiming to solve the above mentioned problems keeping in view the various design constraints. We describe the drawbacks of the current approach to solving map segmentation problems, its ineffectiveness to impose geometric shape conditions and its limited utility in solving the online version of the problem. Using an intrinsically interdisciplinary approach, combining elements from variational calculus, computational geometry, geometric probability theory, and vector space optimization, we present an approach where we formulate the problems geometrically and then use a fast geometric algorithm to solve them. We demonstrate our success by solving problems having a particular choice of objective function and enforcing certain shape conditions. In fact, it turns out that such methods

actually give useful insights and algorithms into classical location problems such as the continuous k-medians problem, where the aim is to find optimal locations for facilities. We use a map segmentation technique to present a constant factor approximation algorithm to solve the continuous k-medians problem in a convex polygon. We conclude this thesis by describing how we intend to build on this success and develop algorithms to solve larger classes of these problems.

Contents

List of Figures

Chapter 1

Introduction

1.1 The Map Segmentation Problem

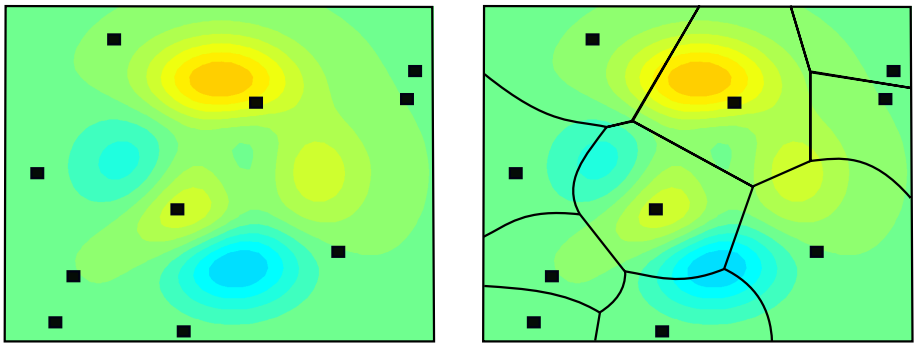
Dividing a territory into smaller sub-regions is a problem encountered in several domains such as vehicle routing, facility location, congressional districting, air traffic control, supply chain management, surveillance, reconnaissance. Indeed, as exemplified in [32], effective division of geographic territory has been a fundamental problem facing society since the times of antiquity:

> *"Homer, in describing the Phaiakian settlement in Scheria, speaks of a circuit wall for the city.... Implicit in the foundation of new colonies was the notion of equality among the members, exemplified in the division of their prime resource, the land. To achieve this, accurate measurement and equitable division were from the outset essential, even when gods or privileged men were to be honored with larger or better assignments."*

The goal of this dissertation is to develop tractable methods for *partitioning* a geographic region into smaller sub-regions to optimally distribute resources or balance workloads among multiple agents. We refer to this problem as a *Map Segmentation* problem. In a typical map segmentation problem, inputs consist of a geographic region R together with some additional problem parameters Ω, such as obstacles, fixed points (representing, for example, vehicle depots), or a probability density function (representing, for example, distribution of demand over the region); see Figure 1.1. We seek a *partition* of R, that is, a collection

of disjoint sub-regions $R_1, ..., R_n$ such that $\bigcup_i R_i = R$, that optimizes a chosen objective function. It is natural to impose additional *shape constraints* on the sub-regions requiring them to be, for instance, compact, convex, star convex, simply connected (not having holes), or merely connected. Some typical optimization objectives include:

- minimizing the workload of a fleet of vehicles assigned to these sub-regions,
- balancing the consumption of a resource,
- using a partition to determine the optimal locations of supply depots, or
- maximizing the surveillance coverage of all regions.



(a) Input: Region R, density function $f(\cdot)$ and locations $p_1, ..., p_n$

(b) Output: Sub-regions $R_1, ..., R_n$

Figure 1.1: Map segmentation problem

Formally, we can write up a map segmentation problem as follows:

$$\begin{aligned} \underset{R_1, R_2 ... R_n}{\text{minimize}} \; & G\left(R_1, ..., R_n | \Omega\right) \quad s.t. \\ & R_i \cap R_j = \emptyset \quad \forall i \neq j \\ & \bigcup_i R_i = R \\ & R_i \text{ obeys a shape condition } \forall i \end{aligned} \tag{1.1}$$

In principle, solving a map segmentation problem is a colossal task, since it involves designing *boundaries* between sub-regions which means solving for an infinite number of unknown variables. Further complications arise when we are forced to reconcile the allocation objectives with geometric shape constraints on

the sub-regions. Solving such problems requires one to combine tools from a variety of disciplines, such as mathematical optimization, computational geometry, geometric probability theory, and geospatial analysis. We will now look at some application of map segmentation problem.

1.1.1 Applications

Voronoi Diagrams

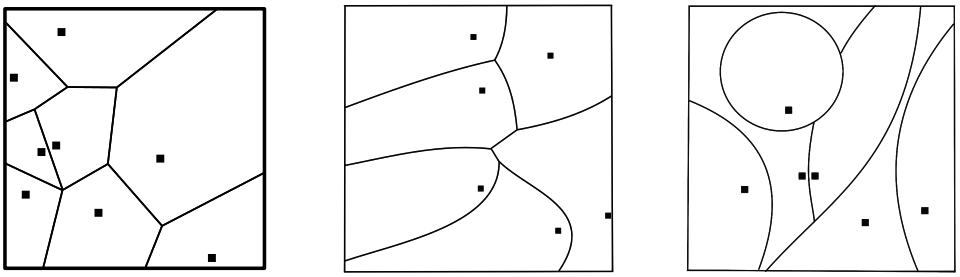
A canonical example of map segmentation is the *Voronoi diagram*, which is the solution to Dewdney and Vranch's so-called "post office problem" [48], along with several other map segmentation problems [13]. The input region is partitioned into smaller sub-regions based on proximity to a set of "landmark" points, as shown in Figure 1.2a. Simply put, given a geographic region R containing a set of landmark (or generator) points $\{p_1, \ldots, p_n\}$ the Voronoi cell associated with p_i, denoted V_i, is defined as the set of points $x \in R$ where $\|x - p_i\| \leq \|x - p_j\|$ for all indices j:

$$V_i = \{x \in R : \|x - p_i\| \leq \|x - p_j\| \, \forall j\} \ ,$$

where $\|\cdot\|$ most commonly denotes the Euclidean norm. All of the Voronoi regions can be shown to be convex polygons (some of them may be infinite) containing their generator point. Voronoi partitions have tremendous applications in almost every field one can imagine. The survey paper [13] and the book [96] are excellent sources for a comprehensive list of its variations, properties and their applications.

Voronoi partitions have been generalized in a number of ways (see [13, 96]). One such variation is by re-defining the way we compute distance between points. If one assigns a weight vector $\mathbf{w} = \{w_1, \ldots, w_n\}$ associated with the landmark points, we can define the distance between a landmark point and point in region to be a weighted combination of Euclidean distance and the corresponding weight. If the distance function is the Euclidean distance to the point minus its additive weight, the resulting diagrams are called *additively-weighted Voronoi diagram*, in which we define Voronoi cells as

$$V_i = \{x \in R : \|x - p_i\| - w_i \leq \|x - p_j\| - w_j \, \forall j\} \ .$$



(a) Simple Voronoi diagram

(b) Additive Voronoi diagram

(c) Multiplicative Voronoi diagram

Figure 1.2: Weighted Voronoi diagrams in the unit square.

Alternatively, if the distance function is defined as a product of Euclidean distance to the point and the weight, then the resulting partition is called the *multiplicatively-weighted Voronoi diagram*, in which the Voronoi cells are defined as:

$$V_i = \left\{ x \in R : \frac{\|x - p_i\|}{w_i} \leq \frac{\|x - p_j\|}{w_j} \, \forall j \right\} .$$

Both these variations are shown in Figure 1.2. Observe that in both of these generalizations, the size of cell V_i increases as w_i increases. It is also not hard to show that the boundaries between adjacent Voronoi cells V_i and V_j are hyperbolic arcs in the additive model and circular arcs in the multiplicative model. Also, it is easy to see that both additive and multiplicative Voronoi diagram reduce to a simple Voronoi diagram if all of the weights are equal. We will later see in Chapter 2 that these partitions enjoy various other useful properties in variational segmentation problems.

Fair Division of Resources

A closely related problem to the map segmentation problem is the *fair division* problem. Here the objective is to distribute a set of goods or resource among n players (or agents) in a "fair" manner. There can be two classes of fair division problems according to the kind of objects to be divided: indivisible (discrete) and divisible (continuous) goods. There may be more than one notion of what is fair or not. Each agent is assumed to have his or her own utility density function $u_i(\cdot)$ defined on the resource. The goal is usually to divide the resource into n pieces in such a way that each agent i receives his or her "favorite" piece i.e. the piece that maximizes his or her utility. Various versions of this problem

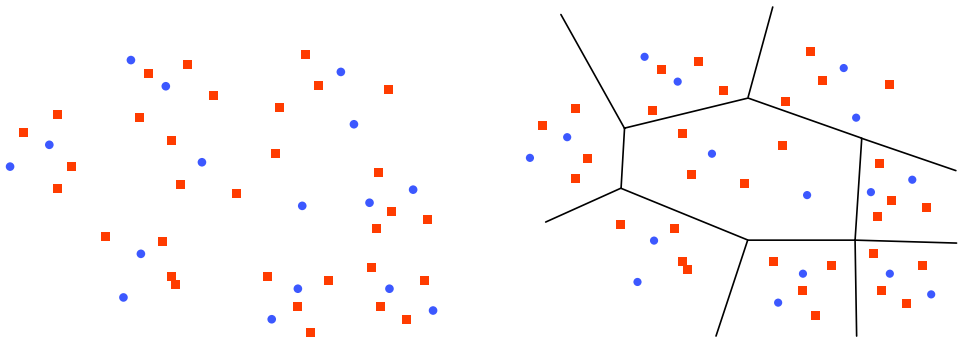


Figure 1.3: Input and output to a red-blue partitioning problem.

exist in multiple applications, such as facility location [7, 17], robotics [99, 100], cake cutting [16], fair division of land[33, 67], air traffic control [18] and vehicle routing [42, 102]. More details on the wide range of practical applications can be found in the survey paper [35].

Red-Blue Partitioning

Another example of a map segmentation problem is a *red-blue partition*. In such a problem, we are given a set P_1 of ng red points and a set P_2 of nh blue points in the plane, and we seek a partition of the plane into n convex pieces such that each piece contains g red points and h blue points, as shown in Figure 1.3. The papers [25, 71, 110] all prove that such a partition exists when the points are in general position and give fast algorithms (or constructive proofs) for finding one. In the framework of map segmentation, $R = \mathbb{R}^2$, $\Omega = P_1 \cup P_2$, "regular" means convex, and the objective function depends on the number of red and blue points in each piece:

$$G\left(R_1, ..., R_n | \Omega\right) := \max_i \left\{ |\text{card}(P_1 \cap R_i) - g| + |\text{card}(P_2 \cap R_i) - h| \right\}$$

where $\text{card}(S)$ denotes the cardinality of set S. It is easy to see that $G\left(R_1, ..., R_n | \Omega\right) = 0$ if and only if each piece R_i contains g red points and h blue points. Red-blue partitioning problems have been extensively studied in the last decade and several non-convex variants have also been considered [19, 21, 27].

Military Relevance

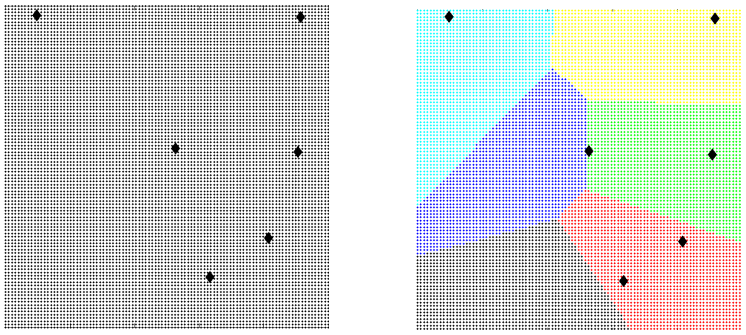
With drastic increase in drones and under water autonomous vehicles, the optimal allocation of autonomous vehicles in a geographic area is becoming increasingly important for military and surveillance operations. Partitioning a geographic area, being a key step in such operations, will underlie the design of control policies for routing unmanned vehicles. In such applications, we may have an *online* problem instance: that is, the relevant problem parameters (such as a probability density describing likely threats or obstacles in the region) may change over time as new information is collected. This might require us to compute optimal sub-regions as a function of time. An additional difficulty arises from the design constraint that requires sub-regions to be constructed in a *decentralized* fashion, meaning that the boundaries between sub-regions must be computed using information purely "local" to an agent, such as its nearby neighbors and information about immediate surroundings. The problem of equitable distribution of workload among a fleet of unmanned or manned vehicles in a military setting is a classic example of *online and decentralized map segmentation* application since the operating environment is dynamic and the vehicles have a limited radius of communication. According to the Office of Naval Research (ONR) Fact Sheet on optimization-based UAV Planning:

> "*The technology being developed by ONR is aimed at creating optimization-based tools for optimally allocating and deploying UAVs ... These decision aids will account for the timing of the selection, allocation, and operation decisions by considering multi-stage mathematical-optimization models.*"

Indeed, designing *online* and *decentralized* optimization algorithms to deploy fleets of unmanned vehicles was recently identified as a "Basic Research Challenge" by the ONR [91].

1.1.2 Solving Map Segmentation Problems

Over the years, there have been numerous attempts at solving various versions of map segmentation problems. We roughly categorize these works in three types of methods listed below.



(a) Input (b) Output

Figure 1.4: Solving map segmentation problem by pixelating the input region.

Integer Programming

The traditional way to solve this problem has been to break the region into small discrete "pixels" and assign binary variables to each pixel. This essentially allows us to reduce the infinite variable set to a finite set of variables, leading to a large combinatorial integer program. This method has been used generously by the operations research community [65, 76, 84, 123, 60, 124], although such an approach suffers from three drawbacks. First, solving integer programs by itself is very inefficient. Large values of n lead to large-scale combinatorial programs which are often computationally intractable. Second, it may be difficult, if not impossible, to impose important geometric shape conditions within a combinatorial framework. Third, integer programs cannot generally be solved efficiently in an online fashion, rendering this approach largely useless in solving the online version of the problem.

Recursive Subdivision

The next most popular approach is to divide the input region R *recursively*. That is, we decompose R into smaller regions, which themselves are further decomposed, and so on and so forth until we reach the required number of sub-regions, as shown in Figure 1.5. Such a method is used extensively in computational geometry for building advanced data structures like Binary Space Partition tree, in which an n dimensional space is recursively subdivided into convex sets by hyperplanes. Other data structures like the Kd-tree, Quadtree, and Octree are also build by similar recursive subdivision techniques. These

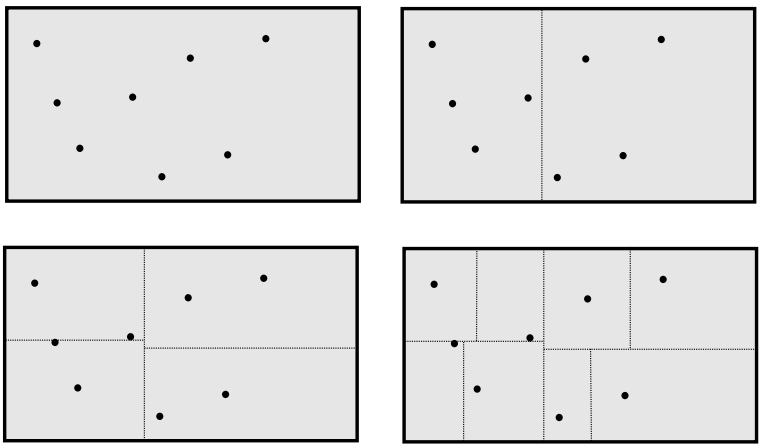
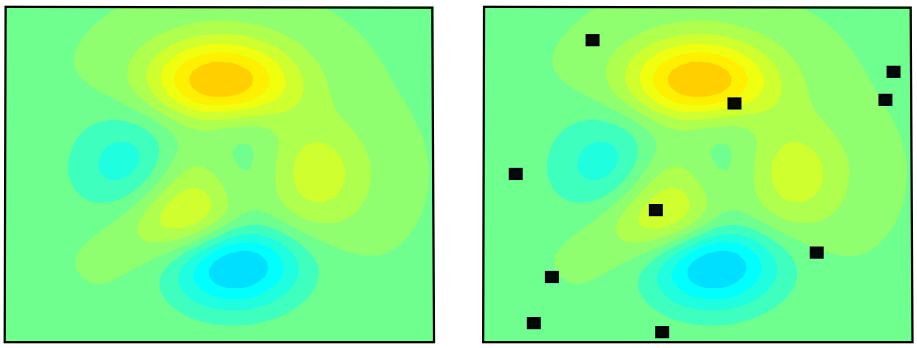


Figure 1.5: A recursive subdivision of a rectangle into $n = 8$ sub-regions; we first divide the rectangle into two pieces, which are themselves subdivided into two more, and so forth.

have immense applications in computer graphics, n-dimensional range searching, CAD design, and robotics. Recursive subdivision techniques have also been used in solving operations search problems. The papers [7, 3] uses recursive techniques to divide a convex region into n equal area convex pieces to balance the load among multiple facilities or vehicles. The papers [41, 42] also use similar techniques to solve other operation research problems.

Low Dimensional Parameterization

Another approach to solving map segmentation problems is by restricting to partitions that can be described using small number of parameters. An example of a such a partition is the *power diagram* [12] (also known as *Laguerre Voronoi diagram* [68]), which is a generalization of Voronoi diagrams. Each generator point p_i in a power diagram is associated with some weight w_i which determines the cell of p_i. It has the property that larger values of w_i cause sub-region R_i to expand while preserving convexity of all the sub-regions. The paper[12] provides efficient algorithms to compute power diagrams. Power diagrams are popular in computational geometry. For example, the power diagram can be used to efficiently compute the volume of a union of spheres [14], to build data structures for testing whether a point belongs to a union of disks [68] and to develop algorithms for finding the closest two balls in a set of balls [61]. It also has been used extensively in robotics community. The authors in [99, 100] give



(a) Input: Region R and density function $f(\cdot)$

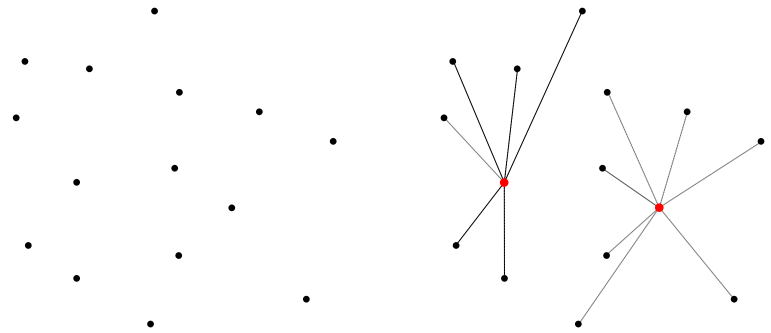
(b) Output: Optimal locations $p_1, ..., p_n$

Figure 1.6: Facility location problem

algorithms for breaking a convex region into sub-regions so as to balance the workload of collection of robots, subject to shape constraint that the sub-regions have to be convex. They prove that an optimal solution to their problem is a power diagram and reduced to the problem to finding just the weights of the power diagram.

1.2 Facility Location

In the map segmentation problem we assume that the location of the set of agents or facilities are fixed. In this sense, these problems can be thought of as the second phase of a *location-allocation* model in which we first find locations of agents and then segment the map to allocate resources among them. Alternatively, we could also think of map segmentation problem to be the first phase an *allocation-location* algorithm. One could first design partitions or service districts for facilities and then place facilities within these districts in a desirable way. Even though this seems counter-intuitive, there are several applications where this is useful. A popular method that uses such an approach is the Lloyd algorithm [80, 122] to compute a centroidal Voronoi tessellation. In this dissertation, we will use a map segmentation technique to help solve a facility location problem, known as the continuous k-medians problem.



(a) Input (b) Output

Figure 1.7: The discrete 2-median problem

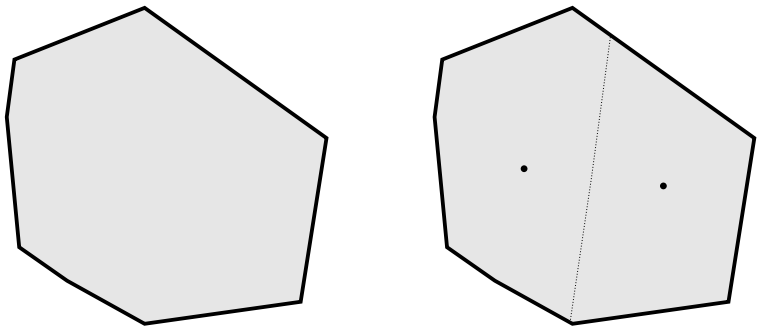
1.2.1 The k-Medians Problem

Given a set of two dimensional demand points, $X = \{x_1, ..., x_n\}$, the objective of the k-medians problem is to chose k landmark points (or facilities) among these n points so as to minimize the average distance between a point in X and its nearest landmark point (see Figure 1.7). The objective function assumes there are no fixed and backbone network costs, but just the transportation costs between facilities and demand points. A natural variation to this problem is the case where the demand points form are a continuum. In such a case, the demand points are assumed to be uniformly distributed in a region R (instead of being a finite set of points P). The objective now is to distribute k points within R, so as to minimize the average distance $\min_i \|x - p_i\|$ between a point x in R and its nearest facility p_i:

$$\underset{p_1,p_2...p_k}{\text{minimize}} \iint_R \min_i \|x - p_i\| dA \quad s.t.$$

$$p_i \in R \quad \forall i$$

We call this problem the *continuous k-medians problem* or the *continuous Fermat-Weber problem* (see Figure 1.8). Although the discrete version of this problem is extremely well studied, the continuous version has received very little attention by researchers. The authors in [43] use a recursive map segmentation technique to give an approximation algorithm for this problem when the input region is a convex polygon. They provide a lower bound to the problem and prove that their algorithm is a 2.73 constant factor approximation algorithm. In this thesis, we will develop a new lower bound to the continuous k-medians problem and



(a) Input (b) Output

Figure 1.8: The continuous 2-median problem

also improve their analysis to reduce the approximation factor from 2.73 to 2.03. We also provide a heuristic method based on the Lloyd algorithm to solve the k-medians problems when the input region is non-convex.

1.3 Overview and List of Publications

The work in this thesis attempts to solve various versions of the map segmentation problem and the continuous k-medians problem for convex and non-convex input regions. Aspects of the dissertation has already been published or submitted for review. Organization of the dissertation along with the list of publications is summarized below:

Chapter 2: In this chapter, we consider the problem of fair division of geographic resources among multiple agents. We assume that each agent has an utility function associated with him or her. Our objective is to divide the input region into sub-regions so as to "balance" the overall utilities on the regions. We present two formulations of this problem and develop a method to solve for the optimal solutions efficiently.

- J.G.Carlsson, R.Devulapalli. "*Dividing a territory among several facilities*", INFORMS Journal on Computing, August 2012.
- J.G.Carlsson, Erik Carlsson, R.Devulapalli. "*Shadow prices in territory division*", Under review, Networks and Spatial Economics.

Chapter 3 In this chapter, we will show how to incorporate certain geometric shape constraints using the formulation described in Chapter 2. We also consider a dynamic version of the map segmentation problem in which the density function describing the distribution of resource varies over time. We derive a set of partial differential equations that describe the evolution of the optimal sub-regions over time.

Chapter 4: This chapter deals with the map segmentation problem when the input region is not simply connected. We consider the problem of minimizing the total workload of service vehicles, like drones, servicing over a geographic territory. We assume that the territory is a connected polygonal region, i.e. a simply connected polygon containing a set of simply connected obstacles. We give a fast algorithm, based on an infinite-dimensional optimization formulation, that divides the territory into compact, connected sub-regions, each of which contains a vehicle depot, such that all regions have equal area.

- J.G.Carlsson, Erik Carlsson, R.Devulapalli. "*Balancing Workload for Autonomous Vehicles over a Geographic Territory.* Accepted in June 2013 to appear in the proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, November 2013.

Chapter 5: This chapter deals with a few non-geographic application of partitioning algorithms. We look at the use of a partitioning algorithm to compute a market-clearing price vector in an *aggregate demand system* or a variation of the classical Fisher exchange market. We also look at use of weighted Voronoi diagram as a tool for data visualization.

- R.Devulapalli, M.Quist, J.G.Carlsson. "*Spatial partitioning algorithms for data visualization*", Accepted in October 2013 to appear in the proceedings of Visualization and Data Analysis, February 2014.

- J.G.Carlsson, Erik Carlsson, R.Devulapalli. "*Computing a market-clearing price vector in an address model of differentiated demand.* Working paper.

Chapter 6: This chapter looks at use of map segmentation techniques to solve facility location problems. Specifically, we look at partitioning techniques

to develop approximation algorithms to solve the continuous k-medians problem in a convex and non-convex region.

- R.Devulapalli, J.G.Carlsson. "*The continuous k-medians problem in a convex polygon.* Under review, Operations Research Letters.

Chapter 2

Map Segmentation for Fair Division of Resources

2.1 Introduction

Segmenting a map to efficiently divide resources among a collection of service vehicles, facilities, or *agents* is a common objective in many disciplines. In practice, it is often desirable to find a *balanced* assignment so as to divide resources in an equitable fashion. Such assignment policies are commonly encountered in robotics [72, 101], queueing theory [15, 66, 74], vehicle routing [42, 64, 102], and facility location [7, 17, 23, 49], among others. One can formulate the map segmentation problem for fair division of resources among n agents as follows. Suppose that R is a geographic region in the plane which we are to partition among n agents, that is, we are to select n sub-regions $R_1, \ldots, R_n$ of R such that $R_i \cap R_j = \emptyset$ for all pairs and $\bigcup_i R_i = R$. We will assume that R is a connected, polygonal region with non-empty interior. Letting $u_i(\cdot)$ denote a "utility density" function associated with agent i, we can represent the overall utility of agent i as the integral $\iint_{R_i} u_i(x)\, dA$, where R_i denotes the sub-region assigned to agent i. In order to generalize this model further, let us also assume that $f(\cdot)$ is a given probability density function (representing population or distribution of a natural resource, for example) on R, so that the overall utility of agent i is the integral $\iint_{R_i} f(x)u_i(x)\, dA$. This problem has been previously considered in many different domains for various particular forms of $f(\cdot)$ and $u_i(\cdot)$ which we will discuss shortly.

The key issue in the preceding problem is how to construct the sub-regions in a *balanced*, or *equitable*, fashion. One way to partition the region is to maximize the overall utilities of the agents while imposing constraints on the amounts of $f(\cdot)$ that are contained in them. That is, our problem can be written as

$$\begin{aligned} \underset{R_1,\ldots,R_n}{\text{maximize}} \sum_{i=1}^{n} \iint_{R_i} f(x)u_i(x)\,dA \qquad & s.t. \\ \iint_{R_i} f(x)\,dA &= q_i \quad \forall i \\ R_i \cap R_j &= \emptyset \quad \forall i \neq j \\ \bigcup_i R_i &= R \end{aligned} \tag{2.1}$$

where the q_i are given constants. A second way to partition the region in a balanced way is to maximize the *minimum* utility of all of the sub-regions:

$$\begin{aligned} \underset{R_1,\ldots,R_n}{\text{maximize}} \min_i \left\{ \iint_{R_i} f(x)u_i(x)\,dA \right\} \qquad & s.t. \\ R_i \cap R_j &= \emptyset \quad \forall i \neq j \\ \bigcup_i R_i &= R\,. \end{aligned} \tag{2.2}$$

In this chapter we show that the boundaries between the optimal sub-regions to problem (2.1) are curves of the form

$$x : u_i(x) - u_j(x) = \text{constant}$$

and that the boundaries between the optimal sub-regions to problem (2.2) are curves of the form

$$x : \frac{u_i(x)}{u_j(x)} = \text{constant}$$

provided that either $u_i(x) > 0$ for all i and x or $u_i(x) < 0$ for all i and x (except for possibly a set of measure zero). Although this turns out to be a simple and immediate consequence of complementary slackness in vector space optimization, it allows us to obtain very concise, constructive proofs to well-known existing results in equitable partitioning. Moreover, our proof technique reduces both balanced partitioning problems to n-dimensional convex optimization problems, which allows us to actually solve both problems efficiently in

practice by computing a set of shadow prices associated with the agents. In Section 2.4, we give fast algorithms for solving (2.1) and (2.2) by showing how to compute a subgradient vector for either problem, which enables us to use (for example) a *cutting plane method* to find the optimal partition.

2.1.1 Related Work

Problems (2.1) and (2.2) have already been studied for specific forms of the functions $u_i(\cdot)$. The case of problem (2.1) where $u_i(x) = -\|x - p_i\|^2$ for fixed points $p_i \in R$ was first analyzed in [11] and later in [99, 100]; the former gives a fast algorithm for optimal partitioning for the case where $f(\cdot)$ is an atomic distribution and the latter gives a control scheme that converges to an optimal partition for smooth $f(\cdot)$. The case of (2.1) where $u_i(x) = -\|x - p_i\|$ and $f(\cdot)$ is a uniform distribution was analyzed in [7] (whose analysis also extends cleanly to general distributions $f(\cdot)$) who also give an approximation algorithm for simultaneously locating the points p_i and designing partitions.

2.2 Applications

Before discussing the solution method for problems (2.1) and (2.2), we first give several geographic applications thereof.

Facility Districting In the paper [7], the authors considered the problem of dividing a territory among a collection of facilities $\{p_1, \ldots, p_n\}$ so as to balance the workloads of those facilities, where the workload of a facility covering region R_i is modeled as

$$\iint_{R_i} f(x)\|x - p_i\| \, dA \, .$$

The simple intuition is that the cost of p_i providing service to a point x is simply proportional to the distance between x and p_i. Thus here, we can consider either problem (2.1) or (2.2) with $u_i(x) = -\|x - p_i\|$. Other variations are possible when we model the cost of service between a demand point x and a facility i is of the form $c\,(x, p_i) = \alpha_i \|x - p_i\|^k$ (that is, $u_i(x) = -\|x - p_i\|^k$). We can also extend this to consider the case where each agent i has a *collection* of facilities $\{p_1^i, \ldots, p_{n_i}^i\}$ (representing multiple branches of a store, for example), which

gives $u_i(x) = -\min_{j\in\{1,\ldots,n_i\}} \|x - p_j^i\|$.

Power Diagrams A *power diagram* of a set of points $\{p_1, \ldots, p_n\} \subset \mathbb{R}^d$ is a generalization of the well-known *Voronoi diagram* in which additive weights w_i are associated with each of the points. Specifically, an arbitrary point $x \in \mathbb{R}^d$ is said to belong to the *power cell* C_i of the point p_i if

$$\|x - p_i\|^2 - w_i \leq \|x - p_j\|^2 - w_j$$

for all indices $j \in \{1, \ldots, n\}$. It turns out that the cells C_i in any power diagram are always convex. Power diagrams are widely applied in such diverse domains as robotics [99, 100, 103], air traffic control [57], and sensor placement [62]. The paper [11] proves that the optimal solution to problem (2.1) is always a power diagram whenever $u_i(x) = -\|x - p_i\|^2$, i.e. when our objective is to minimize the sum of the mass moments of the sub-regions. An immediate corollary (which was independently re-derived in [99, 100]) is that, given any set of points $\{p_1, \ldots, p_n\}$ in a region R, a probability density $f(\cdot)$ defined on R, and any positive vector $(q_1, \ldots, q_n)$ whose entries sum to 1, there always exists a set of weights whose resulting power diagram satisfies $\iint_{C_i} f(x)\, dA = q_i$ for each i.

Fair Division In a *fair division* problem we are concerned with dividing the region R "fairly" [34]. One interpretation of fairness is precisely the formulation (2.2) with $f(x) = 1$ everywhere and generic functions $u_i(\cdot)$. It turns out (as we will show momentarily) that the optimal solution to (2.2) has equal values of $\iint_{R_i} f(x)u_i(x)\, dA$ (i.e. $\iint_{R_i} u_i(x)\, dA$ in our case) for all i; thus, each agent's utility is equal at optimality.

Maximizing Influence The *gravity hypothesis* [111] is a well-known geographic theory that states that the "interaction" between two points x and y generally decays at a rate proportional to the inverse square of the distance between them, i.e. $1/\|x - y\|^2$. Here "interaction" might be measured by economic activity [22], migration [79], or transport [109], for example. It follows that if p_i is the "capital" of region R_i then the total "influence" (economic,

political, or cultural) that p_i exercises over R_i can be approximated as

$$\iint_{R_i} \frac{f(x)}{\|x - p_i\|^2}\, dA$$

where $f(\cdot)$ represents a population density, so that $u_i(x) = 1/\|x - p_i\|^2$. Since the integral blows up near the points p_i it is natural to truncate $u_i(x)$ in a small ϵ-neighborhood of p_i. A natural application of problems (2.1) and (2.2) arises in the division of territory among a collection of "capital cities" $\{p_1, \ldots, p_n\}$ so as to maximize the influence that the cities exercise over their respective domains R_i while respecting overall constraints on the populations of these domains. The related *geographic potential model* [112] postulates that the interaction between two points decays at a rate directly proportional to the inverse of the distance between them, which gives $u_i(x) = 1/\|x - p_i\|$.

Hospital Districting It has been observed among geographers that, in rural regions, the frequency of a person's visits to a hospital decays exponentially in their distance to the hospital [69, 114]. Thus, if a hospital located at a point p_i provides service to a region R_i, the long-term workload that the hospital experiences can roughly be approximated by

$$\iint_{R_i} f(x) \exp(-\|x - p_i\|)\, dA$$

where $f(\cdot)$ represents a population density. Since the goal of districting among hospitals is generally to ensure equity among the service regions, it is therefore natural to consider problem (2.2) with $u_i(x) = \exp(-\|x - p_i\|)$. Exponential distance decay also occurs in measuring ecological similarity between areas [90, 105]; in this case, problem (2.2) determines a partition of R into various sub-regions whose overall similarity with test sites p_i is maximized.

Districts for Vehicle Routing In a *vehicle districting problem* our objective is to design a collection of sub-regions that minimize the workloads of a fleet of vehicles that provide service to a region [64]. Suppose that, on each day, a set of N demand points $\{X_1, \ldots, X_N\}$ is sampled from a probability distribution $\tilde{f}(\cdot)$ defined on a service region R, and each demand point must be serviced by a vehicle (each of which is originally located at a depot $p_i \in R$). The

workload of a vehicle assigned to cover region R_i is simply the length of a travelling salesman tour of all of the sampled points in R_i plus the depot point p_i, $\mathrm{TSP}(p_i \cup \{X_j\} \cap R_i)$. As shown in [40], it turns out that with probability one we have

$$\mathrm{TSP}(p_i \cup \{X_j\} \cap R_i) \to \beta \iint_{R_i} \sqrt{\tilde{f}_c(x)}\, dA + o(\sqrt{N})$$

as $N \to \infty$, where $\tilde{f}_c(\cdot)$ denotes the absolutely continuous part of $\tilde{f}(\cdot)$. In order to balance the workloads of the vehicles, we should then solve an instance of (2.1) where $f(\cdot) = \sqrt{\tilde{f}_c(x)}$ (normalized so that $f(\cdot)$ integrates to 1) and $q_i = 1/n$, so that all vehicles have the same asymptotic workload. As a utility function it is natural to use $u_i(x) = -\|x - p_i\|$ or $u_i(x) = -\|x - p_i\|^2$ so that our sub-regions are as "compact" as possible. We will see that in fact the optimal sub-regions will always be connected when such functions are chosen.

Police Dragnet Design It has been hypothesized [38, 97] that a *logarithmic* relationship exists between the distance from a criminal's home base to a potential target location and the likelihood that the offender chooses to offend in that location, i.e. that

$$\Pr(\text{criminal strikes at } x | \text{home base at } y) = \max\{a - b \log \|x - y\|, 0\}$$

with $a, b \geq 0$. It follows that if a crime has occurred at a point p_i then the probability that the criminal's home base is located in region R_i is proportional to

$$\iint_{R_i} f(x) \max\{a - b \log \|x - p_i\|, 0\}\, dA$$

where $f(x)$ represents a population density. Thus, given a collection of recent crime locations $\{p_1, \ldots, p_n\}$, one can thus consider the problem of designing police "search regions" to maximize the likelihood of catching the criminals by formulating problem (2.1) with $u_i(x) = \max\{a - b \log \|x - p_i\|, 0\}$. Here we might set $q_i = 1/n$ in the constraint $\iint_{R_i} f(x)\, dA = q_i$ for all i, representing a restriction that each district have an equal population (and thus roughly equal "workloads" for the police investigators).

2.3 Computing Optimal Sub-Regions

In this section we show that the optimal sub-regions for problems (2.1) and (2.2) can be described easily in terms of complementary slackness. For ease of intuition, we give proof sketches based on discretizing the problem; the same result can be derived rigorously using infinite-dimensional vector space optimization theory, specifically Theorem 1 of [83], and we do so in Appendix A.

2.3.1 Sub-Regions for the Max-Sum Problem

In this section we consider the structure of problem (2.1). We provide a proof sketch that characterizes the optimal solutions and refer the reader to Section A.1 of the Appendix for a rigorous proof:

Theorem 2.3.1. *The boundaries between any optimal sub-regions R_i^* and R_j^* to problem (2.1) are of the form*

$$\partial(R_i^*) \cap \partial(R_j^*) \subseteq \left\{x \in R : u_i(x) - u_j(x) = \lambda_i^* - \lambda_j^*\right\}$$

where λ_i^* and λ_j^* are the optimal solutions to the dual problem

$$\begin{aligned} \underset{\boldsymbol{\lambda}}{\text{minimize}} \iint_R f(x) \max_i \{u_i(x) - \lambda_i\}\, dA \qquad & s.t. \\ \mathbf{q}^T \boldsymbol{\lambda} \;&=\; 0\,. \end{aligned}$$

Moreover, if x and i are a point and an index such that $u_i(x) - \lambda_i^* > u_j(x) - \lambda_j^*$ for all $j \neq i$, then $x \in R_i^*$. Thus, the optimal partition $\{R_1^*, \ldots, R_n^*\}$ can be recovered from the optimal dual variables $\lambda_1^*, \ldots, \lambda_n^*$.

Proof sketch. We begin by formulating problem (2.1) as an infinite-dimensional integer program. Setting $I_i(x)$ to be a $\{0, 1\}$-valued function indicating whether

point x is assigned to agent i, we obtain the equivalent formulation

$$\begin{aligned} \underset{I_1(\cdot),\ldots,I_n(\cdot)}{\text{maximize}} \sum_{i=1}^{n} \iint_R f(x)u_i(x)I_i(x)\,dA \qquad & s.t. \\ \iint_R f(x)I_i(x)\,dA &= q_i \quad \forall i \\ \sum_{i=1}^{n} I_i(x) &= 1 \quad \forall x \\ I_i(x) &\in \{0,1\} \quad \forall i, x\,. \end{aligned} \tag{2.3}$$

The linear programming relaxation of (2.3) is given by

$$\begin{aligned} \underset{I_1(\cdot),\ldots,I_n(\cdot)}{\text{maximize}} \sum_{i=1}^{n} \iint_R f(x)u_i(x)I_i(x)\,dA \qquad & s.t. \\ \iint_R f(x)I_i(x)\,dA &= q_i \quad \forall i \\ \sum_{i=1}^{n} I_i(x) &= 1 \quad \forall x \\ I_i(x) &\geq 0 \quad \forall i, x\,. \end{aligned} \tag{2.4}$$

We can discretize the above problem into N grid cells $\square_j$ of area ϵ, where f_j denotes the average value of $f(x)$ on $\square_j$, u_{ij} denotes the average value of $u_i(x)$ on $\square_j$, and z_{ij} denotes the fraction of cell $\square_j$ assigned to agent i, to obtain the approximate formulation

$$\begin{aligned} \underset{Z}{\text{maximize}} \sum_{i=1}^{n}\sum_{j=1}^{N} \epsilon f_j u_{ij} z_{ij} \qquad & s.t. \\ \sum_{j=1}^{N} \epsilon f_j z_{ij} &= q_i \quad \forall i \\ \sum_{i=1}^{n} z_{ij} &= 1 \quad \forall j \\ z_{ij} &\geq 0 \quad \forall i, j\,. \end{aligned} \tag{2.5}$$

The dual problem to (2.5), which has variables $\boldsymbol{\lambda} \in \mathbb{R}^n$ and $\boldsymbol{\varsigma} \in \mathbb{R}^N$, is

$$\begin{aligned} \underset{\boldsymbol{\lambda},\boldsymbol{\varsigma}}{\text{minimize}} \sum_{i=1}^{n} q_i\lambda_i + \sum_{j=1}^{N} \varsigma_j \qquad & s.t. \\ \epsilon f_i\lambda_i + \varsigma_j \;&\geq\; \epsilon f_j u_{ij} \quad \forall i,j\,. \end{aligned}$$

Introducing new variables $\sigma_j := \varsigma_j/(\epsilon f_j)$, we can rewrite the above as

$$\begin{aligned} \underset{\boldsymbol{\lambda},\boldsymbol{\sigma}}{\text{minimize}} \sum_{i=1}^{n} q_i\lambda_i + \sum_{j=1}^{N} \epsilon f_j\sigma_j \qquad & s.t. \\ \sigma_j \;&\geq\; u_{ij} - \lambda_i \quad \forall i,j \end{aligned}$$

which is a discretization of the problem

$$\begin{aligned} \underset{\boldsymbol{\lambda},\sigma(\cdot)}{\text{minimize}} \sum_{i=1}^{n} q_i\lambda_i + \iint_R f(x)\sigma(x)\,dA \qquad & s.t. \\ \sigma(x) \;&\geq\; u_i(x) - \lambda_i \quad \forall i,x \end{aligned} \tag{2.6}$$

which is equivalent to the unconstrained problem

$$\underset{\boldsymbol{\lambda}}{\text{minimize}} \sum_{i=1}^{n} q_i\lambda_i + \iint_R f(x) \max_i \{u_i(x) - \lambda_i\}\; dA\,.$$

Finally, we note that the above problem is invariant under scalar addition to $\boldsymbol{\lambda}$ because we have assumed that $\sum_{i=1}^{n} q_i = \iint_R f(x)\,dA = 1$ and thus we obtain the convex, n-dimensional dual problem

$$\begin{aligned} \underset{\boldsymbol{\lambda}}{\text{minimize}} \iint_R f(x) \max_i \{u_i(x) - \lambda_i\}\; dA \qquad & s.t. \\ \mathbf{q}^T\boldsymbol{\lambda} \;&=\; 0\,. \end{aligned} \tag{2.7}$$

It remains to show that the optimal partition $\{R_1^*, \ldots, R_n^*\}$ for the original problem (2.1) can be recovered from problem (2.7). Let $\{I_1^*(\cdot), \ldots, I_n^*(\cdot)\}$ denote the optimal solution to the LP relaxation (2.4) and consider any point $x \in R$ and the optimal solution $\boldsymbol{\lambda}^*$ to (2.7). Suppose $\bar{i}$ is the index such that $u_i(x) - \lambda_i^*$ is maximal (assuming such an index is unique). From the complementary slackness conditions of problem (2.6), it must be the case that $I_i^*(x) = 0$ for all indices i

other than $\bar{i}$, and consequently that $I^*_{\bar{i}}(x) = 1$. This completes the proof. □

Remark 2.3.2. The dual variables λ_i have a natural interpretation as *shadow prices* associated with the agents; specifically, suppose that a client at point x must pay a fee of $-(u_i(x) - \lambda_i)$ for selecting agent i. Obviously, each client will choose the agent for which his or her fee is minimized. Thus, the dual problem asks us to choose rates λ_i that maximize the overall revenue that the agents receive from the clients, subject to a cap on the total rate at which they are allowed to charge them.

2.3.2 Sub-Regions for the Max-Min Problem

In this section we consider the structure of problem (2.2). Again, we provide a proof sketch here and refer the reader to Section A.2 of the Appendix for a rigorous proof:

Theorem 2.3.3. *Provided that either $u_i(x) > 0$ for all i and x or $u_i(x) < 0$ for all i and x (except for possibly a set of measure zero), the boundaries between any optimal sub-regions R^*_i and R^*_j to problem (2.2) are of the form*

$$\partial(R^*_i) \cap \partial(R^*_j) \subseteq \left\{ x \in R : \frac{u_i(x)}{u_j(x)} = \frac{\lambda^*_j}{\lambda^*_i} \right\}$$

where λ^*_i and λ^*_j are the optimal solutions to the dual problem

$$\begin{aligned} \underset{\boldsymbol{\lambda}}{\text{minimize}} \iint_R f(x) \max_i \{\lambda_i u_i(x)\}\, dA \qquad & s.t. \\ \sum_{i=1}^{n} \lambda_i &= 1 \\ \lambda_i &\geq 0 \quad \forall i\,. \end{aligned}$$

Moreover, if x and i are a point and an index such that $\lambda^*_i u_i(x) > \lambda^*_j u_j(x)$ for all $j \neq i$, then $x \in R^*_i$. Thus, the optimal partition $\{R^*_1, \ldots, R^*_n\}$ can be recovered from the optimal dual variables $\lambda^*_1, \ldots, \lambda^*_n$, and at optimality, it turns out that all sub-regions have the exact same utility $\iint_{R^*_i} f(x) u_i(x)\, dA$.

Proof sketch. We can similarly formulate problem (2.2) as an infinite-dimensional

integer program given by

$$\begin{aligned} \underset{t,I_1(\cdot),\ldots,I_n(\cdot)}{\text{maximize}}\ t \qquad & s.t. \\ t &\leq \iint_R f(x)u_i(x)I_i(x)\,dA \quad \forall i \\ \sum_{i=1}^{n} I_i(x) &= 1 \quad \forall x \\ I_i(x) &\in \{0,1\} \quad \forall i, x\,. \end{aligned} \tag{2.8}$$

The discretization of the linear programming relaxation of (2.8) is given by

$$\begin{aligned} \underset{t,Z}{\text{maximize}}\ t \qquad & s.t. \\ t &\leq \sum_{j=1}^{N} \epsilon f_j u_{ij} z_{ij} \quad \forall i \\ \sum_{i=1}^{n} z_{ij} &= 1 \quad \forall j \\ z_{ij} &\geq 0 \quad \forall i, j\,. \end{aligned} \tag{2.9}$$

The dual problem to (2.9), which has variables $\boldsymbol{\mu} \in \mathbb{R}^n$ and $\boldsymbol{\varsigma} \in \mathbb{R}^N$, is

$$\begin{aligned} \underset{\boldsymbol{\mu},\boldsymbol{\varsigma}}{\text{minimize}} \sum_{j=1}^{N} \varsigma_j \qquad & s.t. \\ \epsilon f_j u_{ij} \mu_i + \varsigma_j &\geq 0 \quad \forall i, j \\ -\sum_{i=1}^{n} \mu_i &= 1 \\ \mu_i &\leq 0 \quad \forall i, j\,. \end{aligned}$$

Introducing new variables $\lambda_i := -\mu_i$ and $\sigma_j := \varsigma_j/(\epsilon f_j)$, we can rewrite the

above as

$$\begin{aligned} \underset{\boldsymbol{\lambda},\boldsymbol{\sigma}}{\text{minimize}} \sum_{j=1}^{N} \epsilon f_j \sigma_j \qquad & s.t. \\ \sigma_j &\geq u_{ij}\lambda_i \quad \forall i,j \\ \sum_{i=1}^{n} \lambda_i &= 1 \\ \lambda_i &\geq 0 \quad \forall i,j \end{aligned}$$

which is a discretization of the problem

$$\begin{aligned} \underset{\boldsymbol{\lambda},\sigma(\cdot)}{\text{minimize}} \iint_R f(x)\sigma(x)\,dA \qquad & s.t. \\ \sigma(x) &\geq \lambda_i u_i(x) \quad \forall i,x \\ \sum_{i=1}^{n} \lambda_i &= 1 \\ \lambda_i &\geq 0 \quad \forall i \end{aligned}$$

which is equivalent to the convex problem

$$\begin{aligned} \underset{\boldsymbol{\lambda}}{\text{minimize}} \iint_R f(x) \max_i\{\lambda_i u_i(x)\}\,dA \qquad & s.t. \\ \sum_{i=1}^{n} \lambda_i &= 1 \\ \lambda_i &\geq 0 \quad \forall i\,. \end{aligned} \tag{2.10}$$

As before, let $\{I_1^*(\cdot),\ldots,I_n^*(\cdot)\}$ denote the optimal solution to the LP relaxation of (2.8) and consider any point $x \in R$ and the optimal solution $\boldsymbol{\lambda}^*$ to (2.10). Again, if the index $\bar{i}$ that maximizes $\lambda_i u_i(x)$ is unique, it must be the case that $I_{\bar{i}}^*(x) = 1$ and $I_i^*(x) = 0$ for all other i. In addition, by our initial assumption that $u_i(x) > 0$ for all i and x or $u_i(x) < 0$ for all i and x (except for possibly a set of measure zero), it is not hard to show (by a simple perturbation argument) that we must have $\lambda_i^* > 0$ for all i. Complementary slackness then tells us that $t^* = \iint_R f(x)u_i(x)I_i^*(x)\,dA$ for all i at optimality, i.e. the sub-regions all have the exact same utility. □

Remark 2.3.4. The dual variables λ_i again have a natural interpretation as shadow prices associated with the agents; specifically, suppose that a client

at point x must pay a fee of $-\lambda_i u_i(x)$ for selecting agent i. Obviously, each client will choose the agent for which his or her fee is minimized. Again, the dual problem asks us to choose rates λ_i that maximize the overall revenue that the agents receive from the clients, subject to a cap on the total rate at which they are allowed to charge them.

2.4 Sub-Gradients

In this section we show that problems (2.1) and (2.2) can be solved efficiently using convex optimization. Specifically, we show that *subgradients* [29] to the dual problems (2.7) and (2.10) are cheap to compute, and therefore the optimal partition can be computed quickly using, for example, an *analytic center cutting plane method* [28].

2.4.1 Computing Subgradients for Max-Sum Problem

It is straightforward to verify that the vector $\mathbf{g} \in \mathbb{R}^n$, defined by setting

$$g_i := -\iint_{R_i} f(x)\, dA\,,$$

is a subgradient for the objective function

$$h(\boldsymbol{\lambda}) := \iint_R f(x) \max_i \{u_i(x) - \lambda_i\}\; dA$$

for the dual problem (2.7). To see this, consider two vectors $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}'$ and the corresponding partitions $\{R_1, \ldots, R_n\}$ and $\{R'_1, \ldots, R'_n\}$. We want to show that $h(\boldsymbol{\lambda}') \geq h(\boldsymbol{\lambda}) + \mathbf{g}^T(\boldsymbol{\lambda}' - \boldsymbol{\lambda})$, i.e. that

$$\iint_R f(x) \max_i \left\{u_i(x) - \lambda'_i\right\}\; dA \geq \iint_R f(x) \max_i \{u_i(x) - \lambda_i\}\; dA + \mathbf{g}^T(\boldsymbol{\lambda}' - \boldsymbol{\lambda})$$

or equivalently that

$$\iint_R f(x) \max_i \left\{u_i(x) - \lambda'_i\right\}\; dA \geq \sum_{i=1}^{n} \iint_{R_i} f(x)(u_i(x) - \lambda_i)\, dA + g_i(\lambda'_i - \lambda_i)\,.$$

Consider the right-hand side of the above; for each i, we have

$$\begin{aligned}\iint_{R_i} f(x)(u_i(x)-\lambda_i)\,dA &+ g_i(\lambda_i^{'}-\lambda_i)\\ &= \iint_{R_i} f(x)(u_i(x)-\lambda_i)\,dA - (\lambda_i^{'}-\lambda_i)\iint_{R_i} f(x)\,dA\\ &= \iint_{R_i} f(x)(u_i(x)-\lambda_i^{'})\,dA\end{aligned}$$

and therefore we have

$$h(\boldsymbol{\lambda}^{'}) = \underbrace{\iint_R f(x)\max_i\left\{u_i(x)-\lambda_i^{'}\right\}\,dA \geq \sum_{i=1}^{n}\iint_{R_i} f(x)(u_i(x)-\lambda_i^{'})\,dA}_{(*)}$$
$$= h(\boldsymbol{\lambda}) + \mathbf{g}^T(\boldsymbol{\lambda}^{'}-\boldsymbol{\lambda})$$

as desired, where $(*)$ follows from the fact that, by construction, the sub-regions of the partition $\{R_1^{'},\ldots,R_n^{'}\}$ are defined by taking the maximal value of $u_i(x)-\lambda_i^{'}$ and are therefore maximal over all partitions. This completes the proof.

2.4.2 Computing Subgradients for Max-Min Problem

It is straightforward to verify that the vector $\mathbf{g}\in\mathbb{R}^n$, defined by setting

$$g_i := \iint_{R_i} f(x)u_i(x)\,dA\,,$$

is a subgradient for the objective function

$$h(\boldsymbol{\lambda}) := \iint_R f(x)\max_i\{\lambda_i u_i(x)\}\,dA$$

for the dual problem (2.10). To see this, consider two vectors $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}^{'}$ and the corresponding induced partitions $\{R_1,\ldots,R_n\}$ and $\{R_1^{'},\ldots,R_n^{'}\}$. We want to show that $h(\boldsymbol{\lambda}^{'})\geq h(\boldsymbol{\lambda})+\mathbf{g}^T(\boldsymbol{\lambda}^{'}-\boldsymbol{\lambda})$, i.e. that

$$\iint_R f(x)\max_i\{\lambda_i^{'}u_i(x)\}\,dA \geq \iint_R f(x)\max_i\{\lambda_i u_i(x)\}\,dA + \mathbf{g}^T(\boldsymbol{\lambda}^{'}-\boldsymbol{\lambda})$$

or equivalently that

$$\iint_R f(x) \max_i \{\lambda_i^{'} u_i(x)\}\, dA \geq \sum_{i=1}^{n} \iint_{R_i} \lambda_i f(x) u_i(x)\, dA + g_i(\lambda_i^{'} - \lambda_i)\,.$$

Consider the right-hand side of the above; for each i, we have

$$\begin{aligned} \iint_{R_i} \lambda_i f(x) u_i(x)\, dA \;\; &+ \;\; g_i(\lambda_i^{'} - \lambda_i) \\ &= \iint_{R_i} \lambda_i f(x) u_i(x)\, dA + (\lambda_i^{'} - \lambda_i) \iint_{R_i} f(x) u_i(x)\, dA \\ &= \iint_{R_i} \lambda_i^{'} f(x) u_i(x)\, dA \end{aligned}$$

and therefore we have

$$h(\boldsymbol{\lambda}^{'}) = \underbrace{\iint_R f(x) \max_i \{\lambda_i^{'} u_i(x)\}\, dA \geq \sum_{i=1}^{n} \iint_{R_i} \lambda_i^{'} f(x) u_i(x)\, dA}_{(*)}$$

$$= h(\boldsymbol{\lambda}) + \mathbf{g}^T(\boldsymbol{\lambda}^{'} - \boldsymbol{\lambda})$$

as desired, where $(*)$ follows from the fact that, by construction, the sub-regions of the partition $\{R_1^{'}, \ldots, R_n^{'}\}$ are defined by taking the maximal value of $\lambda_i u_i(x)$ and are therefore maximal over all partitions. This completes the proof.

2.5 Algorithms

For the sake of completeness, we give formal descriptions of two algorithms, MaxSumPartition and MaxMinPartition, that solve problems (2.1) and (2.2), in Algorithms 2.5.1 and 2.5.2.

Remark 2.5.1. We would like to point out two advantages that Algorithms 2.5.1 and 2.5.2 possess over existing methods. First, we compute $\boldsymbol{\lambda}^*$ by solving a *convex* optimization problem for which subgradients are cheap to compute. Therefore, these approaches inherit better theoretical convergence properties than, say, the scheme of [99, 100], which uses a gradient descent method for the case where $u_i(x) = -\|x - p_i\|^2$ and whose associated objective function (i.e. the equivalent of $h(\boldsymbol{\lambda})$) is not convex (although the scheme proposed therein is

Input: A connected, polygonal region R with non-empty interior, a probability density function $f(\cdot)$ defined on R, a collection of n utility density functions $u_i(\cdot)$, a vector $\mathbf{q} \in \mathbb{R}^n_+$ such that $\sum_i q_i = 1$, and a threshold ϵ.

Output: A partition of R into n regions $R_1, \ldots, R_n$ that solves problem (2.1) within tolerance ϵ.

Note: this is simply a standard analytic center cutting plane method applied to problem (2.7).

Define the initial polyhedron by $\Lambda = \left\{\boldsymbol{\lambda} \in \mathbb{R}^n : \mathbf{q}^T\boldsymbol{\lambda} = 0 \text{ and } \|\boldsymbol{\lambda}\|_\infty \leq M\right\}$ for a threshold M;

`/* See Lemma A.1.1 of the Appendix to see how to construct a suitable` M. `*/`

while $\text{vol}(\Lambda) > \epsilon$ **do**

  Let $\boldsymbol{\lambda}^0$ be the analytic center of Λ;

  for $i \in \{1, \ldots, n\}$ **do**

    Let R_i denote the sub-region in R for which $u_i(x) - \lambda^0_i$ is strictly maximal;

  end

  Allocate the remaining mass of R (i.e. that which has not been assigned to a subset R_i, if any) lexicographically;

  `/* This lexicographic allocation will not generally be feasible for the original partitioning problem. */`

  for $i \in \{1, \ldots, n\}$ **do**

    Set $g_i := -\iint_{R_i} f(x)\, dA$;

  end

  Set $\Lambda := \Lambda \cap \{\boldsymbol{\lambda} : \mathbf{g}^T\boldsymbol{\lambda} \geq \mathbf{g}^T\boldsymbol{\lambda}^0\}$;

end

Let $\boldsymbol{\lambda}^0$ be the analytic center of Λ;

for $i \in \{1, \ldots, n\}$ **do**

  Let R_i denote the sub-region in R for which $u_i(x) - \lambda^0_i$ is strictly maximal;

end

Allocate the remaining mass of R (i.e. that which has not been assigned to a subset R_i, if any) as described in Section B.1 of the Appendix;

return $\{R_1, \ldots, R_n\}$;

Algorithm 2.5.1: Algorithm MaxSumPartition partitions a given region into sub-regions with pre-specified masses while maximizing the overall utility of n agents.

Input: A connected, polygonal region R with non-empty interior, a probability density function $f(\cdot)$ defined on R, a collection of n utility density functions $u_i(\cdot)$, and a threshold ϵ.

Output: A partition of R into n regions $R_1, \ldots, R_n$ that solves problem (2.2) within tolerance ϵ.

Note: this is simply a standard analytic center cutting plane method applied to problem (2.10).

Define the initial polyhedron by $\Lambda = \{\boldsymbol{\lambda} \in \mathbb{R}^n : \sum_i \lambda_i = 1 \text{ and } \boldsymbol{\lambda} \geq \mathbf{0}\}$;
while $\text{vol}(\Lambda) > \epsilon$ **do**
  Let $\boldsymbol{\lambda}^0$ be the analytic center of Λ;
  for $i \in \{1, \ldots, n\}$ **do**
    Let R_i denote the sub-region in R for which $\lambda_i^0 u_i(x)$ is strictly maximal;
  end
  Allocate the remaining mass of R (i.e. that which has not been assigned to a subset R_i, if any) lexicographically;
  for $i \in \{1, \ldots, n\}$ **do**
    Set $g_i := \iint_{R_i} f(x)u_i(x)\, dA$;
  end
  Set $\Lambda := \Lambda \cap \{\boldsymbol{\lambda} : \mathbf{g}^T\boldsymbol{\lambda} \geq \mathbf{g}^T\boldsymbol{\lambda}^0\}$;
end
Let $\boldsymbol{\lambda}^0$ be the analytic center of Λ;
for $i \in \{1, \ldots, n\}$ **do**
  Let R_i denote the sub-region in R for which $\lambda_i^0 u_i(x)$ is strictly maximal;
end
Allocate the remaining mass of R (i.e. that which has not been assigned to a subset R_i, if any) as described in Section B.2 of the Appendix;
return $\{R_1, \ldots, R_n\}$;

Algorithm 2.5.2: Algorithm MaxMinPartition partitions a given region into sub-regions while maximizing the minimum utility of n agents.

proven to be globally convergent). Secondly, the scheme of [99, 100] requires an explicit expression for the boundary components of the power cells R_i (which are line segments), while ours merely requires that we be able to integrate over the R_i, say using Monte Carlo or quasi-Monte Carlo approximation. This is not an issue when $R \subset \mathbb{R}^2$ (since explicitly representing the boundary components – curves or line segments – is not difficult), but when $R \subset \mathbb{R}^d$ for high d, these boundary components may be difficult to enumerate. Monte Carlo and quasi-Monte Carlo integration methods, on the other hand, do not suffer from this "curse of dimensionality" because they involve merely sampling a large collection of points in R and estimating integrals by working with those points that lie in each region R_i; the convergence rate of the integral is inversely proportional to the square root of the number of samples $\sqrt{N}$. Of course, such methods bring with them their own drawbacks, as described in Section 9.9.6 of [107]:

> Such a convergence rate *does not* depend on the dimension n of the integration domain, and this is a most relevant feature of the Monte Carlo method. However, it is worth noting that the convergence rate is independent of the *regularity* of f; thus, unlike interpolatory quadratures, Monte Carlo methods *do not* yield more accurate results when dealing with smooth integrands.
>
> [The convergence rate of $1/\sqrt{N}$] is extremely weak and in practice one does often obtain poorly accurate results. A more efficient implementation of Monte Carlo methods is based on composite approach or semi-analytical methods; an example of these techniques is provided in [88], where a composite Monte Carlo method is employed for the computation of integrals over hypercubes in $\mathbb{R}^n$.

Given a sample $x \in R$, we can easily determine the region R_i containing x by merely checking which index i maximizes either $u_i(x) - \lambda_i$ or $\lambda_i u_i(x)$ (depending on what problem we are solving). Of course, we will eventually desire some kind of expression for the boundary components of the optimal regions $R_1^*, \cdots, R_n^*$, be it implicit or explicit.

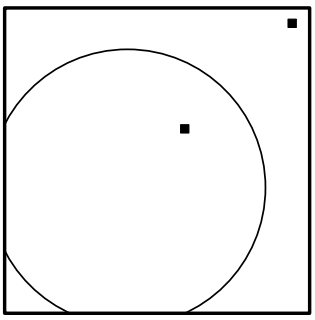


Figure 2.1: The partition shown above has a disconnected sub-region.

2.6 Applications Revisited

Here we discuss the implications of Theorems 2.3.1 and 2.3.3 on the applications listed in Section 2.2.

Facility Districting The formulation (2.2) deals with the problem of designing service districts while minimizing the maximum workload of a facility. Boundaries of the optimal partitions are given by levels sets of:

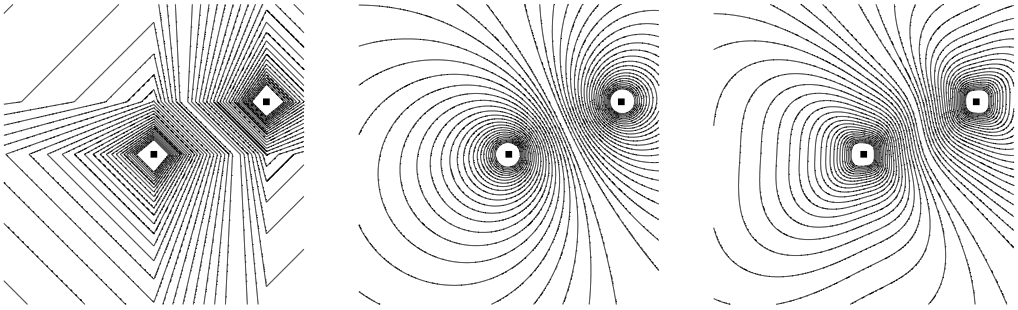
$$\frac{\|x - p_i\|}{\|x - p_j\|}.$$

From the theorem of Apollonius [93], these are just a collection of circular arcs (Apollonian circles), which in turn means that the optimal partition is a multiplicatively weighted Voronoi diagram.

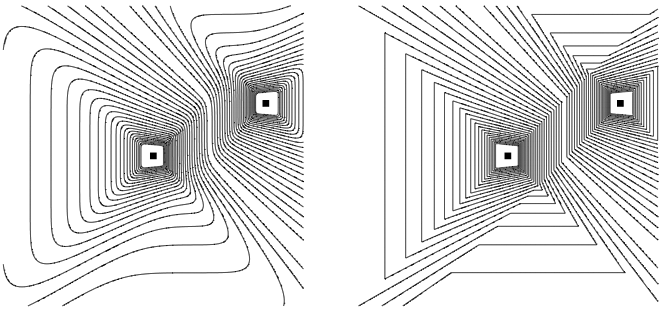
Remark 2.6.1. One drawback to our formulation is that we have not imposed connectivity between regions; indeed, an optimal multiplicatively weighted Voronoi diagram need not be connected, as shown in Figure 2.1. We will address this in Chapter 3, where we give two heuristic methods to impose connectivity among sub-regions.

Remark 2.6.2. Note that, the theorm of Apollonius (that the points x for which $\lambda_i^* \|x - p_i\| = \lambda_j^* \|x - p_j\|$ are circular arcs) applies only to the Euclidean norm $\|\cdot\|_2$. Some Apollonian curves for other p-norms are shown in Figure 2.2.

One other way of balancing workload among facilities is by minimizing the aggregate workload over all facilities with an equal area constraint. This is same as problem (2.1) when $u_i(x) = -\|x - p_i\|$. Here, the optimal boundaries are



(a) $p = 1$ (b) $p = 2$ (c) $p = 3$



(d) $p = 10$ (e) $p = \infty$

Figure 2.2: Apollonian curves for various p-norms.

level sets of

$$\|x - p_i\| - \|x - p_j\|,$$

which are *hyperbolic* arcs. The optimal partition is an additively weighted Voronoi diagram and it turns out that the sub-regions are *star-convex*; that is, if point x is assigned to facility p_i, then so is every point x' on the segment connecting x and p_i. This is true for the same reason that an optimal (bipartite) Euclidean matching has no crossing edges.

Remark 2.6.3. It is not hard to see that the primal problem (2.1) when $u_i(x) = -\|x - p_i\|$, is a special "mixed" case of the Monge-Kantorovich transportation problem [121]: our objective is to "transport" the continuously distributed demand to the finite collection of facilities, while obeying capacity constraints and minimizing the aggregate transportation cost.

Power Diagrams By applying Theorem 2.3.1 to the case where $u_i(x) = -\|x - p_i\|^2$ for points $p_i \in R$, we find concise proofs of Theorems 1 and 3 and Corollary 1 from [11] (and equivalently Theorem 4.1 from [99] and Theorem 3.1 from [100]), all of which say (in one form or another) that the boundaries between optimal sub-regions to problem (2.1) are a power diagram, and therefore for any set of points $\{p_1, \ldots, p_n\} \subset R$ and any constraint vector $\mathbf{q}$ with $q_i \geq 0$ for all i and $\sum_i q_i = 1$, there exists a power diagram based at the points p_i such that $\iint_{C_i} f(x)\, dA = q_i$ for each cell C_i. As Algorithm 2.5.1 shows, it is simple to compute the optimal weight vector $\boldsymbol{\lambda}^*$ via the formulation (2.7), which improves over the control policy proposed in [99, 100] (which also provably converges to $\boldsymbol{\lambda}^*$) as mentioned in Remark 2.5.1. Note that, at optimality, it may be the case that $p_i \notin R_i^*$ (since the cells in a power diagram are not guaranteed to contain their associated point).

An additional observation is that the boundaries between optimal sub-regions to problem (2.2) are level sets of the function $\|x - p_i\|^2 / \|x - p_j\|^2$ which are always arcs of a circle (specifically, an "Apollonian circle" [94]), as shown in Figure 2.3.

Fair Division Consider the fair division problem (2.2) where each utility function $u_i(x)$ is a bivariate normal distribution with mean μ_i and covariance matrix Σ_i. Taking logarithms, it is easy to see that the boundaries between optimal

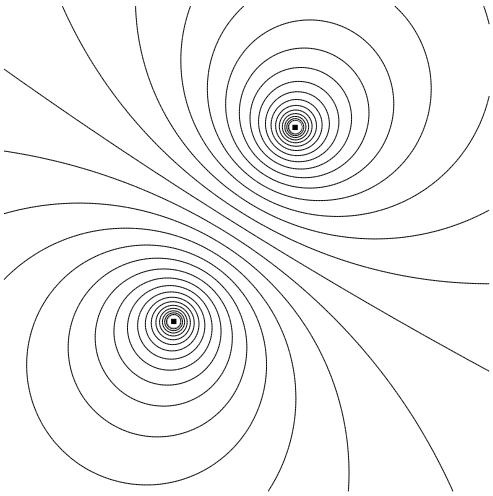


Figure 2.3: Apollonian circles: the level sets of the function $\|x - p_i\|^2 / \|x - p_j\|^2$.

sub-regions are level sets of the function

$$(x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i) - (x - \mu_j)^T \Sigma_j^{-1} (x - \mu_j)\,, \tag{2.11}$$

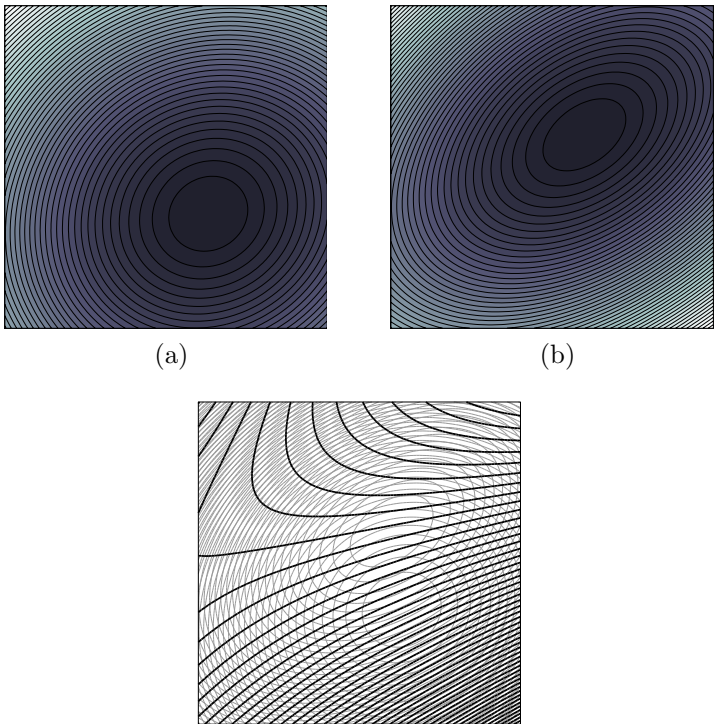
which are conic sections, as shown in Figure 2.4. It is worth mentioning that the sub-regions may not be connected.

Maximizing Influence Theorem 2.3.1 tells us that when $u_i(x) = 1/\|x - p_i\|^2$, the boundaries between optimal sub-regions to problem (2.1) are quartic curves that are level sets of the function

$$\frac{1}{\|x - p_i\|^2} - \frac{1}{\|x - p_j\|^2}\,.$$

We are not aware of any name given to such curves, although they can easily be parameterized by expressing them in bipolar coordinate form and then applying a Euclidean transformation [81]. Such curves are shown in Figure 2.5. When we consider problem (2.2), it is easy to see that the boundaries between optimal sub-regions are Apollonian circles again.

Hospital Districting When we consider problem (2.2) with $u_i(x) = \exp(-\|x - p_i\|)$, it is easy to see that the boundaries between optimal sub-regions are simply hy-



(a) (b)

(c)

Figure 2.4: In (2.4a) and (2.4b) we have two normal distributions and in (2.4c) we show the level sets of the function (2.11); the light gray curves are the level sets of the normal distributions.

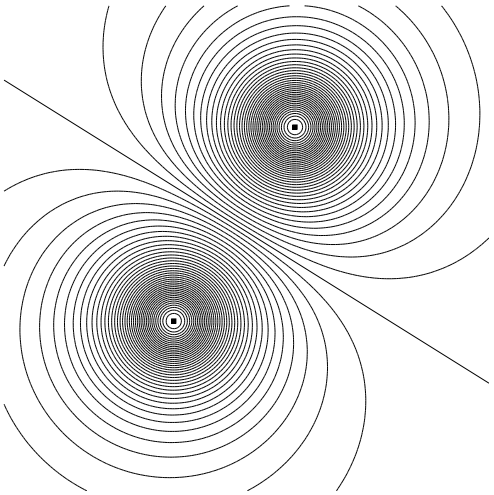
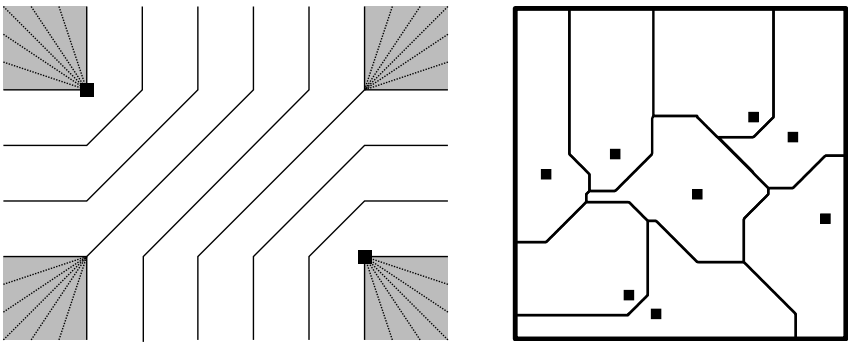


Figure 2.5: Level sets of the function $1/\|x - p_i\|^2 - 1/\|x - p_j\|^2$.



(a) (b)

Figure 2.6: In (2.6a) we show the level sets of the function $\|x-p_i\|_1 - \|x-p_j\|_1$ and in (2.6b) we show an optimal solution to (2.1) on the unit square where $f(\cdot)$ is the uniform distribution, $q_i = 1/n$ for all n, and $u_i(x) = -\|x-p_i\|_1$. The problem of recovering an optimal partition given the dual variables λ_i^* is not entirely trivial because the boundary components may not be one-dimensional (i.e. the shaded regions in (2.6a)), although a partition satisfying the necessary properties can indeed be recovered as explained in Section B.1 of the online supplement.

perbolas, that is, the level sets of

$$\|x - p_i\| - \|x - p_j\| \ .$$

Districts for Vehicle Routing As we have seen in the preceding examples, in the vehicle districting problem as formulated previously, we are guaranteed to have either hyperbolic arcs (when $u_i(x) = -\|x - p_i\|$) or straight lines (when $u_i(x) = -\|x-p_i\|^2$) as boundary components, and in either case the optimal sub-regions are guaranteed to be at least connected (if not convex). If we insist that our boundary components be line segments (for the appearance of simplicity, for example) but we also want $p_i \in R_i^*$ for all i (which does not necessarily hold for a power diagram), another possibility is to use $u_i(x) = -\|x - p_i\|_1$ or $u_i(x) = -\|x - p_i\|_\infty$; the resulting boundary components are shown in Figure 2.6.

Police Dragnet Design When $u_i(x) = \max\{a - b\log\|x - p_i\|, 0\}$, it is easy to verify that the boundaries between optimal sub-regions to problem (2.1) are Apollonian circles, by the same arguments put forth earlier in this section.

2.7 Computational Complexity

Up to this point we have described how to solve the dual problems (2.7) and (2.10) efficiently by constructing subgradient vectors. However, we have not yet discussed the added computational complexity incurred by evaluating these integrals numerically. To this end, we find the following result (originally given in Section 7.4 of [70], but re-stated using the language of Section 9.9 of [107]) useful:

Theorem 2.7.1. *Let $\Omega \subset \mathbb{R}^2$ be a domain of integration equipped with a triangulation $\mathcal{T}_h$ consisting of N_T triangles, where h is the maximum edge length in $\mathcal{T}_h$. There exists a positive constant K_1, independent of h, such that the error E induced using either the composite midpoint formula*

$$\iint_\Omega f(x)\,dA \approx \sum_{T\in\mathcal{T}_h} \text{Area}(T) f(\text{centroid}(T))$$

or the composite trapezoidal formula

$$\iint_\Omega f(x)\,dA \approx \frac{1}{3}\sum_{T\in\mathcal{T}_h} \text{Area}(T) \sum_{j=1}^{3} f(\text{vertex}_j(T))$$

is bounded by

$$|E| \leq K_1 h^2\,\text{Area}(\Omega) M_2\ ,$$

where M_2 is the maximum value of the modules of the second derivatives of the integrand $f(\cdot)$.

It follows that, if our desired error in integration is ϵ in our problem, then the maximum length of any edge in the triangulation must be at most $\epsilon^{1/2}(\text{Area}(R)M_2K_1)^{-1/2}$ (since we often have curved arcs separating the R_i, an exact triangulation is impossible, but the added computational complexity therein is beyond the scope of this work). If we break R into N_T triangles, the maximum edge length will generally be $\mathcal{O}(\sqrt{\text{Area}(R)/N_T})$ and we therefore need to break R into $\mathcal{O}(\text{Area}(R)^2 M_2/\epsilon)$ triangles. Thus, the complexity of evaluating the subgradient vectors in Section 2.4 is quadratic in $\text{Area}(R)$, linear in the maximum modules of the second derivatives of $f(\cdot)$, and inversely proportional to the desired precision ϵ.

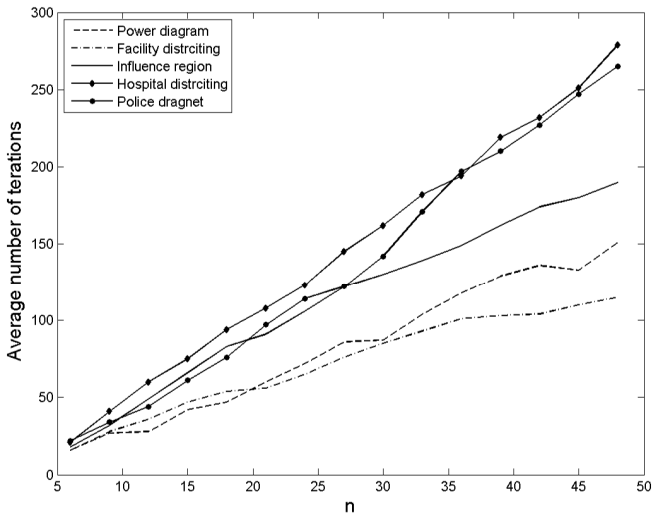
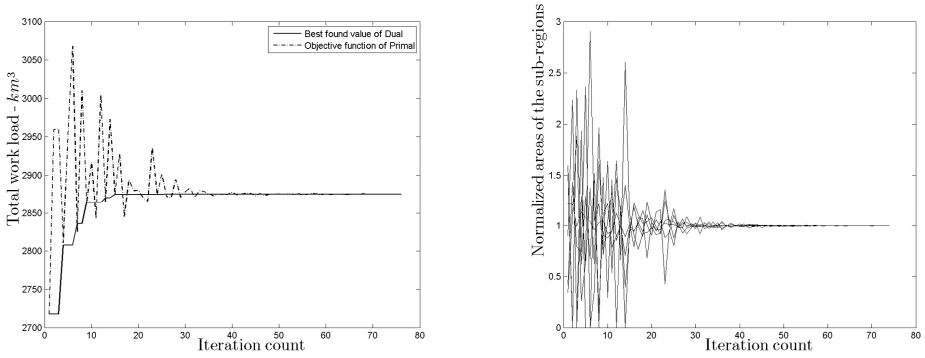


Figure 2.7: The average number of cutting plane method iterations for various utility functions.

2.8 Computational Experiments

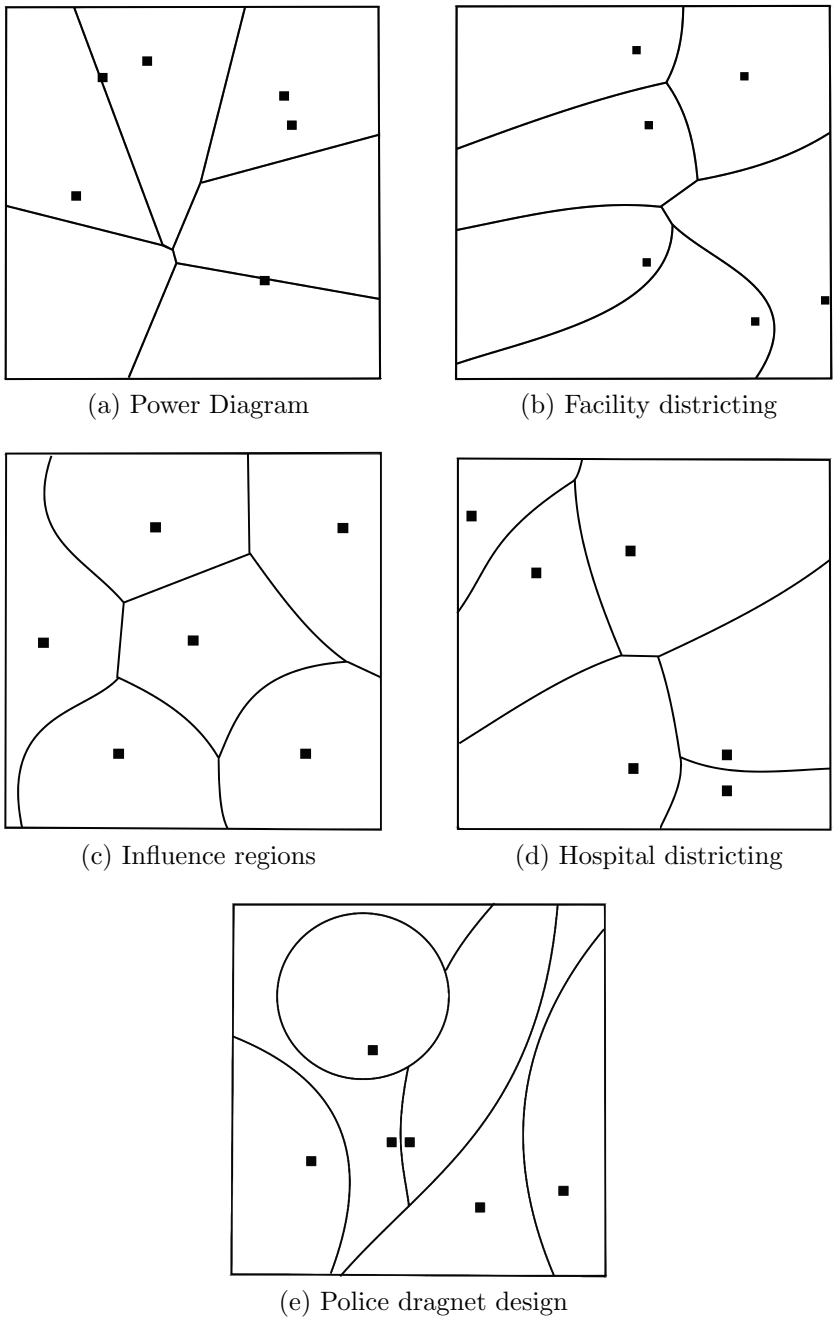
In what follows we solve various instances of (2.1) and (2.2) when R is either the unit square or a geographic map, and $f(\cdot)$ is either the uniform distribution or a population density. We provide the results of two numerical simulations: in the first simulation, R is the unit square and $f(\cdot)$ is the uniform distribution; in the second, R is a map of Ramsey County, Minnesota, and $f(\cdot)$ is a population density. For all problems we use $q_i = 1/n$ for all i; the agents' locations are randomly chosen. Figure 2.7 shows the number of cutting plane iterations (averaged over 5 random samples) required for $n = 6$ through $n = 50$ for the unit square. Figures 2.8 and 2.10 show the convergence of our algorithm for $n = 6$ in the unit square and $n = 7$ in Ramsey County from both a primal and dual perspective. Figures 2.9 and 2.11 show the various optimal partitions that are computed by our algorithm. For all simulations we used a tolerance threshold of 1%.



(a) Objective function - unit square

(b) Normalized areas of the partitions - unit square

Figure 2.8: Convergence of the analytic center cutting plane algorithm for the unit square.



(a) Power Diagram

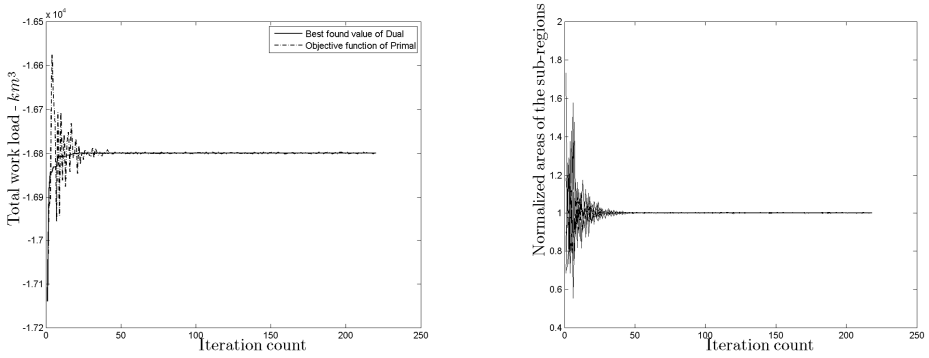
(b) Facility districting

(c) Influence regions

(d) Hospital districting

(e) Police dragnet design

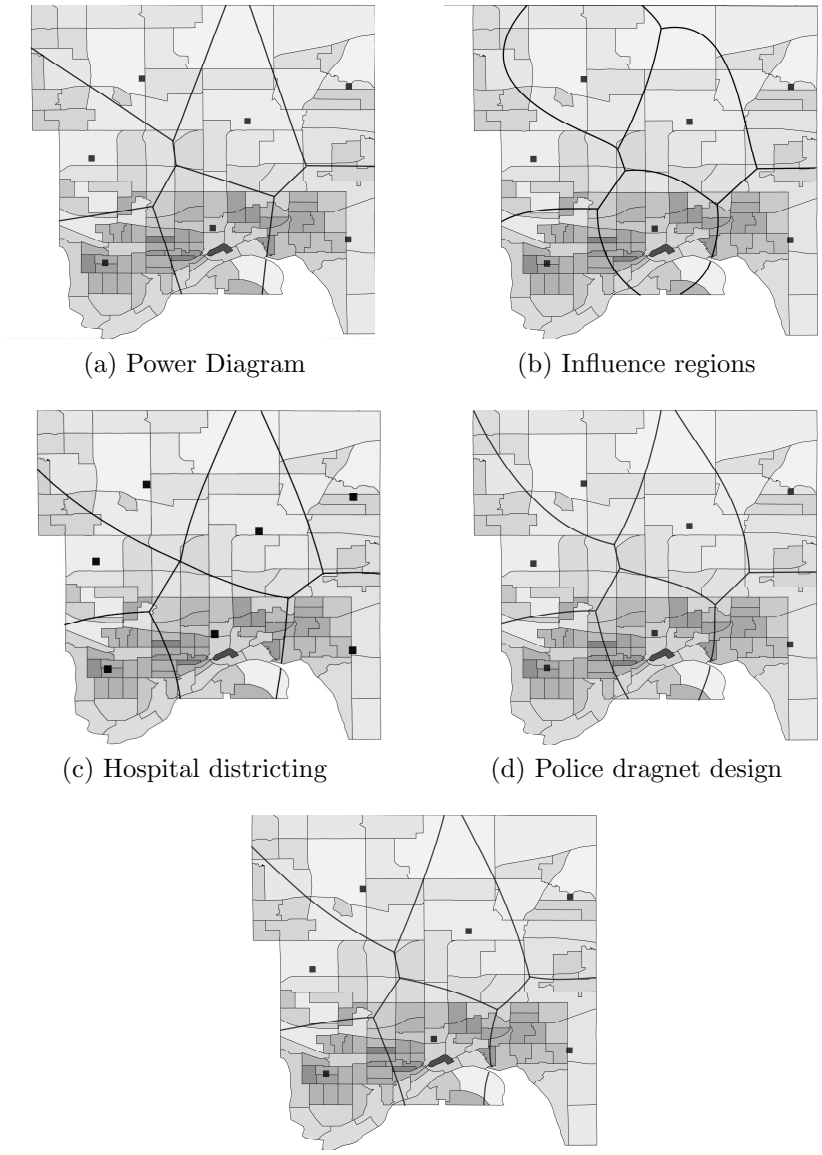
Figure 2.9: Partitioning the unit square into equal area sub-regions for various utility functions.



(a) Objective function

(b) Normalized area of the partitions - Ramsey county

Figure 2.10: Performance of the analytic center cutting plane algorithm for Ramsey county.



(a) Power Diagram

(b) Influence regions

(c) Hospital districting

(d) Police dragnet design

(e) Facility districting

Figure 2.11: Partitioning Ramsey County for various utility functions.

Chapter 3

Enforcing Shape Properties and Dynamic Partitioning

3.1 Enforcing Shape Properties

The map segmentation algorithms 2.5.1 and 2.5.2 described in Chapter 2 do not always ensure that the sub-regions will be connected. We also have not put a bound on the maximum distance from a facility to a point assigned to it. In practice these are both clearly desirable properties and we show that they can be enforced using penalty functions or additional constraints.

3.1.1 Enforcing Connectivity

We have seen previously that the optimal partition to an instance of (2.1) or (2.2) may not be connected. In practice, connectivity is clearly a desirable property; for example, by state constitution, statute, or guideline, 23 US states have passed rulings that their congressional districts must be contiguous [78]. Having observed that the optimal solution to problem (2.1) is always connected when $u_i(x) = -\|x - p_i\|$, one way to enforce connectivity of sub-regions is to augment problems (2.1) or (2.2) with a *penalty term* $-\|x - p_i\|$, giving the

modified problems

$$\begin{aligned}\underset{R_1,\ldots,R_n}{\text{maximize}}\,(1-\mu)\sum_{i=1}^{n}\iint_{R_i} f(x)u_i(x)\,dA \;&-\; \mu\sum_{i=1}^{n}\iint_{R_i} f(x)\|x-p_i\|\,dA \;\; s.t.\\ \iint_{R_i} f(x)\,dA \;&=\; q_i \quad \forall i \\ R_i \cap R_j \;&=\; \emptyset \quad \forall i \neq j \\ \bigcup_i R_i \;&=\; R\end{aligned} \tag{3.1}$$

and

$$\begin{aligned}\underset{R_1,\ldots,R_n}{\text{maximize}}\,(1-\mu)\min_i\left\{\iint_{R_i} f(x)u_i(x)\,dA\right\} \;&-\; \mu\sum_{i=1}^{n}\iint_{R_i} f(x)\|x-p_i\|\,dA \;\; s.t.\\ R_i \cap R_j \;&=\; \emptyset \quad \forall i \neq j \\ \bigcup_i R_i \;&=\; R\,.\end{aligned} \tag{3.2}$$

It is obvious that problem (3.1) is itself merely an instance of (2.1) (with a modified utility function) and thus its boundary components may be solved using duality as before. It is not hard to show that the dual to problem (3.2) is

$$\begin{aligned}\underset{\boldsymbol{\lambda}}{\text{minimize}} \iint_R f(x)\max_i\{\lambda_i u_i(x)-\mu\|x-p_i\|\}\,dA \qquad & s.t.\\ \sum_i \lambda_i \;&=\; 1-\mu \\ \lambda_i \;&\geq\; 0 \quad \forall i\end{aligned}$$

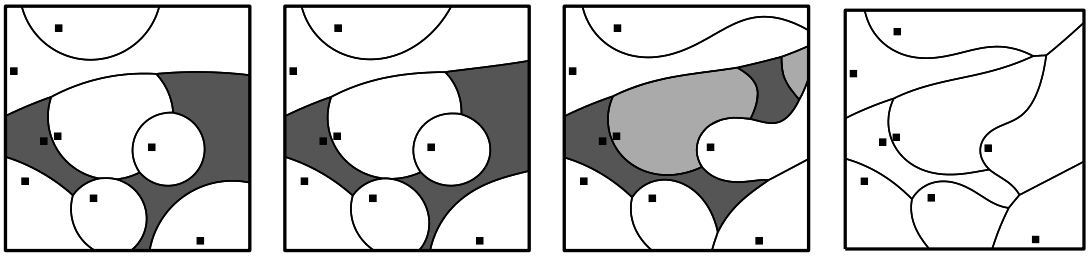
and consequently the optimal boundaries to (3.2) must satisfy

$$\lambda_i u_i(x) - \mu\|x-p_i\| = \lambda_j u_j(x) - \mu\|x-p_j\|\,.$$

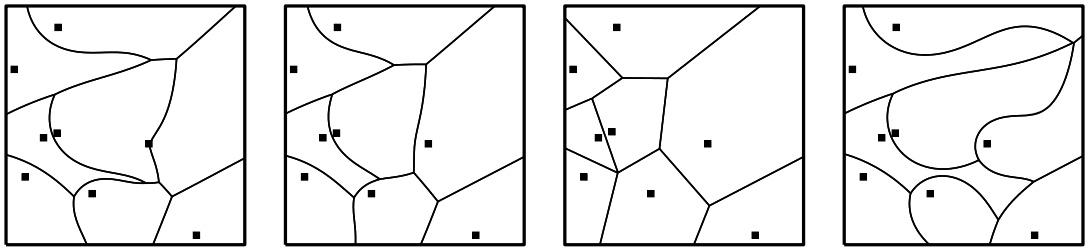
Figure 3.1 shows the effect of this penalty term on an instance of (3.2) where we use the gravity model utility functions given by $u_i(x) = 1/\|x-p_i\|^2$.

3.1.2 Diameter Constraint

We may also impose a constraint on the maximum distance r between a point x and its assigned facility. We use the problem (2.2) with $u_i(x) = -\|x-p_i\|$ for

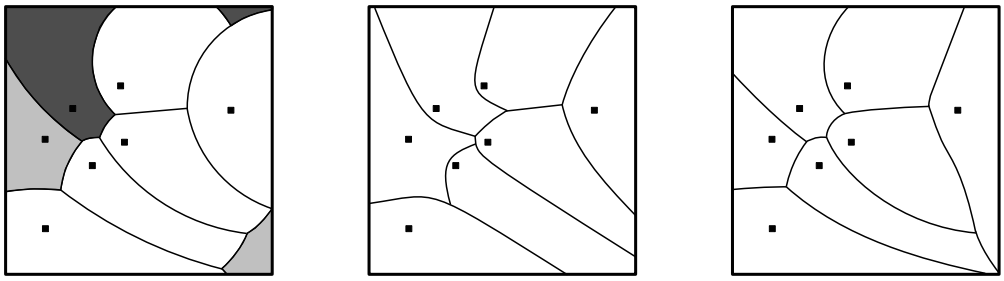


(a) $\mu = 0$ (b) $\mu = 0.167$ (c) $\mu = 0.333$ (d) $\mu = 0.5$



(e) $\mu = 0.67$ (f) $\mu = 0.83$ (g) $\mu = 1$ (h) $\mu^* = 0.3863$

Figure 3.1: The optimal partitions to problem (3.2) with varying μ, where $f(x)$ is the uniform distribution and we use the gravity model utility functions given by $u_i(x) = 1/\|x - p_i\|^2$ (it turns out that we can parameterize the boundary curves efficiently by using a transformation to bipolar two-center coordinates). Disconnected regions are indicated by shading. Figure (3.1h) shows the partition induced when $\mu = \mu^* = 0.3863$ is the "threshold" value (which we found using the method of bisection) for which the partition is connected.



(a) $\mu = 0$ (b) $\mu = 1$ (c) $\mu = 0.42$

Figure 3.2: Load-balancing partitions with $u_i(x) = \log \|x - p_i\|$ by adding a penalty term of $-\|x - p_i\|$. When $\mu = 0$ we have a multiplicative Voronoi diagram with disconnected Apollonian circles as boundary, when $\mu = 1$ we have an additive Voronoi diagram with connected hyperbolic arcs. (For $\mu = 0.42$ we have connected Cartesian ovals).

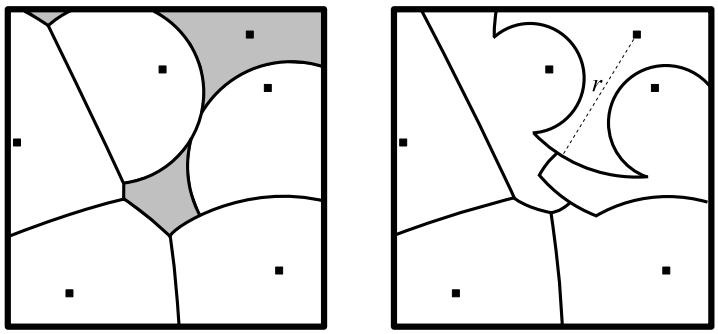


Figure 3.3: At left, a load-balancing partition without a distance constraint; the shaded region is disconnected. We can force our regions to be connected by imposing the distance constraint as shown at right.

demonstrating this. The integer program in this case is

$$\begin{aligned}
\underset{I_1(\cdot),\ldots,I_n(\cdot),t}{\text{minimize}}\ t \qquad & s.t. \\
t &\geq \iint_C I_i(x)\,\|x-p_i\|\,dA \quad \forall i \\
I_i(x) &= 0 \quad \forall i : \|x-p_i\| > r \\
\sum_{i=1}^{n} I_i(x) &= 1 \quad \forall x \in C \\
I_i(x) &\in \{0,1\} \quad \forall i, x\ .
\end{aligned} \tag{3.3}$$

The dual of the linear relaxation is

$$\underset{\lambda,\sigma(\cdot)}{\text{maximize}} \iint_C \sigma(x)\,dA \qquad s.t. \tag{3.4}$$

$$\begin{aligned}
\sigma(x) &\leq \lambda_i\,\|x-p_i\|\,dA \quad \forall x \in C,\ \forall i : \|x-p_i\| \leq r \\
\sum_{i=1}^{n} \lambda_i &\leq 1 \\
\lambda_i &\geq 0 \quad \forall i\ .
\end{aligned}$$

The optimal sub-region boundaries are a collection of circular arcs that come either from Apollonian circles or from the distance constraint. An example of such a partition is shown in Figure 3.3. When we impose an equal-area constraint, the optimal boundaries are either Cartesian ovals or the circles that arise from the distance constraint, as shown in Figure 3.4.

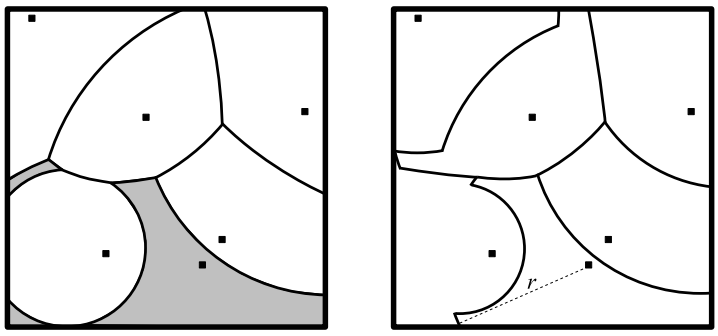


Figure 3.4: At left, an equal-area partition without a distance constraint; the shaded region is disconnected. Again we force our regions to be connected by imposing the distance constraint as shown at right.

3.2 Dynamic Partitioning

Notional conventions We adopt the standard notation of vector calculus; in particular, we let $\partial(\cdot)$ denote the *boundary operator* and we let $\nabla\cdot$ denote the *divergence operator*. We use a double integral $\iint_R f(x)\,dA$ to denote integration over a planar region.

In this section we consider the problem of partitioning a region optimally when the density $f(x)$ varies over time. For ease of exposition we shall focus on the case where R is a planar region, although the analysis herein extends naturally to higher dimensions. We consider the case where $f(x;t)$ is a probability density on $R \in \mathbb{R}^2$ that evolves according to a *vector field* $\vec{V}(x,t) : R \times \mathbb{R} \to \mathbb{R}^2$ that maps a given point $x \in R$ at a specified time t to a vector $(v_1(x,t), v_2(x,t))$; the physical interpretation of this system, naturally, is that an infinitesimal amount of mass located at point x moves in the direction $(v_1(x,t), v_2(x,t))$ at time t at a speed of $\|(v_1(x,t), v_2(x,t))\|$. Such a model of transport is canonical in a variety of contexts, including population migration [118], meteorology [63], and oceanography [89]. In such a setting, the density $f(x;t)$ is known to obey the *advection equation*

$$\frac{\partial f(x;t)}{\partial t} + \nabla \cdot (f(x;t)\vec{V}(x,t)) = 0 \tag{3.5}$$

where $\nabla\cdot$ is the *divergence operator*, defined as

$$\nabla \cdot \begin{pmatrix} h_1(x_1, x_2; t) \\ h_2(x_1, x_2; t) \end{pmatrix} := \frac{\partial h_1(x_1, x_2; t)}{\partial x_1} + \frac{\partial h_2(x_1, x_2; t)}{\partial x_2}\,.$$

In this section we will show how equation (3.5) allows us to define the changes in the optimal Lagrange multiplier vector $\boldsymbol{\lambda}^*$ over time, and thereby the optimal partition $R_1^*(t), \ldots, R_n^*(t)$ (since $\boldsymbol{\lambda}^*$ uniquely defines the optimal sub-regions R_i^*). Let $\boldsymbol{\lambda}^*(t)$ denote the optimal Lagrange multiplier vector at time t and let $R_i(\boldsymbol{\lambda}^*(t))$ denote the ith sub-region constructed according to Theorem 2.3.1 (it is of course true that $R_i(\boldsymbol{\lambda}^*(t)) = R_i^*(t)$, but for notational purposes we prefer to emphasize the dependence of $R_i^*(t)$ on $\boldsymbol{\lambda}^*(t)$). Using the result of Section 2.4.1, we can easily see that the optimality conditions of problem (2.7) simply require that

$$F_i(\boldsymbol{\lambda}; t) = q_i \tag{3.6}$$

for all $i \in \{1, \ldots, n\}$ and all t, where we have defined (for ease of notation)

$$F_i(\boldsymbol{\lambda}; t) := \iint_{R_i(\boldsymbol{\lambda})} f(x; t)\, dA\,.$$

Since $\sum_i q_i = \iint_R f(x; t) = 1$ for all t, we find that one of these constraints is redundant and therefore it will suffice to require that (3.6) holds for $i \in \{1, \ldots, n-1\}$ and all t; we also commensurately assign $\lambda_n = -(q_1\lambda_1 + \cdots + q_{n-1}\lambda_{n-1})/q_n$, to comply with the dual constraint that $\mathbf{q}^T\boldsymbol{\lambda} = 0$. We shall now use (3.6) to define the partial derivatives $\partial\lambda_i^*/\partial t$. We first introduce two well-known lemmas:

Lemma 3.2.1. *Let R_τ be a family of compact regions in the plane defined by*

$$R_\tau := \{x \in \mathbb{R}^2 : h(x) \leq \tau\}$$

where $h(x) : \mathbb{R}^2 \to \mathbb{R}$ is a smooth function. Suppose $f(x)$ is a density on $\mathbb{R}^2$ and define

$$m(\tau) := \iint_{R_\tau} f(x)\, dA$$

for all τ. Then

$$\frac{dm}{d\tau} = \int_{\partial R_\tau} \frac{f(x)}{\|\nabla h\|}\, ds\,,$$

where ds denotes a line integral over the boundary of R_τ, ∂R_τ.

Proof. This is a special case of the *coarea formula* [75]. □

Lemma 3.2.2. (Divergence theorem [2]) *Let R be a compact planar region with a smooth boundary. If $\vec{W}(x)$ is a smooth vector field defined on R, then*

$$\iint_R \nabla \cdot \vec{W}\, dA = \int_{\partial R} \vec{w} \cdot \vec{n}\, ds\,,$$

where $\vec{n}$ denotes the outward-facing unit normal vector pointing out of R.

Note that Lemma 3.2.1 gives us a clean expression for the partial derivatives $\partial F_i/\partial \lambda_j$:

$$\frac{\partial F_i}{\partial \lambda_j} = \begin{cases} 0 & \text{if } \partial R_i \cap \partial R_j = \emptyset \\ \sum_{j' : \partial R_i \cap \partial R_{j'} \neq \emptyset} \int_{\partial R_i \cap \partial R_{j'}} \frac{f(x;t)}{\|\nabla u_i(x) - \nabla u_{j'}(x)\|}\, ds & \text{if } i = j \\ -\int_{\partial R_i \cap \partial R_j} \frac{f(x;t)}{\|\nabla u_i(x) - \nabla u_j(x)\|}\, ds & \text{otherwise} \end{cases}$$

where we have suppressed the dependency of the R_i's on $\boldsymbol{\lambda}$ purely for notational compactness. Similarly, combining equation (3.5) with Lemma 3.2.2 gives us a clean expression for the partial derivatives $\partial F_i/\partial t$:

$$\begin{aligned} \frac{\partial F_i}{\partial t} &= \frac{\partial}{\partial t} \iint_{R_i(\boldsymbol{\lambda})} f(x;t)\, dA \\ &= \iint_{R_i(\boldsymbol{\lambda})} \frac{\partial}{\partial t} f(x;t)\, dA \\ &= -\iint_{R_i(\boldsymbol{\lambda})} \nabla \cdot (f(x;t)\vec{V}(x,t))\, dA \\ &= -\int_{\partial R_i(\boldsymbol{\lambda})} (f(x;t)\vec{V}(x,t)) \cdot \vec{n}\, ds\,. \end{aligned}$$

We thus have in hand expressions for $\partial F_i/\partial \lambda_j$ and $\partial F_i/\partial t$. The following result allows us to describe the optimal dual variables $\boldsymbol{\lambda}^*$ in terms of t:

Theorem 3.2.3. (Special case of the implicit function theorem [2]) *Consider*

the system of $n-1$ *equations in* n *variables*

$$\begin{aligned} F_1(\lambda_1,\ldots,\lambda_{n-1};t) - q_1 &= 0 \\ &\vdots \\ F_{n-1}(\lambda_1,\ldots,\lambda_{n-1};t) - q_{n-1} &= 0 \end{aligned}$$

and a point $(\boldsymbol{\lambda}_0, t_0)$ *that satisfies the system. Suppose that each of the functions* F_i *has continuous first partial derivatives with respect to each of the variables* λ_i *and* t *near* $(\boldsymbol{\lambda}_0, t_0)$*. Finally, suppose that*

$$\det\left[\frac{\partial F_i}{\partial \lambda_j}\right]\Bigg|_{(\boldsymbol{\lambda}_0,t_0)} \neq 0\,, \tag{3.7}$$

where $[\partial F_i/\partial\lambda_j]$ *denotes the Jacobian matrix of functions* $F_1,\ldots,F_{n-1}$ *with respect to* $\lambda_1,\ldots,\lambda_{n-1}$*. Then there exist functions* $\phi_1(t),\ldots,\phi_{n-1}(t)$ *such that*

$$\phi_i(t_0) = \lambda_i$$

for $i \in \{1,\ldots,n-1\}$ *and such that the equations*

$$\begin{aligned} F_1(\phi_1(t),\ldots,\phi_{n-1}(t);t) - q_1 &= 0 \\ &\vdots \\ F_{n-1}(\phi_1(t),\ldots,\phi_{n-1}(t);t) - q_{n-1} &= 0 \end{aligned}$$

hold for all t *sufficiently near* t_0*. Moreover,*

$$\frac{\partial \phi_j}{\partial t} = -\frac{\det\left[\frac{\partial F_i}{\partial\lambda_1\cdots\partial t\cdots\partial\lambda_{n-1}}\right]}{\det\left[\frac{\partial F_i}{\partial\lambda_j}\right]}\,,$$

where the numerator denotes the matrix obtained by replacing the j*th column of* $[\partial F_i/\partial\lambda_j]$ *with the vector of partial derivatives* $\partial F_i/\partial t$*.*

Corollary 3.2.1. If $\boldsymbol{\lambda}^*(t_0)$ is an optimal solution to problem (2.7) at time t_0, and if $f(x;t)$ evolves according to equation (3.5), then the optimal dual variables

$\boldsymbol{\lambda}^*(t)$ satisfy the differential equation

$$\left.\frac{\partial \lambda_i^*}{\partial t}\right|_{t=t_0} = -\left.\frac{\det\left[\frac{\partial F_i}{\partial \lambda_1 \cdots \partial t \cdots \partial \lambda_{n-1}}\right]}{\det\left[\frac{\partial F_i}{\partial \lambda_j}\right]}\right|_{t=t_0}. \tag{3.8}$$

Remark 3.2.4. We can further elaborate on a sufficient condition for (3.7) to hold. Let $\mathfrak{J} = [\mathfrak{J}_{ij}] := [\partial F_i/\partial \lambda_j]$ denote the Jacobian matrix of partial derivatives with respect to λ_1 through λ_{n-1} as before and note that $\mathfrak{J}_{ij} = 0$ if regions i and j do not share a boundary. Construct a graph G with $n-1$ nodes, where notes i and j share an edge if regions i and j share a boundary (G may not be connected since region n does not have a corresponding node). Assume without loss of generality that $\mathfrak{J}$ is decomposed into a block-diagonal form wherein each block $\mathfrak{B}_k$ consists of the connected components of G. Further note that, by construction, $\mathfrak{J}$ is diagonally dominant. If $\mathfrak{J}_{ij} < 0$ for all neighboring regions i and j (which always holds in all of our examples if $f(x) > 0$, for instance), then in each block, there exists at least one row (namely, any row i such that regions i and n share a boundary) at which this diagonal dominance is strict. Each block $\mathfrak{B}_k$ is therefore *irreducibly diagonally dominant* and thus nonsingular [120], which guarantees nonsingularity of $\mathfrak{J}$.

3.2.1 Computational Experiments

In this section, we consider a problem in which the density $f(\cdot)$ varies over time as in equation (3.5). We let R be the unit square and we let $f(x; t = 0)$ be a normal distribution with mean $(0.5, 0.5)$ and variance 0.3 with zero covariance. We let $u_i(x) := -\|x - p_i\|$ where the points p_i are regularly spaced in R.

As a vector field $\vec{V}(x;t)$ we use $v_1(x_1, x_2, t) := -\sin(x_1')\cos(x_2') + 0.1$ and $v_2(x_1, x_2, t) := \cos(x_1')\sin(x_2')$, where x_1' and x_2' are obtained by applying an affine map to x_1 and x_2 (we use such a map so as to have a vector field that is not aligned with the coordinate axes), and we also apply a dampening filter to force $\|\vec{V}(x,t)\|$ to be small when x is near the boundary of R (this allows us to sidestep the issue of mass entering or exiting R). The field is shown in Figure 3.5. Note that we have chosen to keep $\vec{V}$ constant over time in order to make the flow of $f(x;t)$ more recognizable.

In order to simulate the advection over time we use the Clawpack simulator

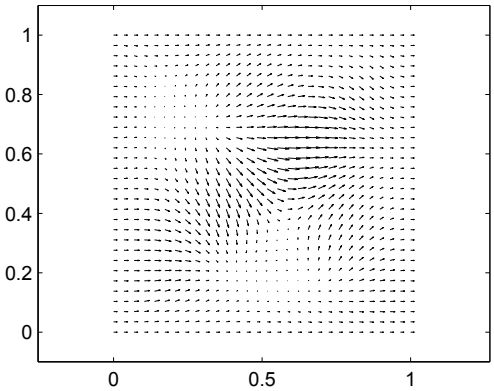
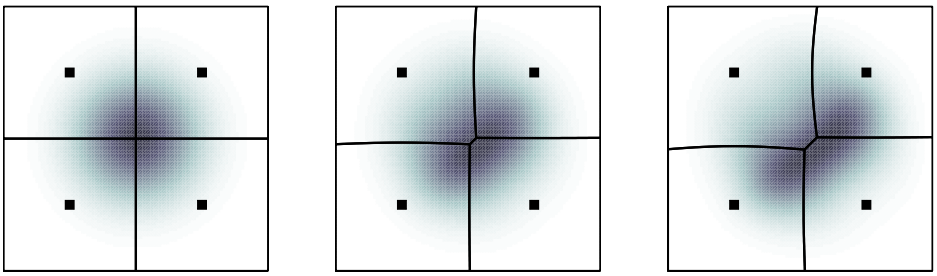


Figure 3.5: The vector field $\vec{V}(x_1, x_2; t)$.

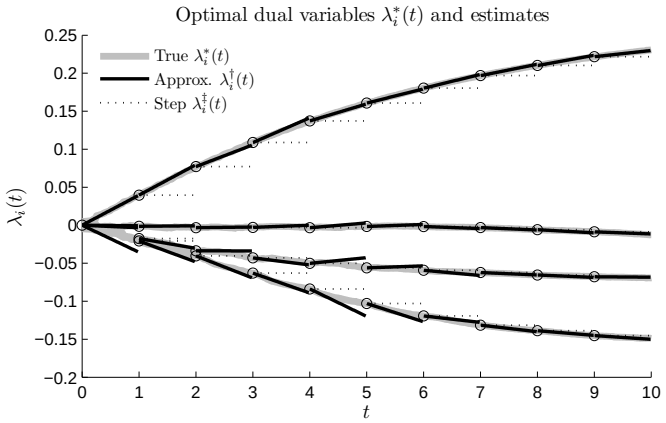
[77] for the period $t \in [0, 10]$, with 200×200 grid cells, using a timestep of $\Delta t = 0.01$. For each of the 1000 advection iterations we compute the optimal dual variables $\lambda_i^*(t)$ and the optimal partitions, which are shown in Figure 3.6. We also compute approximately optimal dual variables $\lambda_i^\dagger(t)$ using Corollary 3.2.1 as follows: for $t \in \{0, 1, \ldots, 9\}$, we let $\lambda_i^\dagger(t) = \lambda_i^*(t)$, and for non-integer t, we let $\lambda_i^\dagger(t)$ be the value of $\lambda_i^*(t)$ as prescribed by equation (3.8). Thus, $\lambda_i^\dagger(t)$ is a piecewise linear function that "resets" itself whenever t is an integer. For purposes of comparison we also let $\lambda_i^\ddagger(t)$ be a *step* function that "resets" itself whenever t is an integer, that is, $\lambda_i^\ddagger(t) := \lambda_i^*(\lfloor t \rfloor)$. In summary, $\lambda_i^\dagger(t)$ is a first-order approximation of $\lambda_i^*(t)$ and $\lambda_i^\ddagger(t)$ is a zeroth-order approximation. Figure 3.7a shows the trajectories of these two approximations against the true optimal trajectories $\lambda_i^*(t)$. Figure 3.7b shows the resulting values of $F_i(\boldsymbol{\lambda}; t)$ under these two approximations; note that the regions defined by $\lambda_i^\dagger(t)$ are, not surprisingly, consistently more balanced than those of the step approximation $\lambda_i^\ddagger(t)$.



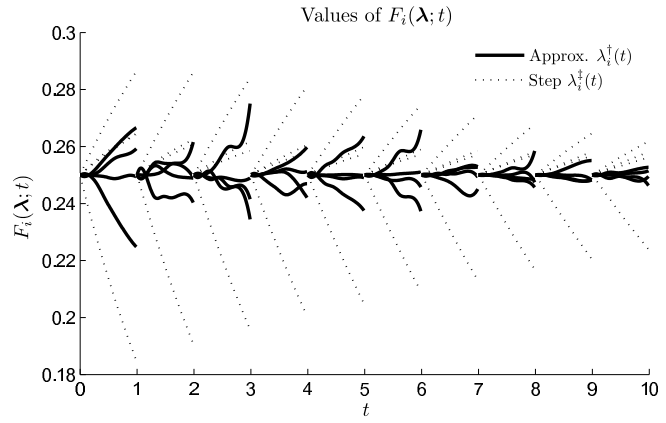
(a) $t = 0$ (b) $t = 2$ (c) $t = 4$

(d) $t = 6$ (e) $t = 8$ (f) $t = 10$

Figure 3.6: Advection over the unit square for $t \in [0, 10]$ and the optimal partitions $R_1^*(t), R_2^*(t), R_3^*(t), R_4^*(t)$.



(a)



(b)

Figure 3.7: The optimal dual variables $\lambda_i^*(t)$ and their approximations $\lambda_i^\dagger(t)$ and $\lambda_i^\ddagger(t)$, and the resulting masses of the induced partitions.

Chapter 4

Dividing a Territory with Obstacles

4.1 Introduction

In Chapters 2 and 3, we developed a technique to solve the map segmentation problem while restricting ourselves to simply connected regions or simple polygons. In practical applications such as in military and robotics, the input region is often not only non-convex but has "holes" (or obstacles) in them. For example, drones service a geographic region which may contain mountains or buildings which pose as obstacles for them. Allocating workload for a fleet of drones will hence involve partitioning a region which is no more simply connected. There exists little literature which attempt to partition such a region with holes. This problem is critical to solve specially with the tremendous increase in usage of drones, thanks to the *FAA Modernization and Reform Act of 2012* [92].

In this chapter, we consider the problem of dividing a given geographic territory among a set of vehicles in such a way as to minimize the total workload imposed on them while simultaneously ensuring that all vehicles service the same amount of territory. More precisely, we are given a planar region R and a set of n fixed landmark points ("vehicle depots") $P = \{p_1, \ldots, p_n\}$, and our objective is to design sub-regions $\{R_1, \ldots, R_n\}$ of equal area that minimize the workloads of the vehicles p_i in providing service to their respective sub-regions R_i. The set $\{R_1, \ldots, R_n\}$ should obviously be a *partition* of R, that is, we should require that $\bigcup_i R_i = R$ and that the interiors do not overlap, $\text{int}(R_i) \cap \text{int}(R_j) = \emptyset$ for

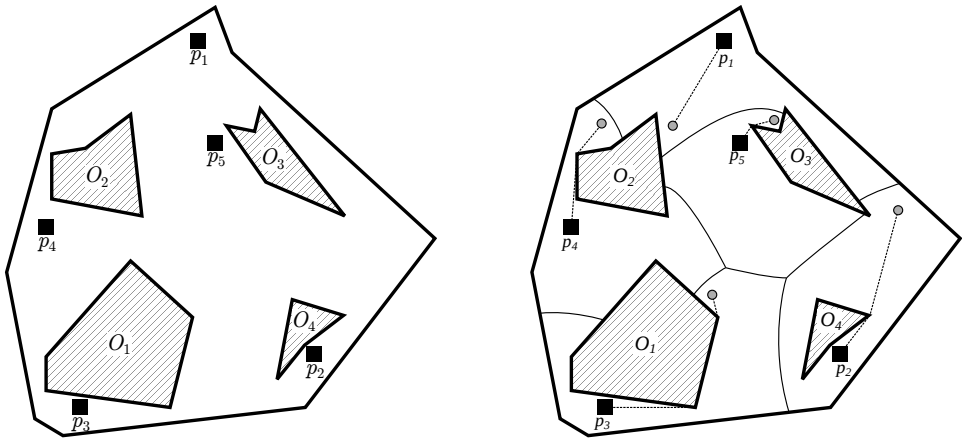


Figure 4.1: Inputs (4.1a) and outputs (4.1b) to our partitioning problem; we are given a region R containing $m = 4$ obstacles and $n = 5$ points p_i, and we design $n = 5$ sub-regions of equal area corresponding to the vehicles that minimize the workloads as defined in (4.1).

all distinct pairs i, j. Since, the input region R over which the drones travel need not always be a convex region; we will consider a more general case in which the region R contains a collection of impenetrable obstacles, such as tall buildings or mountains. To incorporate a realistic ground environment, we consider R to be represented by a simple polygon that contains a set of *obstacles* $\{O_1, \ldots, O_m\}$ (also simple polygons) as shown in Figure 4.1a.

To model the workload of vehicle i in providing service to region R_i, we find it useful to introduce what we call the *Fermat-Weber* function $\mathrm{FW}(R_i, p_i)$, defined as

$$\mathrm{FW}(R_i, p_i) := \iint_{R_i} d(x, p_i)\, dx \tag{4.1}$$

where $d(\cdot, \cdot)$ is the length of the shortest path between a pair of points when obstacles are taken into account. A vehicle i travels a distance of $d(x, p_i)$ to service a point x. In practice, drones have less fuel holding capacity or short lived batteries and hence they are often required to return to their depots periodically for re-fueling or updating information. The function $\mathrm{FW}(R_i, p_i)$ thus defines the workload of the i^{th} vehicle and it is simply proportional to the average distance between p_i and a point uniformly randomly sampled in its service region R_i.

It is obvious that, by seeking to minimize the total vehicle workloads, our partition will be *compact*, in the sense that a vehicle should not be assigned to

service points that are far away from it. In fact, the partitions that we produce turn out to be *connected*, so that all districts are contiguous; see Figure 4.1. In addition, we may require that these service sub-regions be computable in a *decentralized* fashion [91]: that is, the sub-region assigned to vehicle i should be computable using only "local" information to vehicle i (such as nearby neighbors or information about its surroundings), and the optimal boundary between two sub-regions should be computable using only knowledge available to those two vehicles.

4.1.1 Contribution

Designing boundaries between sub-regions is an infinite dimensional problem and is usually extremely complicated. The traditional way to solve it is to break the region into small discrete "pixels" and assign binary variables to each pixel. This essentially becomes a large combinatorial problem which is computationally intractable for large problem instances. In this chapter, we show how we transform our partitioning problem into a simple n-dimensional convex optimization problem that determines the optimal partition boundaries using a sequence of *cutting planes*. Our algorithm can be readily applied to any planar connected region that admits a representation via poly-lines. Our algorithm also has the advantage of *decentralization*, where each vehicle depot can compute its sub-region by obeying a simple control law and communicating with its nearby neighboring depot points. A side consequence of our analysis is a collection of simple and immediate proofs of established results in equitable partitioning [7, 10, 101] and an affirmative answer to a question posed in the concluding remarks of [6, 7].

As will be explained, our result generalizes to the cases where unequal areas are desired (say we require instead that $\text{Area}(R_i) = q_i$ for some input vector $\mathbf{q}$) and where we have a probability density function $f(\cdot)$ defined on R and we place conditions on the *masses* of the regions, $\iint_{R_i} f(x)\,dx$, rather than the areas.

The algorithm of this work presents two clear advantages over the previous methods: first, it is easy to show that when the input region R contains obstacles (and is thus no longer simply connected), an equitable relatively convex partition of the type just described may not exist (see Figure 1 of [20]). Second, the algorithms of [41, 42] and [39] do not force regions to be *compact*: it is possible for

the output regions to be extremely long and skinny, which is clearly undesirable from a practical standpoint (see Figure 10 of [41] and Figure 14 of [39]).

Here we tackle these two problems in the following way: first, as we will show, our algorithm produces sub-regions $\{R_1, \ldots, R_n\}$ that are *relatively star-convex* to R: for any point x in sub-region R_i, the shortest path in R from x to p_i is itself contained in R_i (see Figure 4.1b). This is a weaker condition than relative convexity but still offers considerable practical utility; for example, it automatically produces connected sub-regions. Second, we are able to explicitly enforce our sub-regions to be compact; in fact, one interpretation of our problem is that we are trying to find the partition that is as compact as possible; we will make this explicit in Section 4.2. The algorithm in this work is also novel in that the work in [41, 42] and [39] generally uses well-established principles of discrete geometry such as ham sandwich cuts and binary and ternary space partitions, whereas here we use techniques from vector space optimization and linear programming complementarity.

4.1.2 Related Work

A well-studied related problem in operations research is the *Fermat-Weber problem*, in which our objective is to place a facility p (or collection of facilities) in C so as to minimize the average distance between points in C and p. Discrete and continuous versions of this problem are discussed at length in [50], and [56] gives the first polynomial-time algorithm for various versions of the 1-norm problem. The authors also prove that the 1-norm problem with multiple facilities is NP-hard for large n.

Two other variations commonly encountered in continuous facility placement are the n-center problem [115], in which the objective is to cover C with n identical circles with the smallest possible radius, and the minimum equitable radius problem [116], in which the objective is to place n facilities whose Voronoi cells have equal area while minimizing the maximum distance from a point to its assigned facility. In most continuous facility placement problems, the partition of C is given by the Voronoi diagram of the facilities [95]. Thus, one other contribution of our work is to show that not-insignificant savings can be made when the partition is also an optimization variable, and in fact that it can be optimized for a given set of facilities in a tractable way.

Considerably less work is published on the problem of partitioning C optimally when the depot points are fixed and almost all of those consider the inpur region to convex or simple connected. One notion of "partitioning" discussed in [24] is to allow facilities to have variable "coverage radii" r_i, where the cost $\phi(r_i)$ is a monotonically increasing function; the problem is to find the optimal number, location, and coverage radii of a collection of facilities. The authors in [41, 42] give a fast algorithm that takes as input a *convex* polygon C and a point set $P = \{p_1, \ldots, p_n\}$ and divides C into n sub-regions $\{C_1, \ldots, C_n\}$ such that each C_i is convex, each C_i contains a point, and all sub-regions have equal area. In [39] they extend this algorithm to the case where the input region is a simply connected polygon S (i.e. a connected polygon and devoid of any obstacles) and we have an arbitrary probability density $f(\cdot)$ defined on S. Rather than producing equal-area sub-regions $\{S_1, \ldots, S_n\}$, our objective is to find a partition such that $\iint_{S_i} f(x)\, dA$ is equal for all sub-regions ([41] is thus a special case of this problem in which $f(\cdot)$ is a uniform distribution). Since it is not always possible to divide S into *convex* pieces in this case, our algorithm divides S into sub-regions that are *relatively convex* to S: for any two points x, y contained in a sub-region S_i, the shortest path in S from x to y (which may not be a straight line segment) is itself contained in S_i (see Figures 2, 3, and 12 of [39] for examples). This is clearly a natural generalization of convexity, because when S happens to be convex, the shortest path from x to y in S is indeed a straight line segment, and therefore all sub-regions S_i will be convex.

In addition to the work mentioned, the general problem of equitably partitioning a region has been studied from many perspectives. They are of particular interest to the robotics community. For example, the paper [101] considers the problem of dividing a convex region into convex equal-area sub-regions using power diagrams. The authors give a decentralized control policy that provably converges to the desired partition. More recently, [53] considers a hybrid problem in which the objective is to simultaneously place the facilities P and design coverage regions associated with the facilities using only asynchronous (and possibly noisy) pairwise communication between facilities. The authors give an algorithm that provably converges to a *centroidal Voronoi partition*, that is, a Voronoi diagram $\{V_1, \ldots, V_n\}$ such that p_i is the center of mass of each sub-region V_i. Other papers making use of power diagrams as a partitioning policy include [85] and [104].

Closely related papers [6, 7] considers the problem of partitioning a *convex* region C so as to minimize the aggregate workload over all facilities while imposing an equal-area constraint. The authors describe a constant-factor approximation algorithm for dividing C into equal-area convex pieces to maximize the minimum "fatness" of any piece. This in turn gives an approximation algorithm for the problem of minimizing the aggregate workload over all facilities when facility placement is variable, as well as the sub-region boundaries. The authors also give a constant-factor approximation algorithm for solving the convex case of precisely the problem that we address here, which will be introduced in the following section. The paper [10] considers the problem of dividing a convex region into convex sub-regions according using *power diagrams*, a natural extension of *Voronoi diagrams*. The authors develop a rich and elegant theory relating such partitions to the infinite-dimensional least-squares problem, which we will touch on in Remark 4.3.4 of this chapter.

Notational Conventions and Technical Assumptions

Our notational conventions in this chapter will be as follows: in addition to the Fermat-Weber function $\mathrm{FW}(\cdot,\cdot)$ and the distance function we have introduced, we also remark that, when we refer to the input region R, we implicitly take the obstacles into account (so that $R = S \setminus \bigcup_i O_i$ for some simply connected polygon S). We will assume throughout this chapter that $\mathrm{Area}(R) = 1$.

4.2 Problem Formulation

We begin by formally stating our optimization problem for designing the service districts R_i. Suppose that $\mathrm{Area}(R) = 1$ and that the cost of service between a demand point x and depot i is simply $d(x, p_i)$, where the points p_i are fixed and given as inputs. If demand is uniformly distributed over R, the average workload on vehicle at depot i, when assigned to provide service to region R_i,is precisely

$$\mathrm{FW}(R_i, p_i) = \iint_{R_i} d(x, p_i)\, dx$$

as stated previously. It is obvious that the total workload on all vehicles is minimized when each point x is merely assigned to its nearest depot, i.e. when

$\{R_1, \ldots, R_n\}$ is a Voronoi partition of P in R. In order to balance the workloads of the vehicles, we will impose the constraint that $\text{Area}(R_i) = \text{Area}(R)/n = 1/n$ for each i. We can thus write our optimization problem as

$$\begin{aligned} \underset{R_1,\ldots,R_n}{\text{minimize}} \sum_{i=1}^{n} \text{FW}(R_i, p_i) \qquad & s.t. \\ \text{Area}(R_i) &= 1/n \\ \bigcup_{i=1}^{n} R_i &= R\,. \end{aligned} \tag{4.2}$$

It is easy to see that the objective function of (4.2) forces regions to be compact because it minimizes the average distance between demand points and their assigned vehicle depots. It is also worth mentioning that (4.2) is a special case of the famous *Monge-Kantorovich transportation problem* [121] in the plane in which one Radon measure is the uniform distribution and the other is atomic: our objective is to "transport" the continuously distributed demand to the finite collection of facilities, while obeying capacity constraints and minimizing the aggregate transportation cost.

4.3 Optimal Partitioning

For ease of exposition, we find it helpful to give an outline of this section:

- We first show how to formulate our problem (4.2) as an infinite-dimensional optimization problem (4.3), which we then relax to an infinite-dimensional linear program (4.4).
- We next take the *dual* (in the sense of linear programming) of problem (4.4), and show that this dual (4.7) can be expressed in terms of n Lagrange multipliers (Theorem 4.3.1).
- Finally we show that the optimal solution to problem (4.4) can be recovered via the optimal solution to (4.7) (Theorem 4.3.2), and that this optimal solution to (4.4) is also a solution to problem (4.3) and therefore to our original problem (4.2) (Theorem 4.3.3).

It is clear that we can re-write our initial formulation (4.2) as an infinite-dimensional optimization problem by introducing indicator function variables

$I_1(\cdot), \ldots, I_n(\cdot)$ defined on R in the following fashion:

$$\begin{aligned} \underset{I_1(\cdot),\ldots,I_n(\cdot)}{\text{minimize}} \iint_R \sum_{i=1}^{n} d(x,p_i) I_i(x)\, dx \qquad & s.t. \\ \iint_R I_i(x)\, dx &= 1/n \quad \forall i \\ \sum_{i=1}^{n} I_i(x) &= 1 \quad \forall x \in R \\ I_i(x) &\in \{0,1\} \quad \forall i, x\,. \end{aligned} \tag{4.3}$$

Thus, R_i consists of precisely those points x for which $I_i(x) = 1$. The above problem is, of course, an infinite-dimensional *integer* program which we expect to be difficult to solve. We can relax the integrality constraint that $I_i(x) \in \{0,1\}$ to obtain a related infinite-dimensional *linear* program:

$$\begin{aligned} \underset{I_1(\cdot),\ldots,I_n(\cdot)}{\text{minimize}} \iint_R \sum_{i=1}^{n} d(x,p_i) I_i(x)\, dx \qquad & s.t. \\ \iint_R I_i(x)\, dx &= 1/n \quad \forall i \\ \sum_{i=1}^{n} I_i(x) &= 1 \quad \forall x \in R \\ I_i(x) &\geq 0 \quad \forall i, x\,. \end{aligned} \tag{4.4}$$

We can discretize the above problem into N grid cells $\square_j$ of area $\epsilon = 1/N$, where d_{ij} denotes the distance between p_i and the center of cell $\square_j$ and z_{ij} denotes the fraction of cell $\square_j$ assigned to p_i, to obtain the approximate formulation

$$\begin{aligned} \underset{Z=[z_{ij}]}{\text{minimize}} \sum_{i=1}^{n} \sum_{j=1}^{N} \epsilon d_{ij} z_{ij} \qquad & s.t. \\ \sum_{j=1}^{N} \epsilon z_{ij} &= 1/n \quad \forall i \\ \sum_{i=1}^{n} z_{ij} &= 1 \quad \forall j \\ z_{ij} &\geq 0 \quad \forall i, j\,. \end{aligned} \tag{4.5}$$

It is a standard exercise in linear programming to verify that the dual problem to (4.5), which has Lagrange multipliers $\boldsymbol{\lambda} \in \mathbb{R}^n$ and $\boldsymbol{\sigma} \in \mathbb{R}^N$, is

$$\begin{aligned} \underset{\boldsymbol{\lambda},\boldsymbol{\sigma}}{\text{maximize}}\, \frac{1}{n}\sum_{i=1}^{n}\lambda_i + \sum_{j=1}^{N}\epsilon\sigma_j \qquad & s.t. \\ \lambda_i + \sigma_j \;&\leq\; d_{ij} \quad \forall i,j\,. \end{aligned}$$

which is a discretization of the problem

$$\begin{aligned} \underset{\boldsymbol{\lambda},\sigma(\cdot)}{\text{maximize}}\, \frac{1}{n}\sum_{i=1}^{n}\lambda_i + \iint_R \sigma(x)\,dx \qquad & s.t. \\ \sigma(x) \;&\leq\; d(x,p_i) - \lambda_i \quad \forall i,x \end{aligned} \tag{4.6}$$

which is itself equivalent to the unconstrained problem

$$\underset{\boldsymbol{\lambda}}{\text{maximize}}\, \frac{1}{n}\sum_{i=1}^{n}\lambda_i + \iint_R \min_i \{d(x,p_i) - \lambda_i\}\; dx\,.$$

Finally, we note that the above problem is translation-invariant in $\boldsymbol{\lambda}$ because we have assumed that $\text{Area}(R) = 1$ and thus we obtain the convex, n-dimensional dual problem

$$\begin{aligned} \underset{\boldsymbol{\lambda}}{\text{maximize}} \iint_R \min_i \{d(x,p_i) - \lambda_i\}\; dx \qquad & s.t. \\ \sum_{i=1}^{n}\lambda_i \;&=\; 0 \end{aligned} \tag{4.7}$$

so that we can state our first result:

Theorem 4.3.1. *The dual problem of the infinite-dimensional linear program (4.4) is the finite-dimensional convex problem (4.7).*

Proof. There is nothing more to do here, except to provide some justification for the discretization that we introduced to obtain (4.5) from (4.4). This can be made entirely rigorous using established principles of vector space optimization [82]. This can be easily shown to be a special case of Theorem 2.3.1, which is then proved in Section A.1. □

We will next show that the optimal solution to (4.4), $I_1^*(\cdot), \ldots, I_n^*(\cdot)$, can be recovered from the optimal solution to (4.7), $\boldsymbol{\lambda}^*$: consider any point $x \in R$ and the optimal solution $\boldsymbol{\lambda}^*$ to (4.7). Suppose $\bar{i}$ is the index such that $d(x, p_i) - \lambda_i^*$ is minimal (assuming such an index is unique). From basic linear programming theory, we know that the *complementary slackness* conditions of problem (4.6) stipulate that $I_i^*(x) = 0$ for all indices i other than $\bar{i}$, and consequently that $I_{\bar{i}}^*(x) = 1$. From this we have proven our second result:

Theorem 4.3.2. *The optimal solution $I_1^*(\cdot), \ldots, I_n^*(\cdot)$ to the infinite-dimensional linear program (4.4) must satisfy*

$$I_i^*(x) = \begin{cases} 0 & \text{if } d(x, p_i) - \lambda_i^* > d(x, p_j) - \lambda_j^* \text{ for some } j \\ 1 & \text{if } d(x, p_i) - \lambda_i^* < d(x, p_j) - \lambda_j^* \text{ for all } j \neq i \end{cases}$$

where $\boldsymbol{\lambda}^*$ is the optimal solution to (4.7). If neither of the two cases above holds at a point x, then if $I_i^*(x) > 0$ at a point x, it must be the case that $d(x, p_i) - \lambda_i^* \leq d(x, p_j) - \lambda_j^*$ for all j, i.e. the index i is among the minimal indices. In other words, the optimal solution to 4.3 is an additively weighted Voronoi diagram.

Note that Theorem 4.3.2 tells us that the optimal solution to the linear relaxation (4.4) has the useful property that $I_i^*(x) \in \{0, 1\}$, except possibly for those points x such that $d(x, p_i) - \lambda_i^* = d(x, p_j) - \lambda_j^*$ for two indices i, j. We will next show that there actually exists an optimal solution $I_1^*(\cdot), \ldots, I_n^*(\cdot)$ to the linear relaxation (4.4) in which $I_i^*(x) \in \{0, 1\}$ for *all* $x \in R$. This must, therefore, also be a solution to the problem (4.3) and therefore to our original partitioning problem (4.2).

Theorem 4.3.3. *There exists an optimal solution $I_1^*(\cdot), \ldots, I_n^*(\cdot)$ to the infinite-dimensional linear program (4.4) such that $I_i^*(x) \in \{0, 1\}$ for all $x \in R$, which is therefore also an optimal solution to (4.3). Thus, the optimal solution to problem (4.2) is an additively weighted Voronoi partition of R with respect to P.*

Proof. Given the optimal Lagrange muliplier vector $\boldsymbol{\lambda}^*$ from (4.7), we define the

"strict dominance regions" $R_i^+ \subseteq R_i^*$ as

$$R_i^+ = \left\{ x \in R : d(x, p_i) - \lambda_i^* < d(x, p_j) - \lambda_j^* \text{ for all } j \neq i \right\} .$$

It is easy to see by construction that each R_i^+ is *relatively star-convex* to R: if $x \in R_i^+$, the shortest path (or paths) in R from x to p_i is contained in R_i^+. From basic Euclidean geometry it is also clear that the boundary between two strict dominance regions R_i^+ and R_j^+ consists of a collection of hyperbolic arcs, since a hyperbola is the locus of points where the absolute value of the difference of the distances to two foci is constant (if we measure point-to-point distances using the ℓ_1 or ℓ_∞ norms, while still taking obstacles into account, the boundary between two strict dominance regions consists instead of a collection of line segments). Given $\boldsymbol{\lambda}^*$, we can construct these arcs for all R_i^+ efficiently using the *continuous Dijkstra paradigm* [87]. The question remains of how to allocate the remaining area of R that does not lie in a strict dominance region (see Figure 4.2).

It is not hard to show that each such region is polygonal (as opposed to being bounded by hyperbolic arcs), and indeed, determining a proper assignment is a fairly straightforward procedure. Since each such "ambiguous dominance region" is potentially associated with more than one point p_i, we associate with each such region R_k^- (not indexing by i because these regions are not yet associated with individual facilities) an index set $\mathcal{I}_k \subseteq \{1, \ldots, n\}$. By construction, each R_k^- must have a *reflex vertex* r_k, that is, a point in R_k^- such that $d(x, p_i) = d(x, r_k) + d(r_k, p_i)$ for all $x \in R_k^-$ and $i \in \mathcal{I}_k$; an example of this is the point marked v in Figure 4.2. Consider the functions $I_i^*(\cdot)$ (which we have not

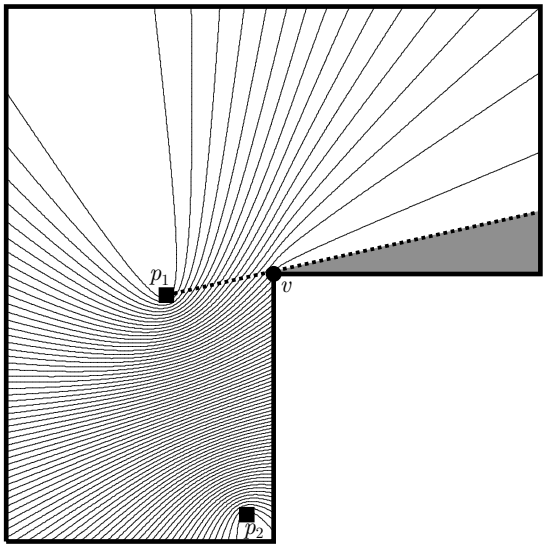


Figure 4.2: As λ_1 and λ_2 vary, the induced sub-regions change as indicated above. The points x in the shaded region do not have a unique minimal index $i_{\min}$ because $\lambda_1 - \lambda_2 = d(v, p_1) - d(v, p_2)$. The point v is therefore the *reflex vertex* associated with the shaded region.

yet defined on R_k^-) for $i \in \mathcal{I}_k$. The total cost due to R_k^- is

$$\begin{aligned}
& \iint_{R_k^-} \sum_{i \in \mathcal{I}_k} d(x, p_i) I_i^*(x)\, dx \\
= & \iint_{R_k^-} \sum_{i \in \mathcal{I}_k} (d(x, r_k) + d(r_k, p_i)) I_i^*(x)\, dx \\
= & \iint_{R_k^-} d(x, r_k) \underbrace{\left(\sum_{i \in \mathcal{I}_k} I_i^*(x) \right)}_{=1} dx + \iint_{R_k^-} \sum_{i \in \mathcal{I}_k} d(r_k, p_i) I_i^*(x)\, dx \\
= & \underbrace{\iint_{R_k^-} d(x, r_k)\, dx}_{\text{constant}} + \sum_{i \in \mathcal{I}_k} d(r_k, p_i) \iint_{R_k^-} I_i^*(x)\, dx
\end{aligned}$$

which we observe only depends on $\iint_{R_k^-} \sum_{i \in \mathcal{I}_k} I_i^*(x)\, dx$, that is, the *areas* of region R_k^- assigned to the facilities $i \in \mathcal{I}_k$, as opposed to the particular assignment patterns $I_i^*(x)$. Thus, in order to assign these amounts to the facilities, we can

simply construct a matrix A such that

$$a_{ik} = \begin{cases} 1 & \text{if } i \in \mathcal{I}_k \\ 0 & \text{otherwise} \end{cases}$$

and then consider the problem of constructing a matrix Z that gives the areas assigned to each facility from each R_k^-, i.e. satisfying the constraints

$$\begin{aligned} \sum_i a_{ik} z_{ik} &= \iint_{R_k^-} dx \quad \forall k \\ \sum_k a_{ik} z_{ik} &= c_i - \iint_{R_i^+} dx \quad \forall i \\ z_{ik} &\geq 0 \quad \forall i, k \,. \end{aligned} \tag{4.8}$$

It follows from the argument above that the total cost due to these allocations is then $\sum_{i,k} b_{ik} z_{ik}$, where B is a matrix such that $b_{ik} = d(r_k, p_i)$. In fact, it is not hard to show that all feasible solutions to (4.8) have the same objective value $\sum_{i,k} b_{ik} z_{ik}$: this is precisely because, for any ambiguous dominance region R_k^-, we must have $d(r_k, p_i) - d(r_k, p_j) = \lambda_i - \lambda_j$ for all $i, j \in \mathcal{I}_k$. Thus, the columns of B are simply the vector $\boldsymbol{\lambda}^*$, translated by a scalar, and with all indices $j \notin \mathcal{I}_k$ set to 0:

$$B = \begin{pmatrix} | & & | & & | \\ \boldsymbol{\lambda}^*_{\mathcal{I}_1} + \beta_1, & \cdots, & \boldsymbol{\lambda}^*_{\mathcal{I}_k} + \beta_k, & \cdots, & \boldsymbol{\lambda}^*_{\mathcal{I}_K} + \beta_K \\ | & & | & & | \end{pmatrix}$$

where K is the number of ambiguous dominance regions. After finding a feasible (and therefore optimal) set of allocations Z^*, we merely have to find a way to assign the areas z^*_{ik} so that the final regions R_i^* are relatively star-convex. This is trivial as shown in Figure 4.3. □

Some further commentary is in order at this time:

Remark 4.3.4. The special case of Theorem 4.3.3 where $d(x, y) = ||x - y||_2$ (i.e. where R is convex and without obstacles) was already proven in [6, 7] (whose analysis extends to our case with obstacles without incident), and the special case where $d(x, y) = ||x - y||_2^2$ was proven in [10]. However, our proof

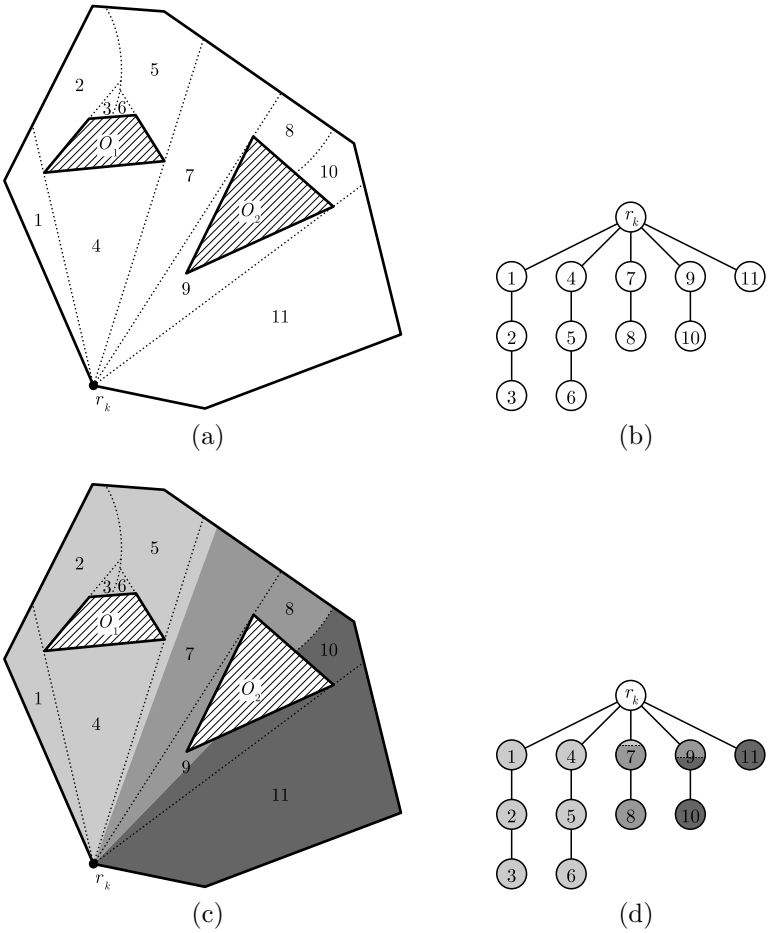
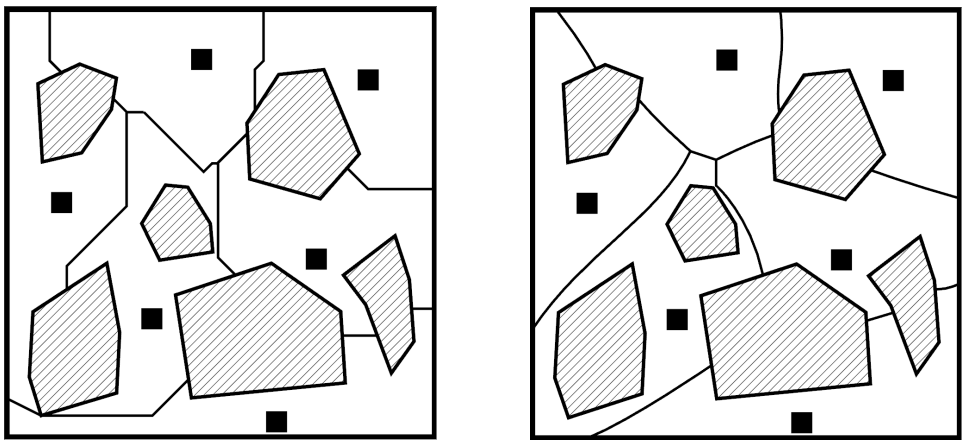


Figure 4.3: Given an ambiguous dominance region R_k^- (4.3a) and a set of allocations Z^*, it is straightforward to divide R_k^- into pieces that are relatively star-convex by constructing a shortest-path tree (4.3a and 4.3b) and then assigning regions based on (for example) a depth-first search of the tree (4.3c and 4.3d).



(a) (b)

Figure 4.4: Equitable partitioning under the ℓ_1 norm (4.4a) versus the ℓ_2 norm (4.4b).

of Theorem 4.3.3 is novel in two ways: first, it generalizes these prior results by establishing that both are simple and immediate consequences of duality theory in linear programming (since our proofs carry forward for *any* cost function $d(x, p_i)$). Second, as we will demonstrate in the following section, Theorem 4.3.3 is actually *constructive* because one can find the optimal Lagrange multipliers $\boldsymbol{\lambda}^*$ (and thereby the optimal solution to our original partitioning problem (4.2)) efficiently, for any cost function $d(x, p_i)$.

Remark 4.3.5. If we desire regions with *line segments* as boundaries instead of hyperbolas (perhaps for computational efficiency), we can measure point-to-point distances using the ℓ_1 or ℓ_∞ norms (while still taking obstacles into account). Figure 4.4 shows the difference between partitions induced by using such a distance metric.

Remark 4.3.6. It is straightforward to verify that the optimal solution to problem (4.2) is still an additively weighted Voronoi diagram when we require that all sub-regions have the same mass of some *probability density* $f(\cdot)$ (instead of equal areas). Note also that we have not made explicit use of the requirement that all sub-regions have *equal* area (or mass); indeed Theorem 4.3.3 remains true for any arbitrary set of assignments of areas or masses to the sub-regions (provided, of course, that the desired areas or masses sum to Area(R)).

4.4 Super-Gradients for Solving the Dual Problem

In the previous section, we showed that our partitioning problem can be reduced to the n-dimensional convex optimization problem (4.7). In this section, we describe how to solve problem (4.7) efficiently. Our key tool in doing so is to show how to quickly construct a *supergradient* vector [30] of (4.7), defined as follows:

Definition 4.4.1. A *supergradient* of a function $h(\boldsymbol{\lambda})$ is a vector $\mathbf{g}$ such that $h(\boldsymbol{\lambda}') \leq h(\boldsymbol{\lambda}) + \mathbf{g}^T(\boldsymbol{\lambda}' - \boldsymbol{\lambda})$ for all $\boldsymbol{\lambda}'$.

This is useful for us because we can then solve (4.7) efficiently using a standard *analytic center cutting plane method* (ACCPM) to determine $\boldsymbol{\lambda}^*$ within arbitrary precision [31]. For the moment, for a given weight vector $\boldsymbol{\lambda}$, we let $R_i(\boldsymbol{\lambda})$ denote the additively weighted Voronoi cell corresponding to point p_i.

Theorem 4.4.2. *The vector* $\mathbf{g}$ *defined by*

$$g_i := \text{Area}(R_i(\boldsymbol{\lambda}))$$

is a supergradient for the concave function

$$h(\boldsymbol{\lambda}) = \iint_R \min_i \{d(x, p_i) - \lambda_i\}\, dx\,.$$

Proof. Consider two vectors $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}'$ and the corresponding additively weighted Voronoi partitions $\{R_1, \ldots, R_n\}$ and $\{R_1', \ldots, R_n'\}$ (we omit the dependence of the partitions on $\boldsymbol{\lambda}$ for brevity). We want to show that $h(\boldsymbol{\lambda}') \leq h(\boldsymbol{\lambda}) + \mathbf{g}^T(\boldsymbol{\lambda}' - \boldsymbol{\lambda})$, i.e. that

$$\iint_R \min_i \{d(x, p_i) - \lambda_i'\}\, dx \leq \iint_R \min_i \{d(x, p_i) - \lambda_i\}\, dx - \mathbf{g}^T(\boldsymbol{\lambda}' - \boldsymbol{\lambda})$$

or equivalently that

$$\sum_{i=1}^{n} \iint_{R_i'} d(x, p_i) - \lambda_i'\, dx \leq \sum_{i=1}^{n} \iint_{R_i} d(x, p_i) - \lambda_i\, dx - g_i(\lambda_i' - \lambda_i)\,.$$

Consider the right-hand side of the above; for each i, we have

$$\begin{aligned} & \iint_{R_i} d(x,p_i) - \lambda_i \, dx - g_i(\lambda_i^{'} - \lambda_i) \\ = & \iint_{R_i} d(x,p_i) - \lambda_i \, dx - (\lambda_i^{'} - \lambda_i) \iint_{R_i} dx \\ = & \iint_{R_i} d(x,p_i) - \lambda_i^{'} \, dx \end{aligned}$$

and therefore we have

$$\sum_{i=1}^{n} \iint_{R_i} d(x,p_i) - \lambda_i^{'} \, dx \geq \sum_{i=1}^{n} \iint_{R_i^{'}} d(x,p_i) - \lambda_i^{'} \, dx$$

since by construction, the sub-regions of the partition $\{R_1^{'}, \ldots, R_n^{'}\}$ are defined by looking at the minimal value of $d(x,p_i) - \lambda_i^{'}$ and are therefore minimal over all partitions. This completes the proof. □

We thus have in hand a fast method for computing supergradients for problem (4.7). We can therefore solve this problem quickly using an *analytic-center cutting plane method* (ACCPM), as given in Algorithm 4.4.1. The basic idea of this algorithm is to identify a polyhedron that is guaranteed to contain the optimal mulitplier λ^* and to subsequently reduce the size of this polyhedron at each iteration of the algorithm. The supergradient given by Theorem 4.4.2 provides a *cutting plane* which helps reduce the size of the polyhedron at every iteration step.

Remark 4.4.3. It is mentioned in the concluding remarks of [6, 7] that, for the special case where R is a convex polygon,

> "[I]t would be interesting to devise heuristics to approximate the additive-weighted Voronoi diagram that induces an optimal subdivision."

Algorithm 4.4.1 thus gives an affirmative answer to precisely this question as well as giving a general solution for non-convex regions.

Input: A polygonal region R with area 1 containing a collection of obstacles $O_1, \ldots, O_m$, a collection of points $P = \{p_1, \ldots, p_n\} \subset R$, and a threshold ϵ.

Output: A partition of R into n compact relatively star-convex regions $R_1, \ldots, R_n$, such that $p_i \in R_i$ and $|\operatorname{Area}(R_i) - \epsilon| \in \mathcal{O}(\epsilon)$ for all i.

Note: this is simply a standard analytic center cutting plane method applied to problem (4.7).

Define the initial polyhedron by
$\Lambda = \left\{\boldsymbol{\lambda} \in \mathbb{R}^n : \mathbf{1}^T\boldsymbol{\lambda} = 0 \text{ and } |\lambda_i - \lambda_j| \leq d(p_i, p_j)\right\}$;
while $\operatorname{vol}(\Lambda) > \epsilon$ **do**
  Let $\boldsymbol{\lambda}^0$ be the analytic center of Λ;
  Construct the strict dominance regions R_i^+ with multiplier vector $\boldsymbol{\lambda}^0$;
  Allocate the remaining area in ambiguous dominance regions lexicographically; that is, assign region R_k^- to the facility i with minimal index $i \in \mathcal{I}_k$;
  `/* This lexicographic allocation will not generally be feasible for the original partitioning problem. */`
  for $i \in \{1, \ldots, n\}$ **do**
    Set $g_i := \operatorname{Area}(R_i)$;
  end
  Set $\Lambda := \Lambda \cap \{\boldsymbol{\lambda} : \mathbf{g}^T\boldsymbol{\lambda} \geq \mathbf{g}^T\boldsymbol{\lambda}^0\}$;
end
Let $\boldsymbol{\lambda}^0$ be the analytic center of Λ;
Construct the strict dominance regions R_i^+ with multiplier vector $\boldsymbol{\lambda}^0$;
Allocate the remaining area in ambiguous dominance regions so that $\operatorname{Area}(R_i) = 1/n$ for all i;
return $\{R_1, \ldots, R_n\}$;

Algorithm 4.4.1: Algorithm ObstaclePartition(R, O, P) partitions a region with obstacles into compact relatively star-convex sub-regions.

4.5 Computational Complexity

Since Algorithm 4.4.1 is simply an analytic center cutting plane method (whose complexity is well-understood [59]) for solving the convex problem (4.7), it will suffice to discuss the computational complexity of each iteration. For any given vector $\boldsymbol{\lambda}$, we can construct the corresponding strict and ambiguous dominance regions (i.e. the weighted Voronoi diagram) using the *continuous Dijkstra paradigm* in $\mathcal{O}(N^{5/3})$ steps, where N is the total number of vertices of R and the obstacles [87]. When $f(\cdot)$ is the uniform distribution on R, we can also compute the areas of these cells in $\mathcal{O}(N^{5/3})$ steps and therefore each iteration requires only $\mathcal{O}(N^{5/3})$ computations. When $f(\cdot)$ is non-uniform, the computational com-

plexity depends on the chosen numerical integration scheme. To describe this we use the following result (originally given in Section 7.4 of [70], but re-stated using the language of Section 9.9 of [107]):

Theorem 4.5.1. *Let $\Omega \subset \mathbb{R}^2$ be a domain of integration equipped with a triangulation $\mathcal{T}_h$ consisting of N_T triangles, where h is the maximum edge length in $\mathcal{T}_h$. There exists a positive constant K_1, independent of h, such that the error E induced using either the composite midpoint formula*

$$\iint_\Omega f(x)\,dA \approx \sum_{T\in\mathcal{T}_h} \text{Area}(T) f(\text{centroid}(T))$$

or the composite trapezoidal formula

$$\iint_\Omega f(x)\,dA \approx \frac{1}{3}\sum_{T\in\mathcal{T}_h} \text{Area}(T) \sum_{j=1}^{3} f(\text{vertex}_j(T))$$

is bounded by

$$|E| \leq K_1 h^2\,\text{Area}(\Omega) M_2\ ,$$

where M_2 is the maximum value of the modules of the second derivatives of the integrand $f(\cdot)$.

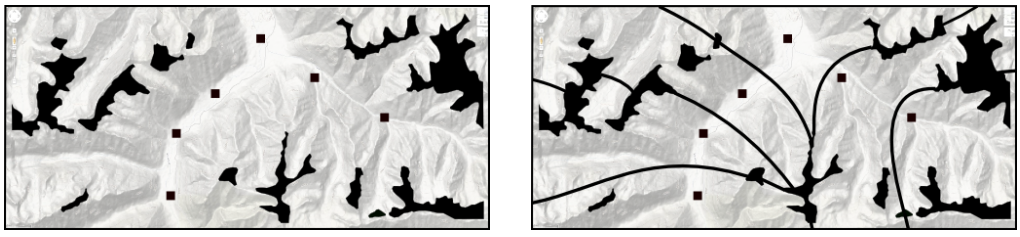
It follows that, if our desired error in integration is ϵ in our problem, then the maximum length of any edge in the triangulation must be at most

$$\epsilon^{1/2}(\text{Area}(R) M_2 K_1)^{-1/2}$$

(since we have hyperbolic arcs separating the R_i, an exact triangulation is impossible, but the added computational complexity therein is beyond the scope of this work). If we break R into N_T triangles, the maximum edge length will generally be $\mathcal{O}(\sqrt{\text{Area}(R)/N_T})$ and we therefore need to break R into $\mathcal{O}(\text{Area}(R)^2 M_2/\epsilon)$ triangles. Thus, the added complexity of non-uniform $f(\cdot)$ is quadratic in $\text{Area}(R)$, linear in the maximum modules of the second derivatives of $f(\cdot)$, and inversely proportional to the desired precision ϵ. Each iteration of Algorithm 4.4.1 therefore requires $\mathcal{O}(N^{5/3} + \text{Area}(R)^2 M_2/\epsilon)$ total steps.

4.6 Computational Results

In this section, we present the results of numerical simulation based on a practical military application. We consider a military conflict zone which is located in a hilly region. Unmanned aerial vehicles (UAV's) are used for aerial surveillance to monitor activities in such regions. A fixed number of UAV's are available for this purpose. Our task is to strategically divide the area into sub-regions such that the total work load for all the UAV's is minimized, i.e. such that all regions have equal area, as in the second example of Section 4.2. Typical low-cost UAV's used by the military for aerial surveillance have a *service ceiling*, or maximum usable altitude, of 5000 meters. As a sample dataset we use the troubled region of Jammu & Kashmir close to the India-Pakistan border in the Himalayan mountains, whose high altitude peaks pose an obstruction for the UAV's. Using a contour map of the Drass Sector obtained from Google Maps, we set all regions with an altitude exceeding 5000m as "obstacles" for the UAV's and built polygonal approximations for them. We choose six UAV launch sites randomly in the basin of the region since one generally prefers a flat terrain to launch such vehicles. The setup to this simulation is shown in Figure 4.5a. In Figure 4.5b, we show the equal-area partitions as computed by Algorithm 4.4.1. Figure 4.6 shows performance of the ACCPM method used in the algorithm which converges in 64 iteration steps. The figure in the left shows the normalized areas of the regions that are produced in the various iterations of our algorithm and the figure in the right shows the primal and dual objective function values.



(a) Input to our simulation. The black shaded areas correspond to obstacles, i.e. those regions with an altitude exceeding 5000m

(b) Output of our simulation

Figure 4.5: Optimal partition of Drass-sector

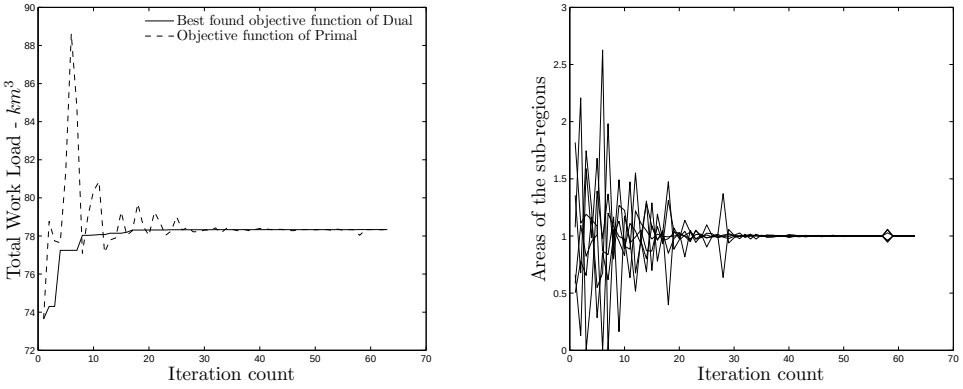
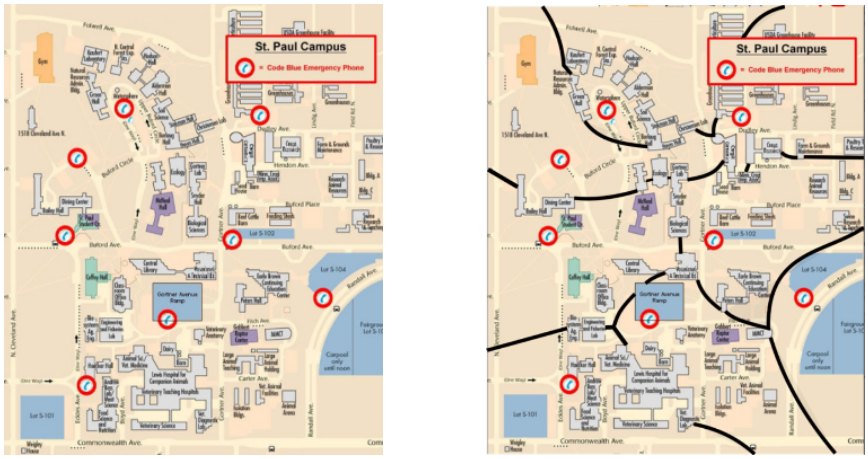


Figure 4.6: Performance of ACCPM in Algorithm 4.4.1for the Drass map.

The second example involves designing districts for Code Blue emergency telephones within a university campus. Code Blue emergency telephones are located throughout many university campuses for safety enhancement. In our simulation, we assume that the phones are located in fixed positions, as obtained from a map of the St. Paul campus of the University of Minnesota; the input is shown in Figure 4.7. We consider the buildings in the area as impenetrable obstacles. Assuming that the demand for these phones arises uniformly throughout the campus, our aim is therefore to divide the map into compact sub-regions so that each sub-region contains a phone and all sub-regions have equal area. The performance of the cutting plane method is illustrated in Figure 4.8.



(a) Input to our simulation. (b) Output of our simulation

Figure 4.7: Optimal partition of St. Paul campus of University of Minnesota

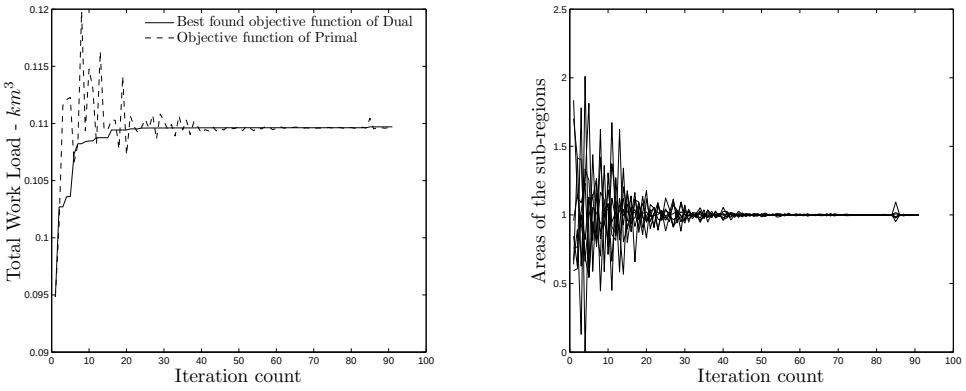


Figure 4.8: Performance of ACCPM in Algorithm 4.4.1for the St. Paul map.

Chapter 5

Non-Geographic Applications of Map Segmentation Problems

5.1 Economic Application

Our analysis of the max-sum (2.1) and the max-min (2.2) problems in Chapter 2 can actually be applied to non-geographic contexts. In this section, we apply Theorems 2.3.1 and 2.3.3 to look at the interaction between an *aggregate demand system* and an *address model*, which are commonly encountered in discrete choice theory in economics. Section 5.1.1 combines the results of Chapter 2 with the theory developed in [4] to show how to efficiently compute a market-clearing price vector in a variety of demand models for a differentiated product. Section 5.1.2 describes a variant on the classical *Fisher exchange market* [58] in which consumers are distributed according to a continuum.

Notational convention We use a double integral $\iint_R f(x)\,dA$ to denote integration over a planar region whereas we use a triple integral $\iiint_R f(x)\,dV$ to denote integration over a domain of arbitrary dimension.

5.1.1 Computing a Market-Clearing Price Vector in an Aggregate Demand System

Let $\mathbf{p} = (p_1, \ldots, p_n)$ denote the prices of n variants of a differentiated product. An *aggregate demand system* (ADS) $\mathbf{D} : \mathbb{R}^n_+ \to \mathbb{R}^n_+$ is a vector-valued function $\mathbf{D}(\mathbf{p})$ such that $D_i(\mathbf{p})$ represents the total demand for variant i from a given

population of consumers when the variants are priced according to $\mathbf{p}$. The paper [4] considers ADSs $\mathbf{D}(\cdot)$ satisfying four conditions:

(A1) $\mathbf{D}(\cdot)$ obeys the *gross substitutes property*:

$$\frac{\partial D_i}{\partial p_j} > 0 \qquad \forall i \neq j\,.$$

(A2) $\mathbf{D}(\cdot)$ is invariant under scalar addition:

$$\mathbf{D}(\mathbf{p} + c) = \mathbf{D}(\mathbf{p}) \qquad \forall c \in \mathbb{R}_+\,.$$

(A3) Aggregate demand for the product is constant:

$$\sum_{i=1}^{n} D_i(\mathbf{p}) = 1 \qquad \forall \mathbf{p} \in \mathbb{R}_+^n\,.$$

(A4) A technical constraint on the partial derivatives of $\mathbf{D}(\cdot)$:

$$\varphi(p_1 - p_n, \ldots, p_{n-1} - p_n) := \frac{\partial^{n-1} D_i}{\partial p_1 \cdots [\partial p_i] \cdots \partial p_n} > 0$$

where the right hand side is the $(n-1)$-th partial derivative of $D_i(\cdot)$ with respect to all prices except for p_i.

As explained in that paper, conditions (A1)-(A4) are satisfied under many standard discrete choice models, such as the *logit*, *probit*, *linear probability*, and *CES* models. One can also relax condition (A3) by regarding the *fraction* of demand attributed to each of the n variants rather than the true aggregate demand.

An alternative model to the ADS is the *address model*, defined as follows:

(B1) Each of the n product variants is represented as a point x_i in a "characteristics space" $\mathbb{R}^m$.

(B2) There is a continuum of consumers distributed in $\mathbb{R}^m$ according to a continuous and strictly positive density function $f(x)$, with

$$\iiint_{\mathbb{R}^m} f(x)\, dV = 1\,,$$

where "dV" denotes a volume differential in $\mathbb{R}^m$.

(B3) Each consumer purchases one unit of the variant that offers the greatest utility. The utility of a consumer located at x, purchasing variant i, is

$$u_i(x) = \alpha_i - \|x - x_i\|^2 - p_i$$

where α_i is a perceived "quality index" of variant i and p_i is the price of variant i as in the ADS.

Under the address model, we then see that the total demand for variant i, written $\tilde{D}_i(\mathbf{p})$, is

$$\tilde{D}_i(\mathbf{p}) = \iiint_{R_i} f(x)\, dV\,, \tag{5.1}$$

where R_i is the "market space" for variant i, i.e. the region for which variant i offers the greatest utility:

$$R_i = \left\{x \in \mathbb{R}^m : \alpha_i - \|x - x_i\|^2 - p_i \geq \alpha_j - \|x - x_j\|^2 - p_j \,\forall j \neq i\right\}\,.$$

Note that for any price vector $\mathbf{p}$, the partition $R_1, \ldots, R_n$ is simply a *power diagram* as described in Section 2.2. We will assume without loss of generality that $\alpha_i = 0$ for all i.

The major insight of [4] is that there exists an *equivalence* between the ADS and address models of demand:

Theorem 5.1.1 (ADS-address equivalency)**.** *Given any ADS* $\mathbf{D}(\cdot)$ *satisfying conditions (A1)-(A4), there exists a density function* $f(\cdot)$ *and a placement of points* $\{x_1, \ldots x_n\} \subset \mathbb{R}^m$ *(where it turns out that* $m = n - 1$*) such that* $\tilde{\mathbf{D}}(\mathbf{p}) = \mathbf{D}(\mathbf{p})$ *for all* $\mathbf{p} \in \mathbb{R}^n_+$*, with* $\tilde{\mathbf{D}}(\cdot)$ *as defined in (5.1).*

It is also worth noting that the equivalence established above is *constructive*; the authors give a closed-form expression for the placement of points x_i and the consumer density function $f(\cdot)$ in terms of $\mathbf{D}(\cdot)$. We can apply our Theorem 2.3.1 to the above result to show that, given an ADS $\mathbf{D}(\cdot)$ and a vector $\mathbf{d} \in \mathbb{R}^n_+$ such that $\sum_i d_i = 1$, we can easily find a *market-clearing price vector*, that is, a vector $\mathbf{p}^*$ such that $D_i(\mathbf{p}^*) = d_i$ for all i. Conceptually, we construct the placement of points $\{x_1, \cdots, x_n\}$ and the consumer density function $f(\cdot)$ as in the above theorem, then solve an instance of problem (2.1) in which we set

$R = \mathbb{R}^m$, $q_i = d_i$, and $u_i(x) = -\|x - x_i\|^2$. We find that the optimal solution $\boldsymbol{\lambda}^*$ to the dual problem (2.7) is precisely the desired price vector $\mathbf{p}^*$.

What is more striking, however, is that we can in fact solve problem (2.1) without taking any integrals whatsoever! Recall from Section 2.4.1 that the key to solving (2.1) is that we can easily construct a subgradient vector $\mathbf{g}$ by defining

$$g_i := -\iiint_{R_i} f(x)\, dV\,.$$

However, by construction, we know that $\iiint_{R_i} f(x)\, dV = \tilde{D}_i(\mathbf{p}) = D_i(\mathbf{p})$, and thus the vector $-\mathbf{D}(\mathbf{p})$ is *itself* a subgradient vector for the dual problem (2.7). To conclude this section, Algorithm 5.1.1 describes formally how to find $\mathbf{p}^*$ given the ADS $\mathbf{D}(\cdot)$.

Input: An ADS $\mathbf{D}(\cdot)$ that satisfies (A1)-(A4), a vector $\mathbf{d} \in \mathbb{R}^n_+$ such that $\sum_i d_i = 1$, and a threshold ϵ.
Output: A *market-clearing price vector*, i.e. a vector $\mathbf{p}^* \in \mathbb{R}^n_+$ such that $\|\mathbf{D}(\mathbf{p}^*) - \mathbf{d}\| \in \mathcal{O}(\epsilon)$.

Define the initial polyhedron by $\mathcal{P} = \{\mathbf{p} \in \mathbb{R}^n : \mathbf{d}^T\mathbf{p} = 0 \text{ and } \|\mathbf{p}\|_\infty \leq M$ for a threshold M;
`/* See Lemma A.1.1 of the Appendix to see how to construct a suitable` M. `*/`
while vol$(\mathcal{P}) > \epsilon$ **do**
  Let $\mathbf{p}^0$ be the analytic center of $\mathcal{P}$;
  Set $\mathbf{g} := -\mathbf{D}(\mathbf{p}^0)$;
  Set $\mathcal{P} := \mathcal{P} \cap \{\mathbf{p} : \mathbf{g}^T\mathbf{p} \geq \mathbf{g}^T\mathbf{p}^0\}$;
end
Let $\mathbf{p}^0$ be the analytic center of $\mathcal{P}$;
Set $\mathbf{p} := \mathbf{p}^0 - \min_i\{p^0_i\}$ (we do this so as to ensure that $\mathbf{p} \in \mathbb{R}^n_+$);
return $\mathbf{p}$;

Algorithm 5.1.1: Algorithm MarketClearingADS finds a market-clearing price vector in an ADS.

5.1.2 Relating Max-Sum and Max-Min Problems

Using Theorems 2.3.1 and 2.3.3, we can show that the optimal solutions to problems (2.1) and (2.2) are related in a particular fashion:

Claim 5.1.2. Let $R^*_1, \ldots, R^*_n$ be an optimal solution to problem (2.2) with generic utility functions $\bar{u}_i(x)$ and define $q^*_i := \iint_{R^*_i} f(x)\, dA$ for all i. Then $R^*_1, \ldots, R^*_n$ is also an optimal solution to problem (2.1) with the same density $f(\cdot)$ and utility functions $u_i(x) = \log \bar{u}_i(x)$ with $q_i = q^*_i$.

Proof. This is immediate; $R_1^*, \ldots, R_n^*$ satisfies the KKT conditions for problem (2.1), as follows from our proofs of Theorems 2.3.1 and 2.3.3. □

This result seems to be relevant in general problems of fair allocation of divisible goods [34], although we are not aware of its existence elsewhere in the literature. One particular application arises when we consider a continuous version of the classical *Fisher exchange market*, as discussed below.

Fisher's Exchange Market

Fisher's exchange market [58] is a special case of the general *Arrow-Debreu problem* [9] in which an economy consists of producers and consumers. A collection of N consumers have money to buy goods and maximize their (linear) utility functions; a collection of n producers sell their goods for money. We assume that each producer produces one unique good, so that the unit prices for the goods are denoted by a vector $\mathbf{p} \in \mathbb{R}_+^n$. Without loss of generality we assume that each producer has exactly one unit of his or her unique good to sell, which can be divided among the consumers. Associated with each consumer is a utility vector $\mathbf{u}_i \in \mathbb{R}_+^n$, a budget b_i, and a decision vector $\mathbf{x}_i \in \mathbb{R}_+^n$. Each agent chooses a bundle of goods by solving the simple linear program

$$\begin{aligned} \underset{\mathbf{x}_i}{\text{maximize}}\, \mathbf{u}_i^T \mathbf{x}_i \qquad & s.t. \\ \mathbf{p}^T \mathbf{x}_i &\leq b_i \\ \mathbf{x}_i &\geq \mathbf{0}\,. \end{aligned} \tag{5.2}$$

The objective of a market organizer is to determine a price vector $\mathbf{p}^*$ such that the market clears, i.e. that when each consumer selects his or her optimal decision vector $\mathbf{x}_i^*$, we have

$$\sum_{i=1}^{N} \mathbf{x}_i^* = \mathbf{e}$$

where $\mathbf{e} \in \mathbb{R}^n$ is a vector whose entries are all 1's. A classical result of Eisenberg and Gale [54] explains how to construct $\mathbf{p}^*$:

Theorem 5.1.3. *The optimal Lagrange multiplier for the equality constraints*

in the following optimization problem is a market-clearing price vector:

$$\begin{aligned} \underset{\mathbf{x}_1,\ldots,\mathbf{x}_N}{\text{maximize}} \sum_{i=1}^{N} b_i \log(\mathbf{u}_i^T \mathbf{x}_i) \qquad & s.t. \\ \sum_{i=1}^{N} \mathbf{x}_i &= \mathbf{e} \\ \mathbf{x}_i &\geq \mathbf{0} \quad \forall i\,. \end{aligned} \tag{5.3}$$

We will use a mild variant of Fisher's model in which agents are distributed according to a *continuum* $f(x)$ defined on a domain $R \in \mathbb{R}^m$, rather than by the index set $\{1,\ldots,N\}$ (here $\mathbb{R}^m$ would be a "characteristics space" in the language of Section 5.1.1). In this setting, an agent located at point $x \in R$ has a utility vector $\mathbf{u}(x) = (u_1(x),\ldots,u_n(x))$, a given budget $b(x)$, and a decision vector $\mathbf{J}(x) = (J_1(x),\ldots,J_n(x))$. The relevant equivalents of (5.2) and (5.3) are then

$$\begin{aligned} \underset{J_1(x),\ldots,J_n(x)}{\text{maximize}} \sum_{i=1}^{n} u_i(x) J_i(x) \qquad & s.t. \\ \sum_{i=1}^{n} p_i J_i(x) &\leq b(x) \\ J_i(x) &\geq 0 \quad \forall i \end{aligned} \tag{5.4}$$

and

$$\begin{aligned} \underset{J_1(\cdot),\ldots,J_n(\cdot)}{\text{maximize}} \iiint_R f(x) b(x) \log\left(\sum_{i=1}^{n} u_i(x) J_i(x)\right) dV \qquad & s.t. \\ \iiint_R f(x) J_i(x)\, dV &= 1 \quad \forall i \\ J_i(x) &\geq 0 \quad \forall i, x \end{aligned} \tag{5.5}$$

respectively.

Rather than clearing the market by selling fixed quantities of goods (which we would accomplish by solving (5.5)), we will show how to solve a related problem apropos of that encountered in Section 5.1.1: suppose that a market organizer wants to select a price vector $\mathbf{p}$ with the intention of setting, for each good i, the *fraction of customers that prefer good i to all other goods.* The

budgets of the agents, $b(x)$, and the amounts available of each good (which we assumed to be 1), are now disregarded. This is a useful model when the market consists of competing variants of products whose aggregate demand is *inelastic* (as opposed to the traditional Fisher market which is perfectly elastic), such as sanitation services or health insurance. For lack of a better phrase, we will call such a price vector an "allocation-clearing price vector".

Theorem 5.1.4. *Let* $\mathbf{q} \in \mathbb{R}^n_+$ *be a vector such that* $\sum_i q_i = 1$ *representing desired consumer allocations and suppose that* $f(x)$ *is a probability density function such that* $\iiint_R f(x)\,dV = 1$ *on a domain* $R \subset \mathbb{R}^m$*. Then the vector* $\mathbf{p}$ *defined by setting* $p_i = e^{\lambda_i^*}$ *for all* i*, where* $\boldsymbol{\lambda}^*$ *is the optimal Lagrange multiplier for the top equality constraints in the following optimization problem, is an "allocation-clearing price vector":*

$$\begin{aligned}
\underset{I_1(\cdot),\ldots,I_n(\cdot)}{\text{maximize}} \iiint_R f(x) \sum_{i=1}^{n} (\log u_i(x)) I_i(x)\,dV \qquad & s.t. \\
\iiint_R f(x) I_i(x)\,dV &= q_i \quad \forall i \\
\sum_{i=1}^{n} I_i(x) &= 1 \quad \forall x \\
I_i(x) &\geq 0 \quad \forall i, x\,.
\end{aligned} \tag{5.6}$$

Proof. Since (5.6) is an instance of (2.1), Theorem 2.3.1 says that at optimality, the region R_i^* (i.e. the region where $I_i^*(x) = 1$) consists of those points x where $\log u_i(x) - \lambda_i^* > \log u_j(x) - \lambda_j^*$ for all $j \neq i$, or equivalently, where $u_i(x)/e^{\lambda_i^*} = u_i(x)/p_i > u_j(x)/p_j$. Conversely, in order to maximize (5.4), a consumer located at point x will allocate its entire budget to precisely that same index i that maximizes $u_i(x)/p_i$. □

For the sake of completeness, problem (5.7) below is simply the discrete analogue of problem (5.6), where we assume that $\mathbf{q} \in \mathbb{R}^n_+$ satisfies $\sum_i q_i = N$. We believe this to be of independent interest because Theorem 5.1.4 still holds, provided we allow consumers to be fractionally allocated to two or more goods

if they are indifferent:

$$\underset{\mathbf{x}_1,\ldots,\mathbf{x}_N}{\text{maximize}} \sum_{i=1}^{N} \log(\mathbf{u}_i)^T \mathbf{x}_i \qquad s.t. \tag{5.7}$$

$$\begin{aligned} \sum_{i=1}^{N} \mathbf{x}_i &= \mathbf{q} \\ \mathbf{e}^T \mathbf{x}_i &= 1 \quad \forall i \\ \mathbf{x}_i &\geq 0 \quad \forall i\,. \end{aligned}$$

5.2 Data Visualization

Spatial partitions of an abstract *information space* are frequently used to visualize data. Weighted Voronoi diagrams are among the most popular ways of partitioning a space. In this section, we will show the use of weighted Voronoi diagrams to represent and visualize data. Specifically, our objective is to find an effective representation of an information space, such as a set of documents, as a planar diagram that conveys relevant information. Similar attempts to ours include [5, 37, 26, 47, 73, 113, 119]. In this setting, each document is represented as a region in the plane, in the same way as is shown in Figure 5.1. There are two major objectives that should be considered in designing such a diagram effectively: first, documents containing similar content should be placed in close geographic proximity to one another. Second, documents with larger significance or relevance should be represented by regions that are larger than those corresponding to documents with less significance.

Being based on the landmark points $\{p_1, \ldots, p_n\}$, the Voronoi framework lends itself well to the first objective described; that is, given a set of n documents and additional information regarding their relationships to one another, one can place the landmark points in a way that is commensurate with the relationships between the documents using graph visualization software such as GraphViz or Gephi. A less-studied problem is how to leverage the Voronoi framework in service of the second objective. To this end, a problem elegantly posed by Reitsma et al. [108] is as follows: suppose that the landmark points $\{p_1, \ldots, p_n\}$ are given in a region with area 1, together with a set of desired areas $\{A_1, \ldots, A_n\}$ that also sum to 1 (and which implicitly are related to the

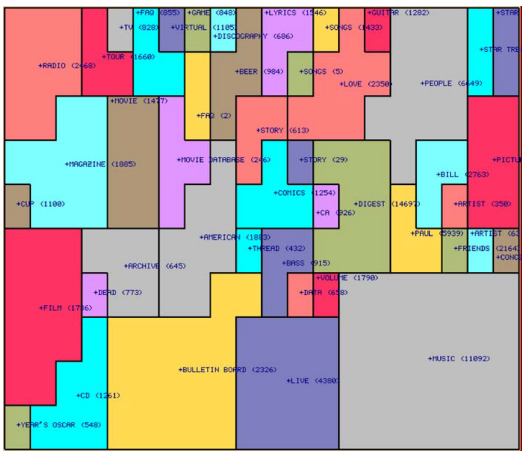


Figure 5.1: A planar map of an information space consisting of a set of documents, as constructed in [46].

significance or relevance of the n documents); can one find a weight vector $\mathbf{w}^*$ in the additive or multiplicative model such that $\text{Area}(V_i) = A_i$ for all i? The authors give an affirmative answer and describe an iterative, raster-based scheme for determining such a weight vector under the multiplicative model. The main drawbacks to this scheme relate to algorithmic efficiency, both in a practical and theoretical sense: as the algorithm is based on a form of fixed-point iteration, there is little in the way of performance guarantees, and consequently, the proposed methodology does not scale well as the problem size becomes large.

The techniques described in 2 give a fast algorithm for finding the desired weight vector $\mathbf{w}^*$ in either the additive or multiplicative model. In essence, we proved the following two results:

Theorem 5.2.1. *Given a planar region R and n fixed landmark points $\{p_1, ..., p_n\}$, one can compute the weights $\{w_1, \ldots, w_n\}$, which results in an additive Voronoi partition with sub-regions $\{V_1, \ldots, V_n\}$ that have areas $\{A_1, \ldots, A_n\}$, by solving the following convex optimization problem:*

$$\text{maximize}_{\boldsymbol{w}} \iint_R \min\{\|x - p_i\| - w_i\}\, dx \qquad s.t. \qquad (5.8)$$

$$\sum_{i=1}^{n} A_i w_i \;=\; 0\,.$$

Specifically, if $\boldsymbol{w}^*$ denotes the optimal solution to problem (5.8), then the

additive Voronoi sub-region V_i has an area A_i and consists of those points $x \in R$ for which $\|x - p_i\| - w_i^*$ is minimal:

$$V_i = \left\{ x \in R : \|x - p_i\| - w_i^* \leq \|x - p_j\| - w_j^* \ \forall j \right\} .$$

Theorem 5.2.2. *Given a planar region R and n fixed landmark points $\{p_1, ..., p_n\}$, one can compute the weights $\{w_1, \ldots, w_n\}$, which results in a multiplicative Voronoi partition with sub-regions $\{V_1, \ldots, V_n\}$ that have areas $\{A_1, \ldots, A_n\}$, by solving the following convex optimization problem:*

$$\begin{aligned} \text{maximize}_{\boldsymbol{w}} \iint_R \min\{\log \|x - p_i\| - w_i\} \, dx \qquad & s.t. \\ \sum_{i=1}^{n} A_i w_i \ &= \ 0 \, . \end{aligned} \tag{5.9}$$

Specifically, if $\boldsymbol{w}^*$ denotes the optimal solution to problem (5.9), then the multiplicative Voronoi sub-region V_i has an area A_i and consists of those points $x \in R$ for which $\log \|x - p_i\| - w_i^*$ is minimal:

$$V_i = \left\{ x \in R : \log \|x - p_i\| - w_i^* \leq \log \|x - p_j\| - w_j^* \ \forall j \right\} .$$

Rather than using fixed-point iteration, our approach is based on principles from variational calculus, specifically duality theory in linear programming over infinite-dimensional vector spaces, and thus inherits excellent theoretical and practical performance guarantees. Further advantages to this method is that, we can show how to apply a "homotopy method" to enable better control over Voronoi regions V_i (this technique was discussed in Section 3.1.1). We now demonstrate the effectiveness of our algorithm in computational experiments applied to a list of major internet sites.

5.2.1 Internet Traffic Visualization

In this section, we show the use of Voronoi space partitions to visualize real data. One of the most popular choices for information visualization is the amount of traffic on various internet websites. The information space is represented by a planar rectangle and the landmark points within this rectangle represent the

locations of internet websites. Once we partition this space, the size of a sub-region is proportional to the total number of page views on the corresponding website. Using Alexa Internet, Inc. (accessed June 7^{th} 2013), we obtained the amount of website traffic for the top 101 websites between the period September 2007 - June 2013. We first chose a subset of this data consisting of 13 popular websites (mainly social media, search engines and e-commerce) and we embed them on a 2-dimensional planar rectangle. A collection of such "maps" that display the relative "sizes" of these 13 major websites, taken between 2007 and 2013 are shown in Figures 5.2c to 5.2e. In these figures, websites with similar content are represented as regions that are in close proximity with one another, such as Facebook and Myspace. The location of the websites are held constants throughout the time period. The partitions are purely additively weighted Voronoi diagrams which produces hyperbolic boundaries. As we can see from the figures, Yahoo! and Myspace were dominant websites in September 2007, but as time progresses we can clearly visualize the decline of Yahoo! and Myspace which are engulfed by Google and Facebook respectively. As of June 2013, Google, Facebook and Youtube together account for more than 85% of the website traffic, relative to these 13 popular websites.

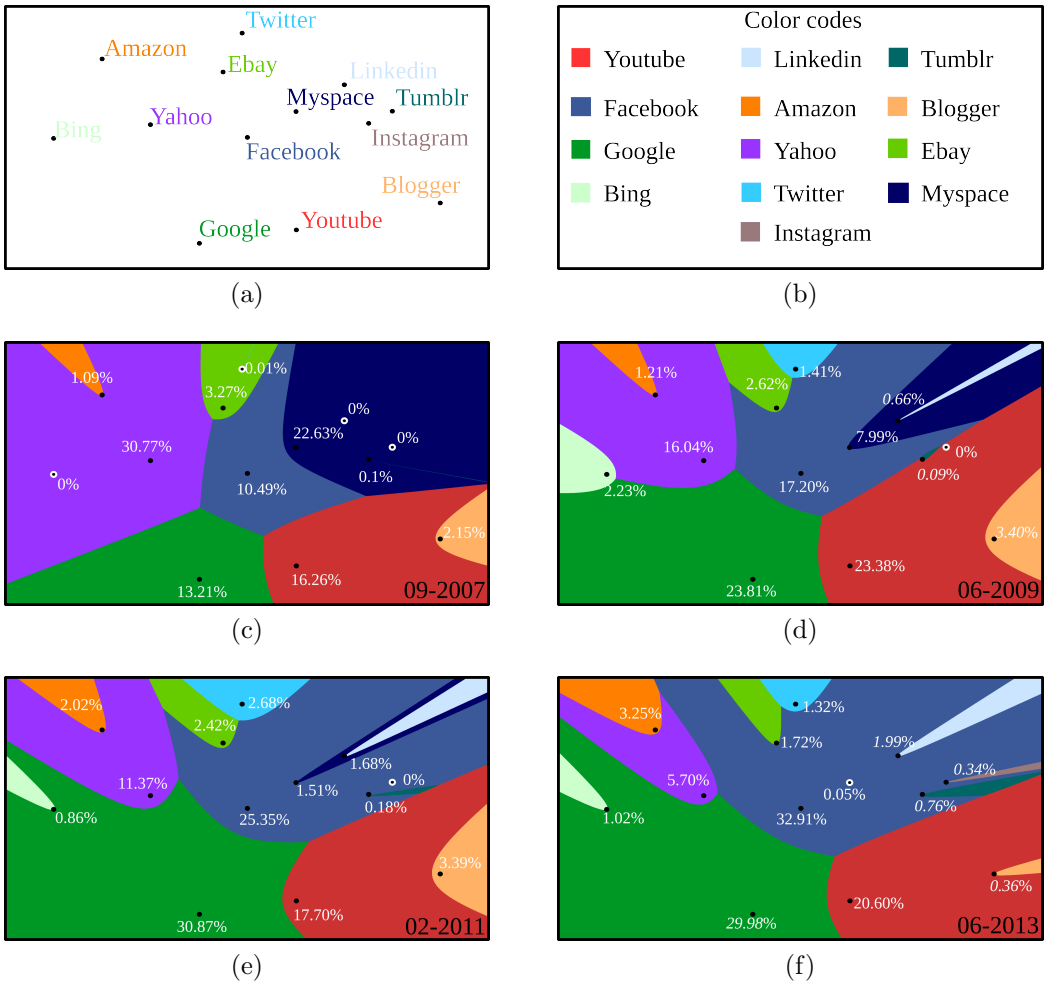
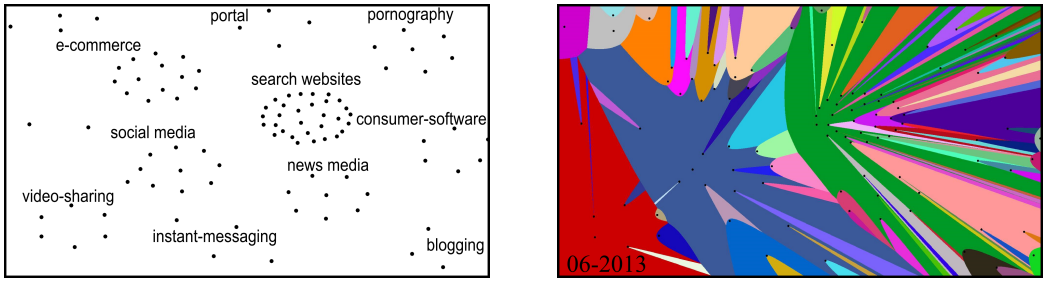


Figure 5.2: A collection of "maps" that display the relative "sizes" of 13 major websites, taken between 2007 and 2013. One can clearly visualize the dominance of Facebook and Google over MySpace and Yahoo! respectively.

We now represent the complete data set of 101 websites obtained from Alexa. Based on their genre, these websites were placed close to each other and their locations were obtained by using an open source tool, *Graphviz* (Graph Visualization Software). This is shown in Figure 5.3a. Based on the traffic information for these websites for June 2013, the map was partitioned using an additively weighted Voronoi diagram. These have a nice property that every landmark point is always guaranteed to be within its assigned sub-region and that all sub-regions are connected, but as we can see from Figure 5.3b, the sub-regions can become long and skinny. For regions with a very small fraction of the total area, this becomes even worse. This is clearly undesirable when visualizing data

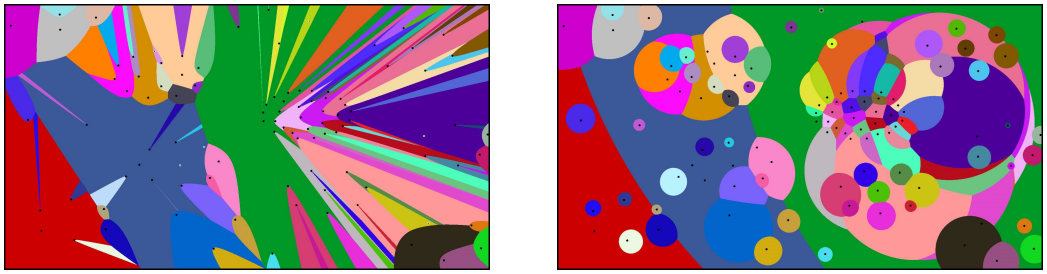
using partitions. One way to getting rid of this problem is by using multiplicative Voronoi diagrams. These have circular boundaries and inherently produce fat circular regions. A comparison between additive and multiplicative Voronoi diagram is shown in Figure 5.4.



(a) Locations

(b) A partitioned map of top 101 websites using additive Voronoi diagrams

Figure 5.3: Locations and partitions of top 101 websites clustered based on their genre (obtained using *Graphviz*). Multiple black dots in the same colored region indicates that websites have zero areas in the partition.



(a) A partitioned map of top 101 websites using additive Voronoi diagrams

(b) A partitioned map of top 101 websites using multiplicative Voronoi diagrams

Figure 5.4: Use of additive Voronoi diagram versus multiplicative Voronoi diagram.

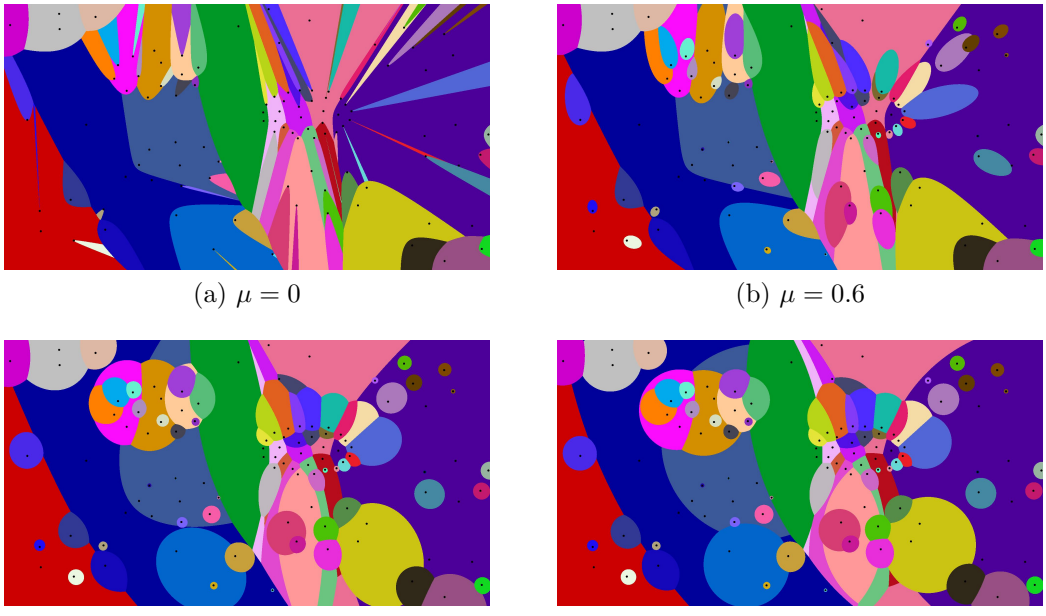
The drawback with using a multiplicative Voronoi partition is that it may happen that some of the sub-regions are disconnected (see Remark 2.6.1). To deal with this problem, we propose a "homotopy approach" to inherit both properties of additive and multiplicative Voronoi diagrams. This has been discussed exclusively in Section 3.1. We formulate a new optimization problem (5.10) that uses an objective function which is a weighted average of optimization problems presented in (5.8) and (5.9).

$$\text{maximize}_{\boldsymbol{w}} \iint_R \min\{\mu \log \|x - p_i\| + (1-\mu)\|x - p_i\| - w_i\}\, dx \qquad s.t. \quad (5.10)$$

$$\sum_{i=1}^{n} A_i w_i \qquad = 0\,.$$

Figure 5.5 shows the effect of this modified objective function. For $\mu = 1$ the partitions correspond to a multiplicative Voronoi diagram, in which sub-regions need not always be connected. But since, for $\mu = 0$ the regions are always connected (because they correspond to an additive Voronoi diagram), it is not hard to show that we can find a "threshold" $\mu \in [0, 1]$ for which the sub-regions are always connected.

We hence, create partitions based on a combination of additive and multiplicative Voronoi diagram using the homotopy method presented above. A collection of such maps between the time period September 2007 and June 2013 are shown in Figure 5.6 (multiple black dots in the same colored region indicates that websites have zero areas in the partition).



(a) $\mu = 0$ (b) $\mu = 0.6$

(c) $\mu = 0.8$ (d) $\mu = 1$

Figure 5.5: Homotopy method to control the fatness of sub-regions while maintaining the areas of sub-regions. By increasing the penalty term, we can increase the "fatness" the skinny regions.

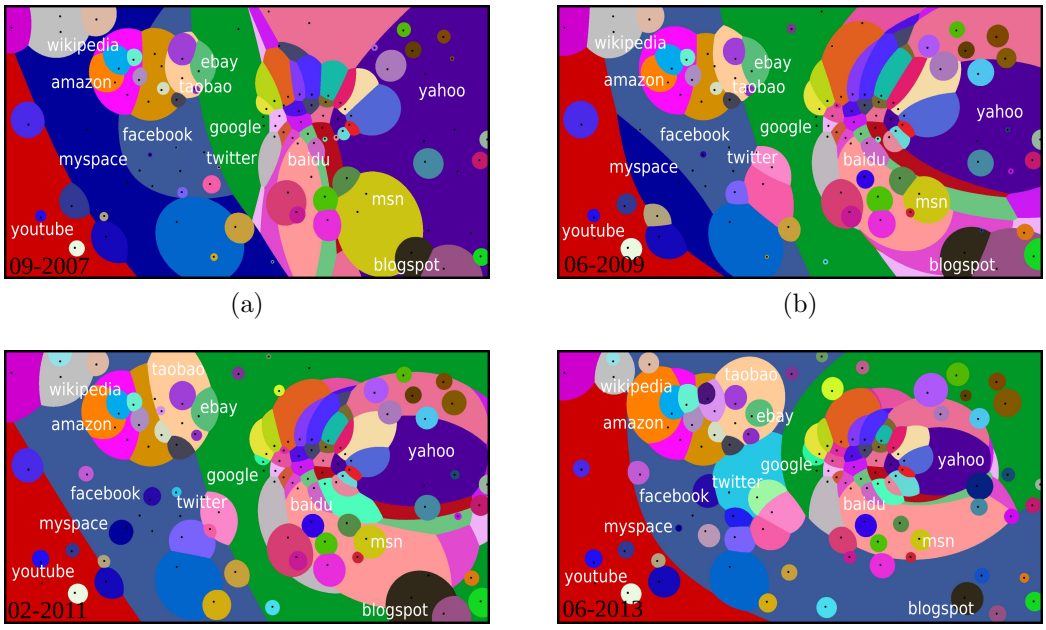


Figure 5.6: A collection of "maps" that display the relative "sizes" of the top 101 major websites, taken between 2007 and 2013.

Chapter 6

The k-Medians Problem

6.1 Introduction

A closely related and well studied class of problems is the facility location problem. The optimization variable here is the location of facilities themselves while trying to minimize the operational costs. Based on how one models the operational costs, several variants of this problem can be formulated. A frequently occurring and well studied problem is the k-medians problem, also called the *multi-source Weber problem* [36]. Here, the objective is to select a set of k "landmark" points so as to minimize the total distance between the landmark points and some other set of "client" or demand points. The most natural setting for this problem is to let the clients be a discrete set of points in the plane, which was proved to be NP-hard in [98]. The papers [8] and [86] both describe PTASes for this case. In a general metric space, [45] describes a factor $6\frac{2}{3}$ approximation algorithm.

Another setting is the case where client points form a continuum. This is a natural problem one encounters in facility location where the number of customers is usually large (say over $100,000$), thereby rendering the corresponding discrete k-medians problem intractable. The objective function in a continuum demand case is popularly known as the *Fermat-Weber* value of the input region. Given a planar region C and a point p, we denote the integral of distances of all points in C relative to p with $\mathrm{FW}(C,p)$. Let p^* be the point for which this quantity, $\mathrm{FW}(C,p)$, is minimized. We call $\mathrm{FW}(C,p^*)$ as the Fermat-Weber of region C and denote it with $\mathrm{FW}(C)$. The point p^* is known as the Fermat-Weber

center or 1-median of C.

$$\begin{aligned} \mathrm{FW}(C,p) &= \iint_C ||x-p||\,dx \\ \mathrm{FW}\,(C) &= \min_p \iint_C \|x-p\|\;dx \end{aligned}$$

Here, $\|\cdot\|$ denotes the Euclidean norm. For simple convex objects like, disk and rectangle, the Fermat-Weber center is the geometric centroid of the object itself. The Lemma′ below lists Fermat-Weber values of some simple convex regions.

Lemma 6.1.1. *For a disk D with radius r,*

$$\mathrm{FW}\,(D) = \frac{2\pi r^3}{3}\,.$$

Also, for an angle $\theta \in [0, 2\pi]$, if S denotes the circular sector of D subtended by θ, then

$$\mathrm{FW}\,(S,c) = \frac{\theta}{3}r^3 = \frac{2\cdot \mathrm{Area}\,(S)}{3}r$$

where c denotes the center of the disk of which S is a sector.

Proof. Trivial. □

Remark 6.1.2. It is well-known that, for a fixed area, the disk is the region with minimal Fermat-Weber value $\mathrm{FW}\,(C)$. Using 6.1.1, this gives us an easy lower bound:

$$\mathrm{FW}\,(C) \geq \frac{2}{3\sqrt{\pi}}A^{3/2}$$

where A is the area of C.

Lemma 6.1.3. *For a rectangle R of height h and width w,*

$$\mathrm{FW}(R) = \frac{1}{6}hw\sqrt{h^2+w^2} + \frac{1}{12}w^3\log\left(\frac{\sqrt{h^2+w^2}+h}{w}\right) + \frac{1}{12}h^3\log\left(\frac{\sqrt{h^2+w^2}+w}{h}\right) \tag{6.1}$$

Proof. The above result can be obtained by evaluating the following integral analytically:

$$\mathrm{FW}(R) = \int_{\frac{-h}{2}}^{\frac{h}{2}}\int_{\frac{-w}{2}}^{\frac{w}{2}} \sqrt{x_1^2+x_2^2}\,dx_1\,dx_2$$

□

Lemma 6.1.4. *For a right triangle $T = \triangle ABC$ whose right angle is located at the point C, with sides $AB = c$, $BC = a$ and $CA = b$, the Fermat Weber value of the triangle relative to point B is given by*

$$\mathrm{FW}(ABC) = \frac{1}{6}abc + \frac{1}{6}a^3 \log\left(\frac{c+b}{a}\right) \tag{6.2}$$

Proof. This is again merely analytic integration. □

It turns out that finding the 1-median of a general convex region is not trivial at all. A recent survey of the existing literature reveals that there is limited work considering the k-medians problem in a continuous demand setting. And even fewer of those consider the problem in an Euclidean norm. The first exact algorithmic study this problem was performed in [55], which describes algorithms for finding optimal solution for 1-median problem for simple polygon or a polygon with holes, under the L_1 norm. Carmi et al. [44] provide a linear-time approximation scheme for finding an approximate L_2 norm Fermat-Weber center of a convex polygon by establishing a lower bound on the average distance of a convex object in terms of its diameter. Authors in [1] improve on these bounds and as a corollary, they give a very simple linear time 2.41-constant factor scheme to compute Fermat-Weber center of a convex polygon.

In this work, we are interested in the multiple-center version of the Fermat-Weber problem. We seek to find k points, $P = \{p_1, ..., p_k\}$, that minimizes the integral of distances of all points in C relative to these k points (demand point is assigned to its nearest landmark point). We denote this objective function using $\mathrm{FW}(C, k)$ which is given by the following integral.

$$\mathrm{FW}(C, k) = \min_{P:|P|=k} \iint_C \min_i \|x - p_i\| \, dx$$

Authors in [55] proved the L_1 norm version of the continuous k-median problem to be NP-hard for large k. The paper [43] is the only work that currently exists that attempts to solve the k-medians problem in a convex polygon. They provide a recursive sub-division technique which they prove is a 2.73 constant factor approximation algorithm for the same problem.

In this chapter, we first consider the algorithm that was presented in [43] for

convex polygons. We develop new upper and lower bounds to the k-medians problem and improve the approximation ratio of their algorithm. When the input is non-convex (including regions with holes), we propose a local optimal solution wherein we use the partitioning algorithm described in Chapter 4 as a sub-routine in Lloyd′s algorithm. This variation of Lloyd′ algorithm appears to perform well in practice, but finding bounds on its performance is non-trivial and still remains an open problem.

6.2 The k-Medians Problem in a Convex Polygon

We first look at solving the k-medians problem when the input region is a convex polygon. We begin by establishing lower and upper bounds of Fermat-Weber of a convex region C, FW(C), in terms of its area and a bounding box that is known to contain C. In Section 6.2.2, we briefly describe the approximation algorithm that was presented in [43]. We then use the bounds on FW(C) to improve the factor of approximation from 2.73 to 2.03.

6.2.1 Bounds on Fermat-Weber of a Convex Polygon

Given a convex polygon C of area A, we want to find upper and lower bounds on FW(C). The main motivation to find these bounds is that, we want to use them in finding upper and lower bounds for the k-medians problem (FW(C, k)). Note that the papers [44], [1] and [52] almost entirely deal with finding upper and lower bounds on FW(C) in terms of A and d, the diameter of C. Summarized in one line, they prove the following bounds: $0.16dA \leq \text{FW}(C) \leq 0.3490dA$.

Unfortunately, these bounds are not useful in our work for two reasons. First, a lower bound in terms of diameter of C does not help us achieve a lower bound on FW(C, k). And secondly, the upper bound can be significantly improved when the convex region is "skinny" (it is easy to show that the Fermat-Weber of a very long skinny rectangle is approximately $0.25dA$). The authors in [1] arrive at their upper bound by making using of the fact that a convex object of diameter d can always be contained in a circle of radius $d/\sqrt{3}$. Rather than using a bounding circle, we find it more useful to derive bounds in terms of a bounding box that contains C. This is because for a long and skinny region,

the area of the minimum bounding circle can be infinitely larger than the area of the convex region itself. But if one considers the minimum bounding box of C, we can prove that the ratio of its area to that of the convex polygon is at most 2 for any convex region (see equation (6.6)).

In the following sections, we will derive lower and upper bounds for a convex polygon C of area A that is contained within a rectangle of dimensions $w \times h$, where $A \in [0, wh]$ (this rectangle need not necessarily be the minimum bounding box of C). The Fermat-Weber value of the rectangle itself is denoted using $\mathrm{FW}_{\square}(w, h)$. Lemma 6.1.3 provides a closed form expression for $\mathrm{FW}_{\square}(w, h)$.

Lower Bounding $\mathrm{FW}(C)$

Here, we derive a simple lower bound that we will use to place an overall lower bound on $\mathrm{FW}(C, k)$. For a given area A, it is a well known fact that disk is the shape that minimizes its Fermat-Weber value. Hence, using the Fermat-Weber of a disk, we can arrive at an easy lower bound on $\mathrm{FW}(C)$: $\phi_{LB_1} = \frac{2}{3\sqrt{\pi}} A^{3/2}$ (see Remark 6.1.2). Combining this with the fact that region is contained within a box of dimensions $w \times h$, we can improve this lower bound.

Lemma 6.2.1. *Let C be a convex region with area A, contained in a box B of dimensions $w \times h$, with $w \geq h$. If B' is a horizontal slab of height h that contains B and D is a disk of radius r centered at the centroid of the box such that* $\mathrm{Area}\,(D \cap B') = A$*, then:*

$$\mathrm{FW}\,(C) \geq \begin{cases} \frac{2}{3}\pi r^3 & \text{if } r < \frac{h}{2} \\ \frac{4r^3}{3}\sin^{-1}\left(\frac{h}{2r}\right) + \frac{1}{3}rh\sqrt{r^2 - \frac{h^2}{4}} + \frac{1}{12}h^3 \log\left(\frac{2r+\sqrt{4r^2-h^2}}{h}\right) & \text{if } r \geq \frac{h}{2} \end{cases} \tag{6.3}$$

Proof. Refer to Figure 6.1 for this proof. The shape C^* that minimizes $\mathrm{FW}\,(C)$ in B' is the intersection of a disk of radius r with a slab of height h, and its Fermat-Weber value varies according to r and h.

Case 1: If $r < h/2$, then C^* is a disk and hence we have $FW\,(C^*) = \frac{2}{3}\pi r^3$ (from Lemma 6.1.1).

(a) Region C^* of minimal Fermat-Weber.

(b) Breaking C^* into triangles and sectors.

Figure 6.1: Lower bound for a convex region of area A within a rectangle of height h

Case 2: If $r \geq h/2$, then C^* is the intersection of a disk and a slab of height h, and the Fermat-Weber of such a region can be found out by evaluating the integral $\iint_{C^*} \sqrt{x_1^2 + x_2^2}\, dx_1\, dx_2$. An easy way to do this is to break C^* into regions consisting of right-angled triangles and circular sectors (Figure 6.1b) and then use Lemmas 6.1.1 and 6.1.4 to arrive at the lower bound. □

We will denote this lower bound using $\Phi_{LB_2}(A, h) = \mathrm{FW}(C^*)$. Note that Φ_{LB_2} is independent of the width of the rectangle (since we have assumed that $w \geq h$).

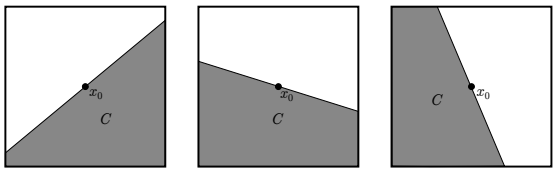
Upper Bounding $\mathrm{FW}(C)$

To find an upper bound on $\mathrm{FW}(C)$, we will answer the following question: What is the shape of the convex region of a given area A, that is contained within a box $w \times h$ and has the maximum Fermat-Weber quantity? Intuitively skinnier regions have larger Fermat-Weber values. But to find the exact shape of such a region is not easy. Interestingly, this becomes tractable if $A = wh/2$. In Lemma 6.2.2 we will provide an upper bound for the case $A = wh/2$. And in Lemma 6.2.5, we will extend this result to find an upper bound for all $A \in [0, wh]$. Here on, we will use $\mathrm{FW}_{1/2}(w, h)$ to denote $0.5\,\mathrm{FW}_{\square}(w, h)$ which is half the Fermat-Weber value of the rectangle.

Lemma 6.2.2. *Let C be a convex region of area A that is contained within a box B of dimensions $w \times h$. If $A = \frac{wh}{2}$, then we have:*

$$\mathrm{FW}(C) \leq \mathrm{FW}_{1/2}(w, h)$$

Proof. Let x_0 denote the center of B. By definition of Fermat-Weber, for any convex region C we have: $\mathrm{FW}(C) \leq \mathrm{FW}(C, x_0)$. Consider the family $\mathcal{C}$ consisting of all convex regions C in B which merely consist of the area lying to



(a) (b) (c)

Figure 6.2: It is obvious that all regions C above have the same Fermat-Weber value relative to x_0, namely $\mathrm{FW}_{1/2}(w, h)$.

one side of a line through x_0, as shown in Figure 6.2. It is obvious that, for any region $C' \in \mathcal{C}$, we have $\mathrm{FW}(C', x_0) = \mathrm{FW}_{1/2}(w, h)$ by a trivial symmetry argument. Therefore, it will suffice to show that we can take *any* convex region C in B with area $wh/2$ and transform it into a region $C' \in \mathcal{C}$ in a manner that does not decrease $\mathrm{FW}(C, x_0)$. This will basically prove our result: $\mathrm{FW}(C) \leq \mathrm{FW}(C, x_0) \leq \mathrm{FW}(\mathcal{C}, x_0) = \mathrm{FW}_{1/2}(w, h)$. We will show how to execute this transformation after making two straightforward observations:

Lemma 6.2.3. *Let s denote a line segment in the plane and let $\vec{v}$ be a vector in the plane, so that $s + t\vec{v}$ is a translation operator on s in the direction $\vec{v}$. Then the function $f(t) = \int_{s+t\vec{v}} ||x||\, dx$ is convex in t.*

Proof. The function $f(t)$ is simply the Fermat-Weber value of the line segment $s + t\vec{v}$ relative to the origin; it is an infinite sum of convex functions and is therefore itself convex. □

Lemma 6.2.4. *For any triangle T with vertices $\{x_1, x_2, x_3\}$, let $T(t)$ denote the sheared triangle $\{x_1, x_2, x_3 + t(x_2 - x_1)\}$. Then the function $g(t) = \mathrm{FW}(T(t), x_0)$ is convex in t for any fixed x_0.*

Proof. This is a corollary of Lemma 6.2.3 because $g(t)$ is an infinite sum of convex functions $f(t)$. □

We can now proceed to prove Lemma 6.2.2. Consider any convex region C with area $wh/2$, contained inside a box B with dimensions $w \times h$, as in the statement of Lemma 6.2.2.

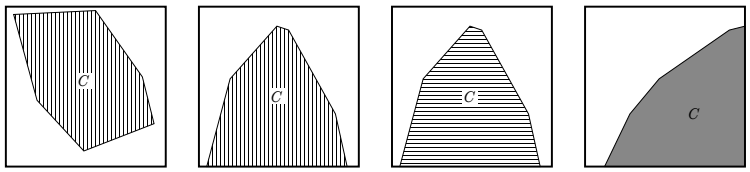


Figure 6.3: Two applications of Lemma 6.2.3 permit us to assume that C takes the shape shown in Figure 6.3d.

By visualizing C as a union of vertical line segments (Figure 6.3a),we can apply Lemma 6.2.3 to assume without loss of generality that C takes the shape shown in Figure 6.3b, i.e. that C is the area lying below some concave function (this is because the function defined in Lemma 6.2.3 is convex and therefore is maximized when each line segment is as far from the origin, x_0, as possible). By visualizing C as a union of *horizontal* line segments (Figure 6.3c) we can subsequently assume without loss of generality that C takes the shape shown in Figure 6.3d, i.e. that C is the area lying below some concave and increasing function. By using a series of shear transformations and iteratively applying Lemma 6.2.4, we can show that the region C takes the form shown in Figure 6.2 and in doing so, we would have only increased the Fermat-Weber value of C relative to x_0, the center of B.

We begin with Figure 6.3d and let $B^{'}$ denote the minimum bounding box of C. We iteratively apply Lemma 6.2.4 to shear the triangular components of C (all the while increasing the Fermat-Weber value of C relative to x_0, the center of B) until it takes the form shown in Figure 6.4a, to the point where it remains merely to shear the two remaining triangular components of C appropriately. We select one such triangular component arbitrarily and shear it in a direction that increases the Fermat-Weber value of C relative to the center x_0 of B (say, the point marked x in Figure 6.4b), until x either touches the bottom edge of B or the leftmost edge of B. Suppose that x touches the leftmost edge of B as shown in Figure 6.4c (the case where x touches the bottom edge of B can be addressed in a similar fashion and we omit it for brevity). It is clear that we can then shear x *downward* until it touches the corner of B as shown in Figure 6.4d. This downward shearing introduces a new vertex, which we call $x^{'}$ (also shown in Figure 6.4d), which we can shear in a direction that increases the Fermat-Weber value, until it either touches the top edge of $B^{'}$ or the left

edge of B. Suppose that $x^{'}$ touches the left edge of B (the case where $x^{'}$ touches the top edge of $B^{'}$ is similar and we omit it for brevity), shown in Figure 6.4e. This leaves us with one final vertex, $x^{''}$, which we can shear in a direction that increases the Fermat-Weber value of C relative to x_0; note that we are free to shear $x^{''}$ until it touches the boundary of B, at which point our shape C now has precisely the shape desired as in Lemma 6.2.2, as shown in Figure 6.4f. This completes the proof.

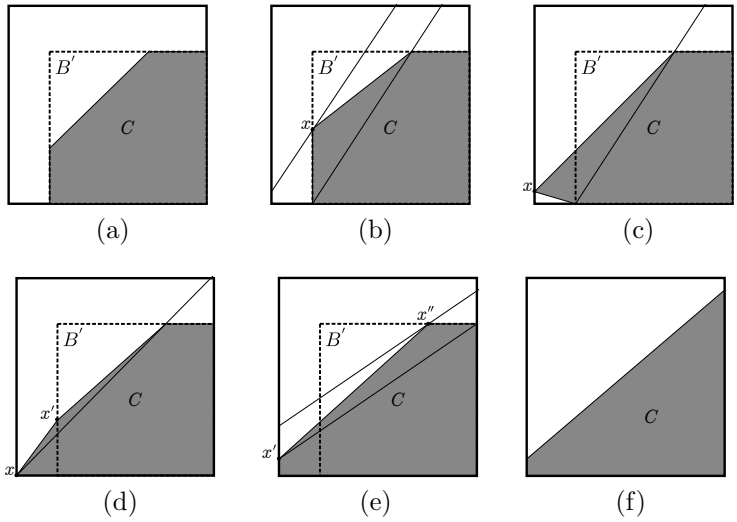


Figure 6.4: Final adjustments to C.

□

Lemma 6.2.5. *Let C be a convex region contained in a box B of dimensions $w \times h$ with $w \geq h$ such that C contains a horizontal line segment whose length is equal to that of width(C). If $A = \text{Area}(C)$, then we have:*

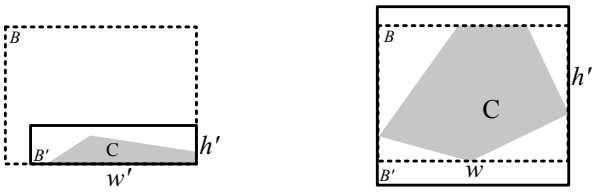
$$\text{FW}(C) \leq \text{FW}_{1/2}\left(w, \frac{2A}{w}\right) \tag{6.4}$$

Proof. We break up the proof into two separate cases:

Case 1: $A < wh/2$. Since $A < \frac{wh}{2}$ and since C contains a line segment whose length is width(C), we can always construct a smaller box B' with dimensions $w' \times h'$ containing the region C, such that $A = \frac{w'h'}{2}$ (see Figure 6.5a). Applying Lemma 6.2.2 for B', we find that $\text{FW}\,(C) \leq \text{FW}_{1/2}\,(w', h')$. By performing some

basic algebra, it is not hard to show that, among all values of w' and h' such that $w' \leq w$, $h' \leq h$ and $w'h' = 2A$, the value $FW_{1/2}(w', h')$ is maximized when $w' = w$ and $h = 2A/w$ (Fermat-Weber value of a rectangle increases with aspect ratio). Thus, we find that:

$$\mathrm{FW}(C) \leq \mathrm{FW}_{1/2}\left(w, \frac{2A}{w}\right)$$



(a) $A < wh/2$ (b) $A \geq wh/2$

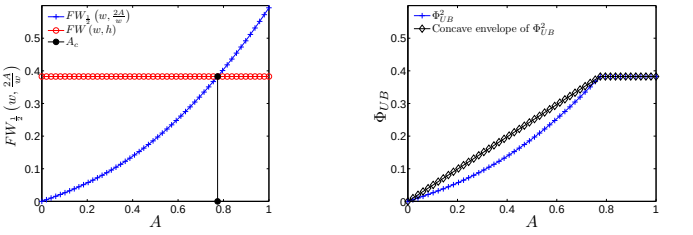
Figure 6.5: We can find a box B' that encloses region C such that $w'h' = 2A$, $\forall A \in [0, wh]$

Case 2: $A \geq wh/2$. Since $A \geq wh/2$, we can obviously find a larger rectangle B', which contains the region C such that $A = \frac{w'h'}{2}$. One easy way to do that is to increase the height of B to h' such that $A = \frac{wh'}{2}$ (see Figure 6.5b), which leads to $h' = \frac{2A}{w} \geq h$. Using Lemma 6.2.2 for B', we arrive at the upper bound which completes the proof. □

It is easy to verify that the upper bound of Lemma 6.2.5 is convex and monotonically increasing in A (see Figure 6.6a). It also has an undesirable property that, for some interval: $A \in (A_c, hw]$, we actually have $\mathrm{FW}_{1/2}\left(w, \frac{2A}{w}\right) > \mathrm{FW}_{\square}(w, h)$, which is the Fermat-Weber value of the rectangle itself. Therefore, $\Phi^2_{UB}(A, w, h) = \min\left\{\mathrm{FW}_{1/2}\left(w, \frac{2A}{w}\right), \mathrm{FW}_{\square}(w, h)\right\}$ will be a more meaningful upper bound . But this function, Φ^2_{UB}, (Figure 6.6b) is neither convex or concave in the desired region of interest. To help simplify the analysis in the later section, we choose the simplest *concave envelope* of the above function as our final upper bound, Φ_{UB} :

$$\Phi_{UB}(A, w, h) = \min\left\{\frac{A}{A_c}\mathrm{FW}_{\square}(w, h), \mathrm{FW}_{\square}(w, h)\right\} \tag{6.5}$$

where A_c is the solution of the equation: $\mathrm{FW}_{1/2}\left(w, \frac{2A_c}{w}\right) = \mathrm{FW}_{\square}(w, h)$. Note



(a) $FW_{1/2}\left(w, \frac{2A}{w}\right)$ (b) Concave envelope of Φ^2_{UB}

Figure 6.6: Upper bound for a convex region of area A within a unit square

that the ratio $\frac{A_c}{hw}$ depends only on the aspect ratio of the rectangle. And by construction, for a given w and h, Φ_{UB} is a piecewise linear concave function.

Remark 6.2.6. Few interesting observations on the Fermat-Weber center can be made at this point. Using Lemma 3.1 of [7] and the lower bound on $\mathrm{FW}(C) \geq 0.16dA$, it can be shown that choosing any point within a convex region is a constant factor-4.17 approximation for the 1-median problem. The center of the smallest enclosing circle of C is a factor 2.41 approximation of Fermat-Weber center, which was proved in [1]. Whereas, using bounds Φ_{UB} and Φ_{LB_2} we can show that the center of the minimum bounding box of a convex region is factor-2 approximation algorithm for the 1-median problem.

6.2.2 The Approximation Algorithm

We now describe the algorithm provided by authors in [43] for the continuous k-medians problem in a convex polygon. Let $\mathrm{AR}(C)$ denote the *aspect ratio* of C, $\mathrm{AR}(C) = \max\{w/h, h/w\}$ and let $\mathrm{diam}(C)$ denote the diameter of C. Our algorithm can be summarized in a single paragraph: the input to our algorithm is a convex polygon C with n vertices and an integer k. We assume without loss of generality that C is aligned so that its diameter coincides with the coordinate x-axis. We then enclose C in an axis-aligned minimum bounding box $\square C$ of dimensions $w \times h$, where $w = \mathrm{diam}(C)$. By aligning the diameter with the x-axis, we must have:

$$\frac{wh}{2} \leq \mathrm{Area}(C) \leq wh \tag{6.6}$$

Note that (6.6) holds for a general minimum bounding box of a convex region too (it is tight when C is a triangle). We now let $k_1 = \lfloor k/2 \rfloor$ and $k_2 = \lceil k/2 \rceil$ and divide $\square C$ into two pieces of areas $\frac{k_1}{k} \cdot \mathrm{Area}(\square C) = \frac{k_1}{k} \cdot wh$ and $\frac{k_2}{k} \cdot \mathrm{Area}(\square C) =$

$\frac{k_2}{k} \cdot wh$ respectively, using a vertical line. This is performed recursively (with the option to split using a horizontal line, if the height of an intermediate sub-region exceeds its width) until all rectangular regions have area $\text{Area}(\square C)/k$. The center of each sub-rectangle is chosen as our landmark points. It may happen that some of these points may lie outside the convex polygon (Figure 6.7c). Such points are simply relocated back into the polygon (Figure 6.7d). The way we to do this operation turns out to be inconsequential in the performance of the algorithm. The entire algorithm is explained in detail in Algorithms 6.2.1 and 6.2.2 and also in Figure 6.7.

Running time This algorithm can be performed with running time $\mathcal{O}(n + k + klogn)$. This is because Algorithm 6.2.1 takes $\mathcal{O}(k)$ operations to partition the rectangle and Algorithm 6.2.2 requires $\mathcal{O}(n)$ operations to find a minimum bounding box of C. The last step of Algorithm 6.2.2 consists of moving the center points to C when necessary, which takes $\mathcal{O}(k \log n)$ operations using a point-in-polygon algorithm [106].

Input: An axis-aligned rectangle R and an integer k.
Output: A partition of R into k rectangles, each having area $\text{Area}(R)/k$.
if $k = 1$ **then**
  return R;
else
  Set $k_1 = \lfloor k/2 \rfloor$ and $k_2 = \lceil k/2 \rceil$;
  Let w denote the width of R and h the height; **if** $w \geq h$ **then**
    With a vertical line, divide R into two pieces R_1 and R_2 with area $\frac{k_1}{k} \cdot \text{Area}(R)$ on the right and $\frac{k_2}{k} \cdot \text{Area}(R)$ on the left;
  else
    With a horizontal line, divide R into two pieces R_1 and R_2 with area $\frac{k_1}{k} \cdot \text{Area}(R)$ on the top and $\frac{k_2}{k} \cdot \text{Area}(R)$ on the bottom;
  end
  return $\mathit{RectanglePartition}(R_1, k_1) \cup \mathit{RectanglePartition}(R_2, k_2)$;
end

Algorithm 6.2.1: Algorithm $\text{RectanglePartition}(R, k)$ takes as input an axis-aligned rectangle R and a positive integer k.

The following Lemma is a simplified restatement of a result from [7] which we will make use of in proving the approximation ratio. This has already been proved in [43].

Input: A convex polygon C and an integer k.
Output: The locations of k points p_i in C that approximately minimize $\mathrm{FW}(C, k)$ within a factor of 2.03.
Let $\square C$ denote a minimal axis aligned bounding box of C. Rotate C so that $\square C$ is aligned with the coordinate axes;
Let $R_1, \ldots, R_k = \mathrm{RectanglePartition}(\square C, k)$;
for $i \in \{1, \ldots, k\}$ **do**
  Let c_i denote the center of R_i;
  if $c_i \in C$ **then**
    Set $p_i = c_i$;
  else
    if $R_i \cap C$ *is nonempty* **then**
      Let $R_i^{'}$ be the minimum axis-aligned bounding box of $R_i \cap C$ and let $c_i^{'}$ denote its center;
      Set $p_i = c_i^{'}$;
    else
      Place p_i anywhere in C;
    end
  end
end
return $p_1, \ldots, p_k$;

Algorithm 6.2.2: Algorithm $\mathrm{ApproxFW}(R, k)$ takes as input a convex polygon C and an integer k.

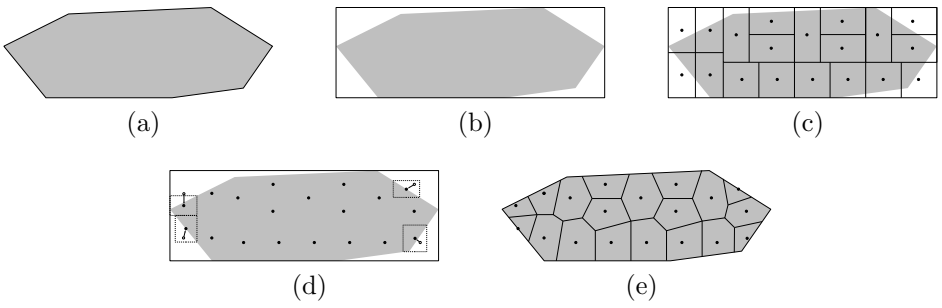


Figure 6.7: The input and output of Algorithm 6.2.2. We begin in (6.7a) with a convex polygon C, whose axis-aligned bounding box $\square C$ is computed in (6.7b). The bounding box is then partitioned into equal-area pieces in (6.7c) using Algorithm 6.2.1. Some of the centers of these pieces are then relocated in (6.7d), and (6.7e) shows the output and Voronoi partition. It turns out that the way we relocate points in (6.7d) does not affect the analysis of the algorithm.

Lemma 6.2.7. *Suppose that $\tilde{R} \subseteq \square C$ is an intermediate rectangle obtained throughout Algorithm 6.2.1, which is further subdivided into $\tilde{R}'$ and $\tilde{R}''$. Then:*

1. *If* $\mathrm{AR}\left(\tilde{R}\right) > 3$, *then*

$$\mathrm{AR}\left(\tilde{R}'\right), \mathrm{AR}\left(\tilde{R}''\right) \leq \mathrm{AR}\left(\tilde{R}\right).$$

2. *If* $\mathrm{AR}\left(\tilde{R}\right) \leq 3$, *then*

$$\mathrm{AR}\left(\tilde{R}'\right), \mathrm{AR}\left(\tilde{R}''\right) \leq 3.$$

Proof. See [43]. □

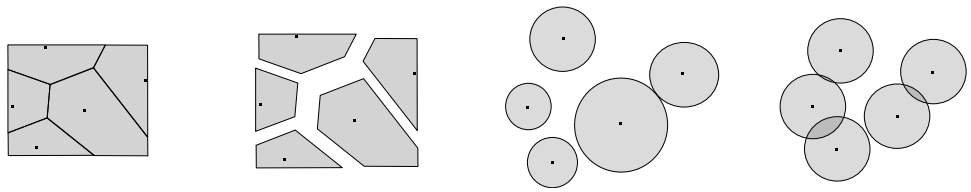
6.2.3 Proof of Constant Factor Approximation

We first establish a lower bound to the continuous k-medians problem in section 6.2.3. To prove that the algorithm is a constant factor approximation, we would need to find the worst case performance of the algorithm. We know that the algorithm breaks the bounding box of C into equal area rectangles of dimensions $w_i \times h_i$, each of which contains a fraction of the area, say A_i, of the convex polygon C. By using the upper bound on $\mathrm{FW}(C)$ we established in section 6.2.1, we can arrive at the worst-case performance of the algorithm by computing the worst value of $\sum_i \phi_{UB}(A_i, w_i, h_i)$ (summed over all the k rectangles). By dividing this quantity by the lower bound on $\mathrm{FW}(C)$, we obtain an approximation ratio, which we prove is bounded above by 2.03. Lets denote the approximation ratio with $\mathcal{A}_{\mathcal{R}}$.

Let C be the given convex polygon with area A, for which B is a bounding box. Without loss of generality, consider B to be of dimensions $w \times 1$, where $w = \mathrm{diam}(C) \geq 1$ as we have assumed throughout. It follows that $\frac{w}{2} \leq A \leq w$. The algorithm breaks the bounding box into k sub-rectangles of equal area $\frac{w}{k}$.

A Simple Lower Bound for $\mathrm{FW}(C, k)$

Let set of points $\{p_1^*, ..., p_k^*\}$ be optimum solution for polygon C, which is itself embedded in a box of dimensions $w \times 1$. If each of k points is assigned area A_i^*



(a) (b) (c) (d)

Figure 6.8: Lower bounding $\mathrm{FW}(C,k)$: Fermat-Weber of each Voronoi cell can be lower bounded by a disk. The sum of Fermat-Weber of disks is least when the areas of all disks are equal, which follows from convexity of Φ_{LB_1}. The same argument holds if we were to replace the disks by slabs (Φ_{LB_2}) shown in Figure 6.1a.

(i.e. if the area of the Voronoi cell associated with the i^{th} point is A_i^*), then the Fermat-Weber value of C must satisfy (refer to Figure 6.8):

$$\mathrm{FW}\left(C,k\right) = \mathrm{FW}\left(A_i^*, p_i^*\right) \geq \sum_{i=1}^{k} \Phi_{LB_1}\left(A_i^*, 1\right)$$

Since each of the summands is *convex* as a function of A_i^*, the right-hand side is *minimized* when each of the points is assigned an equal area $A_i^* = A/k$ (Figure 6.8d). We therefore have: $\mathrm{FW}\left(C,k\right) \geq k \cdot \Phi_{LB_1}\left(A/k, 1\right)$. Replacing Φ_{LB_1} with Φ_{LB_2} gives us a better lower bound and because Φ_{LB_2} is also convex in A, all the arguments still hold.

$$\mathrm{FW}\left(C,k\right) \geq k \cdot \Phi_{LB_2}\left(A/k, 1\right) . \tag{6.7}$$

Simple Upper Bounds for $\mathrm{FW}(C,k)$ under Algorithm 6.2.2

In order to derive an upper bound for $\mathrm{FW}(C,k)$ under Algorithm 6.2.2 we first make a straightforward observation that validates our use of the upper bounding function Φ_{UB} from equation 6.5. Recall that Algorithm 6.2.2 divides the bounding box of C, $\square C$, into k rectangles of equal area, and that C is initially oriented so that its diameter is aligned with the coordinate x-axis. It is easy to see, then, that for any rectangle R_i produced by Algorithm 6.2.1, it must be the case that the region $R_i \cap C$ contains a horizontal line segment whose length is equal to $\mathrm{width}(R_i \cap C)$ (this fact was previously proven in Claim 4.1 of [7]). Thus, we can safely conclude that, if Algorithm 6.2.2 produces k

rectangles with dimensions $w_i \times h_i$, a valid upper bound for $\mathrm{FW}(C,k)$ is indeed $\sum_i \Phi_{UB}(A_i, w_i, h_i)$, where $A_i = \mathrm{Area}(R_i \cap C)$. We can derive a more explicit upper bound for $\mathrm{FW}(C,k)$ by considering the relationship between k and w, as described below.

Case 1: $k < w/3$. Since $k < \frac{w}{3}$, our algorithm divides the bounding box into k rectangles each with area $\frac{w}{k}$ and dimensions $\frac{w}{k} \times 1$. If A_i is the area of C contained within sub-rectangle R_i, then we must have

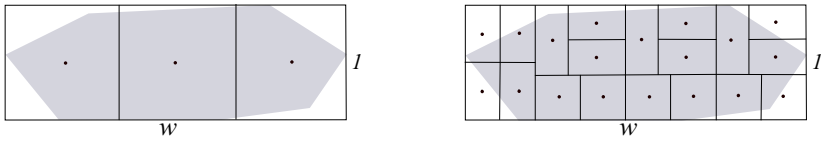
$$\mathrm{FW}(C,k) \leq \sum_{i=1}^{k} \Phi_{UB}\left(A_i, w/k, 1\right) .$$

Since each of the summands is *concave* as a function of A_i, we see that the right-hand side is *maximized* when each of the rectangles is assigned an equal area $A_i = A/k$ of C. We therefore have an overall upper bound given by

$$\mathrm{FW}\left(C,k\right) \leq k \cdot \Phi_{UB}\left(A/k, w/k, 1\right) . \tag{6.8}$$

Using equations (6.7) and (6.8), the worst-case ratio is given by

$$\mathcal{A}_{\mathcal{R}} = \frac{k\Phi_{UB}\left(\frac{A}{k}, \frac{w}{k}, 1\right)}{k\Phi_{LB_2}\left(\frac{A}{k}, 1\right)} . \tag{6.9}$$



(a) $k < w/3$ (b) $k \geq w/3$

Figure 6.9: Skinny and fat cases

Case 2: $k \geq w/3$. Since $k \geq w/3$, using Lemma 6.2.7, we know that Algorithm 6.2.2 divides the bounding box of C into k rectangles each with area $\frac{w}{k}$ and aspect ratio, $\max\{\frac{w_i}{h_i}, \frac{h_i}{w_i}\} \leq 3$. Thus, by our earlier claim, a valid upper bound is

$$FW\left(C\right) \leq \sum_{i=1}^{k} \Phi_{UB}\left(A_i, w_i, h_i\right)$$

It is not hard to verify algebraically that, for any fixed A_i, the function $\Phi_{UB}(A_i, w_i, h_i)$ is maximized when the aspect ratio is as large as possible. Thus, for an upper bound, we can replace w_i with $\sqrt{\frac{3w}{k}}$ and h_i with $\sqrt{\frac{w}{3k}}$. By observing that Φ_{UB} is concave in A_i, we ultimately conclude that:

$$FW(C) \leq k\Phi_{UB}\left(\frac{A}{k}, \sqrt{\frac{3w}{k}}, \sqrt{\frac{w}{3k}}\right) \tag{6.10}$$

Using equations (6.7) and (6.10), the worst case ratio is given by:

$$\mathcal{A}_{\mathcal{R}} = \frac{k\Phi_{UB}\left(\frac{A}{k}, \sqrt{\frac{3w}{k}}, \sqrt{\frac{w}{3k}}\right)}{k\Phi_{LB_2}\left(\frac{A}{k}, 1\right)} \tag{6.11}$$

Combining (6.9) and (6.11) and changing variables: $\alpha = A/k$ and $z = w/k$, we have the following closed-form expression for our approximation ratio $\mathcal{A}_{\mathcal{R}}$:

$$\mathcal{A}_{\mathcal{R}} = \begin{cases} \frac{\Phi_{UB}\left(\alpha, \sqrt{3z}, \sqrt{\frac{z}{3}}\right)}{\Phi_{LB_2}(\alpha, 1)} & \text{if } z \in (0, 3) \\ \frac{\Phi_{UB}(\alpha, z, 1)}{\Phi_{LB_2}(\alpha, 1)} & \text{if } z \geq 3 \end{cases} \tag{6.12}$$

Theorem 6.2.8. *The approximation ratio, $\mathcal{A}_{\mathcal{R}}(\alpha, z)$, is bounded above by 2.03 for all values of $z > 0$ and $\alpha \in \left[\frac{z}{2}, z\right]$.*

Proof. This is straightforward to verify computationally since we have closed-form expressions for the approximation ratio. We break up the proof into 3 cases to prove that $\mathcal{A}_{\mathcal{R}}$ is bounded.

Case 1: $z \in \left(0, \frac{\pi}{4}\right]$ In this case, we will exclusively use the lower bound:

$$\Phi_{LB}(\alpha, 1) = \frac{2}{3\sqrt{\pi}}\alpha^{3/2}$$

from Lemma 6.1.1 It is easy to show that for all rectangles of area z, and aspect ratio 3, $\alpha_c = 0.8844z$, where α_c is defined in the same way as A_c. Hence, we have:

$$\begin{aligned}\Phi_{UB}\left(\alpha, \sqrt{3z}, \sqrt{\frac{z}{3}}\right) &= \min\left\{\frac{\alpha}{0.8844z} FW_{\square}\left(\sqrt{3z}, \sqrt{\frac{z}{3}}\right), FW_{\square}\left(\sqrt{3z}, \sqrt{\frac{z}{3}}\right)\right\} \\ &= \min\left\{\frac{\alpha z^{1/2}}{0.8844} FW_{\square}\left(\sqrt{3}, \sqrt{\frac{1}{3}}\right), z^{3/2} FW_{\square}\left(\sqrt{3}, \sqrt{\frac{1}{3}}\right)\right\} \\ &= \min\left\{0.5374\alpha z^{1/2}, 0.4752 z^{3/2}\right\}\end{aligned}$$

Therefore, the approximation ratio is given by:

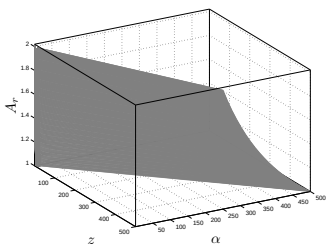
$$\mathcal{A}_{\mathcal{R}} = \frac{\min\left\{0.5374\alpha z^{1/2}, 0.4752 z^{3/2}\right\}}{\frac{2}{3\sqrt{\pi}}\alpha^{3/2}} = \frac{\min\left\{1.4286\alpha z^{1/2}, 1.2635 z^{3/2}\right\}}{\alpha^{3/2}}$$

Clearly, $\mathcal{A}_{\mathcal{R}}$ is highest when $\alpha = \frac{z}{2}$ and hence,

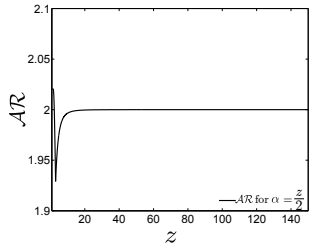
$$\mathcal{A}_{\mathcal{R}} \leq \frac{1.4286\frac{z^{3/2}}{2}}{\left(\frac{z}{2}\right)^{3/2}} = 2.02063$$

as desired.

Case 2: $z \in \left(\frac{\pi}{4}, 500\right]$ The domain $z \in [\frac{\pi}{4}, 500]$, and $\frac{z}{2} \leq \alpha \leq z$ is compact, and we can verify $\mathcal{A}_{\mathcal{R}} \leq 2.03$ computationally using MATLAB, for example (more formally, rather than simply look at a surface plot, we can apply a global branch-and-bound algorithm to maximize $\mathcal{A}_{\mathcal{R}}$ along the domain in question, although we omit this discussion for brevity). Figure 6.10 shows a plot of $\mathcal{A}_{\mathcal{R}}$ as z and α vary, and we can readily observe that $\mathcal{A}_{\mathcal{R}} \leq 2.03$ everywhere throughout.



(a) $\mathcal{A}_{\mathcal{R}}$ is maximum for $\alpha = \frac{z}{2}$ $\forall$ z



(b) $\mathcal{A}_{\mathcal{R}}$

Figure 6.10: The approximation ratio $\mathcal{A}_{\mathcal{R}}$ for a compact region $z \in [\frac{\pi}{4}, 500]$

Case 3: $z > 500$ Here, we will use the concave envelope of $\mathrm{FW}_{1/2}\left(w, \frac{2A}{w}\right)$ as our upper bound. That is:

$$\Phi_{UB} = \frac{A}{wh}\,\mathrm{FW}_{1/2}\,(w, 2h)$$

Since $z > 500$, the rectangle in question is long and skinny; we can then approximate Φ_{UB} using the L_1 norm and Φ_{LB} using the L_∞ norm. Under an L_1 norm, $\mathrm{FW}_{\square}\,(h, w) = 0.25(h^2 w + w^2 h)$. Therefore:

$$\Phi_{UB} = 0.5\frac{\alpha}{z}\,\mathrm{FW}_{\square}\,(z, 2) = 0.25\alpha(z + 2)$$

For the lower bound, it is easy to see that instead of the slab-type shape shown in (Figure 6.1), when we use the L_∞ norm, the shape within our slab of height 1 with minimal Fermat-Weber value is simply a rectangle with dimensions $\alpha \times 1$, whose Fermat-Weber value is at least

$$\Phi_{LB} \geq 0.25\alpha^2 \tag{6.13}$$

The approximation ratio is therefore:

$$\mathcal{A}_{\mathcal{R}} = \frac{\Phi_{UB}}{\Phi_{LB}} \leq \frac{0.25\alpha(z + 2)}{0.25\alpha^2} = \frac{z + 2}{\alpha} < 2.01 \tag{6.14}$$

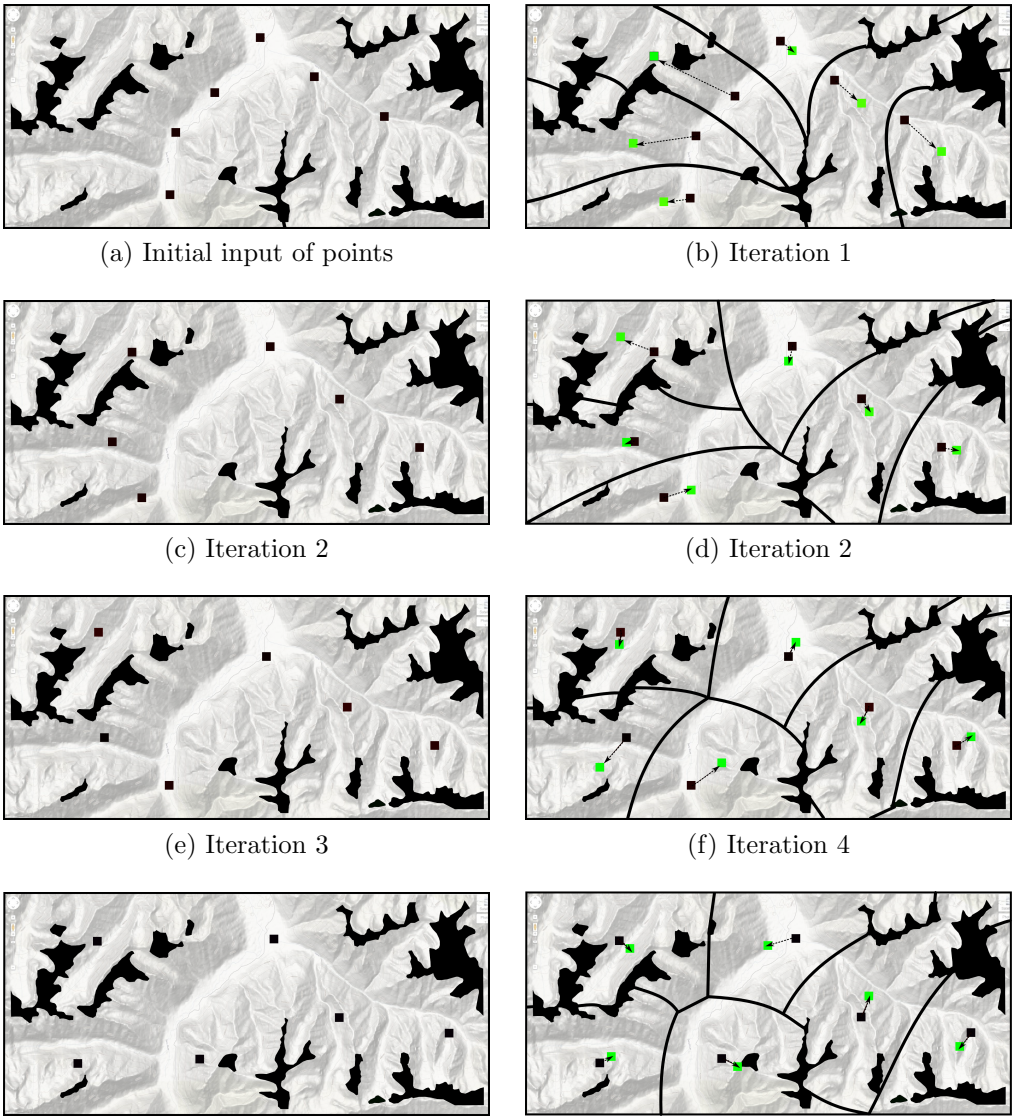
since $z > 500$, which completes the proof. □

6.3 Finding k-Medians in Non-Convex Regions

Finding an algorithm for the k-medians problem in a non-convex region is a really hard problem. The algorithm described above does not extend to a general non convex region either. We present an approach to this problem which uses a natural variation of the well-known *Lloyd algorithm* for computing centroidal Voronoi partitions [80, 122]. The variation is straightforward: for a given initial set of depots P (Figure 4.5a), we compute the optimal partitions using Algorithm 4.4.1 as before. Then, we relocate each depot p_i to the geometric median (the Fermat-Weber center) of its associated region R_i. The new optimal partitions are then re-computed, and so on and so forth. Figure 6.11 provides the first 4 iterations of this algorithm when we use the example of Drass map presented

in Section 4.6. Although this algorithm seems to do well in practice, we are currently unable to provide any kind of performance bound on this algorithm and it remains an unsolved problem.

This algorithm is guaranteed to converge to a solution because the objective function value (the total weighted distance between depots and their assigned sub-regions) decreases at each of the two steps. A detailed proof of convergence can be found in [51]. The problem of relocating each depot p_i to the geometric median of its associated region R_i is called the continuous Fermat-Weber problem and was studied in [55]. For our purposes, we merely estimated the geometric median by discretizing R_i and selecting the best point in the discretization. The locally optimal depot locations and their corresponding partitions for the Drass map is shown in Figure 6.12. The algorithm takes about 18 iterations to converge to the locally optimum. This was also applied to the St. Paul campus map and the result of the algorithm is shown in Figure 6.13.



(a) Initial input of points

(b) Iteration 1

(c) Iteration 2

(d) Iteration 2

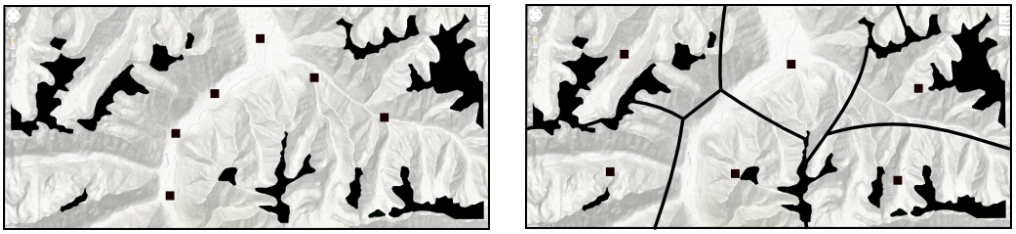
(e) Iteration 3

(f) Iteration 4

(g) Iteration 4

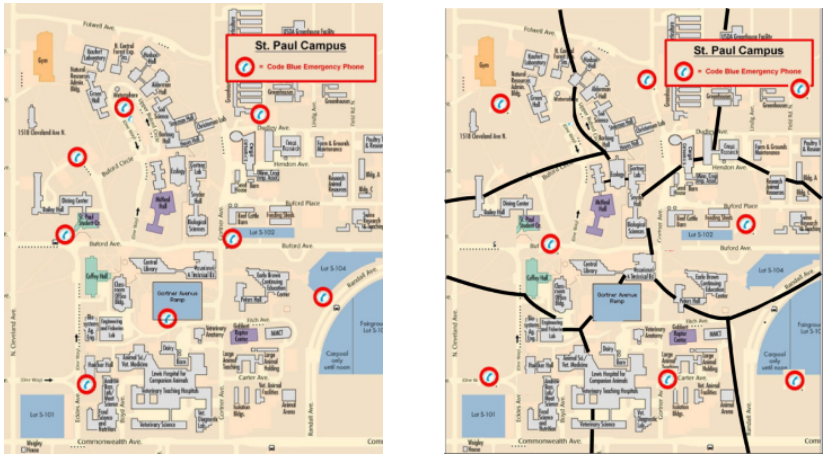
(h) Iteration 4

Figure 6.11: First 4 iterations of modified Lloyd algorithm implemented on Drass map.



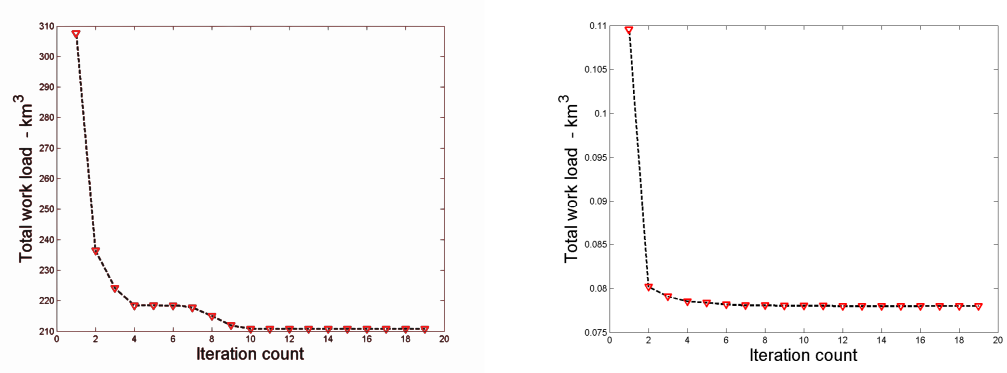
(a) Input (b) Output: Centroidal tesselation

Figure 6.12: Input and output of our simulation when Algorithm 4.4.1 is used as a sub-routine in a Lloyd algorithm.



(a) Input (b) Output: Centroidal tessela-tion

Figure 6.13: Input and output of our simulation when Algorithm 4.4.1 is used as a sub-routine in a Lloyd algorithm.



(a) Drass map. (b) St. Paul campus map.

Figure 6.14: Fermat-Weber value plotted as a function of iterations in Lloyd algorithm.

Chapter 7

Conclusion and Future Work

We have developed a technique based on infinite dimensional optimization framework which allows to us to solve several instances of map segmentation problem. Using an intrinsically interdisciplinary approach, combining elements from variational calculus, computational geometry, geometric probability theory, and vector space optimization, we present an approach where we formulate the problems geometrically and then use a fast geometric algorithm to solve them. Our technique allows us to reduce a class of infinite dimensional optimization problem to a finite dimensional convex problem. We have shown that the boundaries between optimal sub-regions of such problems can be explicitly constructed using complementary slackness; by leveraging this property we are able to solve those map segmentation problems efficiently. This technique has the advantage that we are able to enforce connectivity and impose diameter constraints on the sub-regions. In addition, we also show how to solve such problems dynamically when the underlying demand density varies over time; this might model migration, advection, or addition of new information into an existing model. We demonstrated the efficiency of our algorithms by performing computational experiments for both static and dynamic map segmentation problem.

7.1 Future Work

An exhaustive list of the possible extensions to this work is hard to enlist. However, we present two open problems which we attempted to work towards and were unable to accomplish.

7.1.1 Imposing Convexity

The optimal sub-regions to problems (2.1) and (2.2) under various utility functions are shown in Figure 2.9. As we can see, except when $u_i(x) = -\|x - p_i\|^2$ (power diagram), the optimal sub-regions are non-convex. In practice, convexity of sub-regions is a useful property to have (straight line boundaries are simply easier to deal with). Authors in [7] consider problem (2.1) for $u_i(x) = -\|x - p_i\|$ with the added convexity constraint and provide $(8+\sqrt{2\pi})$ constant factor approximation algorithm that recursively divided the convex region into equal area convex regions. However, it is not trivial how one would impose a convexity constraint for a general framework of map segmentation problem. Providing either exact or approximate solutions for such problems with convexity constraint still remains unclear.

7.1.2 Fermat-Weber Conjecture

The authors in [44] conjecture that the average distance (denoted μ) of points in a convex region from its Fermat-Weber center cannot be greater than one third of its diameter d. This has been followed by a series of papers which attempt to solve this conjecture. Abu-Affash and M.Katz in [1] prove that $\mu \leq 0.3849d$. Their method involves transforming a convex region into circular sectors without decreasing the distance of any point to the center of a bounding circle. Their analysis was first improved to $\mu \leq 0.3489d$ by Dumitrescu et al. in [52] and then to $\mu \leq 0.3444$ in the recent paper [117].

The upper bound on Fermat-Weber of a convex polygon that we developed in Section 6.2.1 is also an attempt towards proving this conjecture. Rather than using a bounding circle of a convex region, our approach was to use its minimum bounding box. The reasoning behind this is that, when the convex region is long and skinny, the area of the minimum bounding circle can be infinitely larger than the area of the convex region itself. But if one considers the minimum bounding box, the ratio of its area to that of the convex polygon is at most 2 for any convex region. Indeed, the upper bound derived in Section 6.2.1 works well for "skinny" convex regions but fails to do so when the convex region is "fat". Improving this bound for fat regions and proving the conjecture still remains an open problem.

Bibliography

[1] A. Karim Abu-Affash and Matthew J. Katz. Improved bounds on the average distance to the fermat–weber center of a convex object. *Inf. Process. Lett.*, 109(6):329–333, February 2009.

[2] R.A. Adams and C. Essex. *Calculus: A Complete Course*. Pearson Education Canada, 2009.

[3] David Adjiashvili and David Peleg. Equal-area locus-based convex polygon decomposition. In *Structural Information and Communication Complexity*, pages 141–155. Springer, 2008.

[4] S. P. Anderson, A. De Palma, and J.-F. Thisse. Demand for differentiated products, discrete choice models, and the characteristics approach. *The Review of Economic Studies*, 56(1):21–35, 1989.

[5] K. Andrews, W. Kienreich, V. Sabol, J. Becker, G. Droschl, F. Kappe, M. Granitzer, P. Auer, and K. Tochtermann. The infosky visual explorer: Exploiting hierarchical structure and document similarities. *Information Visualization*, 1:166–181, 2002.

[6] B. Aronov, P. Carmi, and M. J. Katz. Minimum-cost load-balancing partitions. In *Proceedings of the twenty-second annual symposium on Computational geometry*, SoCG '06, pages 301–308, New York, NY, USA, 2006. ACM.

[7] B. Aronov, P. Carmi, and M.J. Katz. Minimum-cost load-balancing partitions. *Algorithmica*, 54(3):318–336, July 2009.

[8] S. Arora, P. Raghavan, and S. Rao. Approximation schemes for euclidean k-medians and related problems. In *Proceedings of the thirtieth annual*

ACM symposium on Theory of computing, STOC '98, pages 106–113, New York, NY, USA, 1998. ACM.

[9] K. J. Arrow and G. Debreu. Existence of an equilibrium for a competitive economy. *Econometrica: Journal of the Econometric Society*, pages 265–290, 1954.

[10] F. Aurenhammer, F. Hoffmann, and B. Aronov. Minkowski-type theorems and least-squares partitioning. In *Proceedings of the eighth annual symposium on Computational geometry*, SCG '92, pages 350–357, New York, NY, USA, 1992. ACM.

[11] F. Aurenhammer, F. Hoffmann, and B. Aronov. Minkowski-type theorems and least-squares clustering. *Algorithmica*, 20:61–76, 1998. 10.1007/PL00009187.

[12] Franz Aurenhammer. Power diagrams: properties, algorithms and applications. *SIAM Journal on Computing*, 16(1):78–96, 1987.

[13] Franz Aurenhammer. Voronoi diagramsa survey of a fundamental geometric data structure. *ACM Computing Surveys (CSUR)*, 23(3):345–405, 1991.

[14] David Avis, BinayK. Bhattacharya, and Hiroshi Imai. Computing the volume of the union of spheres. *The Visual Computer*, 3(6):323–328, 1988.

[15] Y. Azar. On-line load balancing. In *Online Algorithms*, volume 1442 of *Lecture Notes in Computer Science*, pages 178–195. Springer Berlin / Heidelberg, 1998.

[16] Julius B Barbanel, Steven J Brams, and Walter Stromquist. Cutting a pie is not a piece of cake. *American Mathematical Monthly*, 116(6):496–514, 2009.

[17] O. Baron, O. Berman, D. Krass, and Q. Wang. The equitable location problem on the plane. *European Journal of Operational Research*, 183:578–590, 2007.

[18] Amitabh Basu, Joseph SB Mitchell, and Girish Kumar Sabhnani. Geometric algorithms for optimal airspace design and air traffic controller

workload balancing. *Journal of Experimental Algorithmics (JEA)*, 14:3, 2009.

[19] Jillian Beardwood, John H Halton, and John Michael Hammersley. The shortest path through many points. In *Proc. Cambridge Philos. Soc*, volume 55, pages 299–327. Cambridge Univ Press, 1959.

[20] S. Bereg, P. Bose, and D. Kirkpatrick. Equitable subdivisions within polygonal regions. *Computational Geometry*, 34(1):20–27, 2006. Special Issue on the Japan Conference on Discrete and Computational Geometry 2004.

[21] Sergey Bereg. Orthogonal equipartitions. *Computational Geometry*, 42(4):305–314, 2009.

[22] P.A.G. Bergeijk and S. Brakman. *The Gravity Model in International Trade: Advances and Applications*. Cambridge University Press, 2010.

[23] Oded Berman, Z. Drezner, Arie Tamir, and George O. Wesolowsky. Optimal location with equitable loads. *Annals of Operations Research*, 167(1):307–325, March 2009.

[24] Oded Berman, Zvi Drezner, Dmitry Krass, and George O. Wesolowsky. The variable radius covering problem. *Eur. J. of Oper. Res.*, 196(2):516 – 525, 2009.

[25] Sergei Bespamyatnikh, David Kirkpatrick, and Jack Snoeyink. Generalizing ham sandwich cuts to equitable subdivisions. *Discrete & Computational Geometry*, 24(4):605–622, 2000.

[26] B. Boots and R. South. Modeling retail trade areas using higher-order multiplicatively weighted voronoi diagrams. *Journal of Retailing*, 73:519–536, 1997.

[27] Prosenjit Bose, Erik D Demaine, Ferran Hurtado, John Iacono, Stefan Langerman, and Pat Morin. Geodesic ham-sandwich cuts. *Discrete & Computational Geometry*, 37(3):325–339, 2007.

[28] S. Boyd. Analytic center cutting-plane method. University Lecture, 2010. see `http://www.stanford.edu/class/ee364b/lectures/accpm_slides.pdf`.

[29] S. Boyd. Subgradients. University Lecture, 2010. see `http://www.stanford.edu/class/ee364b/lectures/subgradients_slides.pdf`.

[30] S. Boyd. Subgradients. Lecture slides for EE364b, Stanford University, 2011. See `http://www.stanford.edu/class/ee364b/lectures/subgradients_slides.pdf`.

[31] S. Boyd. Subgradients. Lecture slides for EE364b, Stanford University, 2011. See `http://see.stanford.edu/materials/lsocoee364b/05-localization_methods_slides.pdf`.

[32] Thomas D Boyd and Michael H Jameson. Urban and rural land division in ancient greece. *Hesperia*, pages 327–342, 1981.

[33] Thomas D Boyd and Michael H Jameson. Urban and rural land division in ancient greece. *Hesperia*, pages 327–342, 1981.

[34] S.J. Brams and A.D. Taylor. *Fair division: from Cake-cutting to Dispute Resolution*. Fair Division: From Cake-cutting to Dispute Resolution. Cambridge University Press, 1996.

[35] Steven J Brams and Alan D Taylor. *Fair Division: From cake-cutting to dispute resolution*. Cambridge University Press, 1996.

[36] Jack Brimberg, Pierre Hansen, Nenad Mlandinovic, and Eric. D. Taillard. Improvement and comparison of heuristics for solving the uncapacitated multisource weber problem. *Oper. Res.*, 48:444–460, May 2000.

[37] K. Brner, C. Chen, and K. W. Boyak. Visualizing knowledge domains. *Annual Review of Information Science and Technology*, 37:179–255, 2003.

[38] D. Canter and L. Hammond. A comparison of the efficacy of different decay functions in geographical profiling for a sample of US serial killers. *Journal of Investigative Psychology and Offender Profiling*, 3(2):91–103, 2006.

[39] J.G. Carlsson. Dividing a territory among several vehicles. *INFORMS J. on Comput.*, To appear, 2011.

[40] J.G. Carlsson. Dividing a territory among several vehicles. *INFORMS Journal on Computing*, To appear, 2012.

[41] J.G. Carlsson, B. Armbruster, and Y. Ye. Finding equitable convex partitions of points in a polygon efficiently. *ACM Transactions on Algorithms*, 6(4), 2010.

[42] J.G. Carlsson, D. Ge, A. Subramaniam, and Y. Ye. Solving the min-max multi-depot vehicle routing problem. In *Proceedings of the FIELDS Workshop on Global Optimization*, 2007.

[43] John Gunnar Carlsson, Fan Jia, and Ying Li. An approximation algorithm for the continuous k-medians problem in a convex polygon. *INFORMS Journal on Computing*, 0(0):null, 2013.

[44] Paz Carmi, Sariel Har-Peled, and Matthew J Katz. On the fermat–weber center of a convex object. *Computational Geometry*, 32(3):188–195, 2005.

[45] Moses Charikar, Sudipto Guha, Éva Tardos, and David B. Shmoys. A constant-factor approximation algorithm for the k-median problem. *J. Comput. Syst. Sci.*, 65:129–149, August 2002.

[46] H. Chen, A. L. Houston, R. R. Sewell, and B. R. Schatz. Internet browsing and searching; user evaluations of category map and concept space techniques. *Journal of the American Society for Information Science*, 49:582–608, 1998.

[47] H. Couclelis. Worlds of information: The geographic metaphor in the visualization of complex information. *Cartography and Geographic Information Systems*, 25:209–220, 1998.

[48] AK Dewdney and John K Vranch. A convex partition of r3 with applications to crums problem and knuths post-office problem. *Utilitas Math*, 12:193–199, 1977.

[49] Z. Drezner and A. Suzuki. Covering continuous demand in the plane. *Journal of the Operational Research Society*, 61(5):878–881, 2010.

[50] Zvi Drezner. *Facility Location.* Springer, Berlin, 2001.

[51] Qiang Du, Maria Emelianenko, and Lili Ju. Convergence of the lloyd algorithm for computing centroidal voronoi tessellations. *SIAM journal on numerical analysis*, 44(1):102–119, 2006.

[52] Adrian Dumitrescu and Csaba D. Tóth. New bounds on the average distance from the fermat-weber center of a planar convex body. In *Proceedings of the 20th International Symposium on Algorithms and Computation*, ISAAC '09, pages 132–141, Berlin, Heidelberg, 2009. Springer-Verlag.

[53] J.W. Durham, R. Carli, P. Frasca, and F. Bullo. Discrete partitioning and coverage control for gossiping robots. *IEEE Transactions on Robotics*, 99:1 – 15, 2011.

[54] E. Eisenberg and D. Gale. Consensus of subjective probabilities: the pari-mutuel method. *The Annals of Mathematical Statistics*, 30(1):pp. 165–168, 1959.

[55] S. P. Fekete, J. S. B. Mitchell, and K. Beurer. On the continuous Fermat-Weber problem. *Oper. Res.*, 53:61–76, January 2005.

[56] Sándor P. Fekete, Joseph S. B. Mitchell, and Karin Beurer. On the continuous Fermat-Weber problem. *Oper. Res.*, 53(1):61–76, 2005.

[57] D. Fu-qing, L. Zheng, and W. Li-li. Study on the method of sector division based on the power diagram. In *ICMTMA '09. International Conference on Measuring Technology and Mechatronics Automation, 2009*, volume 2, pages 393 –395, April 2009.

[58] D. Gale. *The Theory of Linear Economic Models.* Economics / mathematics. University of Chicago Press, 1989.

[59] J.-L. Goffin, Z.-Q. Luo, and Y. Ye. Complexity analysis of an interior cutting plane method for convex feasibility problems. *SIAM J. on Optimization*, 6:638–652, March 1996.

[60] Bruce L Golden, Subramanian Raghavan, and Edward A Wasil. *The Vehicle Routing Problem: Latest Advances and New Challenges: latest advances and new challenges*, volume 43. Springer, 2008.

[61] Leonidas J Guibas and Li Zhang. Euclidean proximity and power diagrams. In *CCCG*, 1998.

[62] K.R. Guruprasad and D. Ghose. Coverage optimization using generalized Voronoi partition. `http://arxiv.org/abs/0908.3565v3`, 2011.

[63] G.J. Haltiner and R.T. Williams. *Numerical Prediction and Dynamic Meteorology*. Wiley, 1980.

[64] D. Haugland, S. C. Ho, and G. Laporte. Designing delivery districts for the vehicle routing problem with stochastic demands. *European Journal of Operational Research*, 180(3):997 – 1010, 2007.

[65] Dag Haugland, Sin C Ho, and Gilbert Laporte. Designing delivery districts for the vehicle routing problem with stochastic demands. *European Journal of Operational Research*, 180(3):997–1010, 2007.

[66] Yong He and Zhiyi Tan. Ordinal on-line scheduling for maximizing the minimum machine completion time. *Journal of Combinatorial Optimization*, 6(2):199–206, June 2002.

[67] Theodore P Hill. Determining a fair border. *American Mathematical Monthly*, pages 438–442, 1983.

[68] Hiroshi Imai, Masao Iri, and Kazuo Murota. Voronoi diagram in the laguerre geometry and its applications. *SIAM Journal on Computing*, 14(1):93–105, 1985.

[69] D.R. Ingram, D.R. Clarke, and R.A. Murdie. Distance and the decision to visit an emergency department. *Social Science & Medicine. Part D: Medical Geography*, 12(1):55–62, 1978.

[70] E. Isaacson and H.B. Keller. *Analysis of Numerical Methods*. Dover books on advanced mathematics. Dover Publications, 1994.

[71] Hiro Ito, Hideyuki Uehara, and Mitsuo Yokoyama. 2-dimension ham sandwich theorem for partitioning into three convex pieces. In *Discrete and computational geometry*, pages 129–157. Springer, 2000.

[72] M. Jäger and B. Nebel. Dynamic decentralized area partitioning for cooperating cleaning robots. In *ICRA 2002*, pages 3577–3582, 2002.

[73] S. Kaski, T. Honkela, K. Lagus, and T. Kohonen. Websom: Self-organizing maps of document collections. *Neurocomputing*, 21:101–117, 1998.

[74] H. Kellerer, V. Kotov, M. Grazia Speranza, and Z. Tuza. Semi on-line algorithms for the partition problem. *Operations Research Letters*, 21(5):235 – 242, 1997.

[75] S.G. Krantz and H.R. Parks. *Geometric Integration Theory*. Cornerstones Series. Birkhäuser, 2008.

[76] Hongtao Lei, Gilbert Laporte, and Bo Guo. Districting for routing with stochastic customers. *EURO Journal on Transportation and Logistics*, 1(1-2):67–85, 2012.

[77] R. J. LeVeque. Clawpack. `http://depts.washington.edu/clawpack/`, 2012.

[78] J. Levitt. All about redistricting. `http://redistricting.lls.edu/where-state.php`, 2012. [Online; accessed 07-June-2012].

[79] J.J. Lewer and H. Van den Berg. A gravity model of immigration. *Economics Letters*, 99(1):164–167, April 2008.

[80] S. Lloyd. Least squares quantization in PCM. *Information Theory, IEEE Transactions on*, 28(2):129 – 137, 1982.

[81] E.H. Lockwood. *Book of Curves*. A Book of Curves. Cambridge University Press, 2007.

[82] D. G. Luenberger. *Optimization by Vector Space Methods*. John Wiley & Sons, Inc., 1st edition, 1997.

[83] D.G. Luenberger. *Optimization by vector space methods*. Series in decision and control. Wiley, 1997.

[84] Chryssi Malandraki, David Zaret, Juan R Perez, and Chuck Holland. Industrial engineering applications in transportation. *Handbook of Industrial Engineering: Technology and Operations Management, Third Edition*, pages 787–824, 2001.

[85] Q. Mérigot. A multiscale approach to optimal transport. *Computer Graphics Forum*, 30(5):1583–1592, 2011.

[86] Joseph S. B. Mitchell. Guillotine subdivisions approximate polygonal subdivisions: A simple polynomial-time approximation scheme for geometric TSP, k-MST, and related problems. *SIAM J. Comput.*, 28:1298–1309, March 1999.

[87] J.S.B. Mitchell. Shortest paths among obstacles in the plane. In *Proceedings of the ninth annual symposium on Computational geometry*, SCG '93, pages 308–317, New York, NY, USA, 1993. ACM.

[88] NAG. Fortran library manual. *Numerical Algorithms Group, Ltd., Oxford, UK, Mark*, 17, 1995.

[89] National Research Council (U.S.). Committee on Applied and Theoretical Statistics. Panel on Statistics and Oceanography, Eddy, W.F., National Research Council (U.S.). Board on Mathematical Sciences, and National Research Council (U.S.). Commission on Physical Sciences, Mathematics, and Applications. *Statistics and Physical Oceanography*. National Academy Press, 1993.

[90] J. C. Nekola and P. S. White. Special paper: The distance decay of similarity in biogeography and ecology. *Journal of Biogeography*, 26(4):pp. 867–878, 1999.

[91] Office of Naval Research. Basic research challenge: Decentralized online optimization, 2012. [See: `http://www.onr.navy.mil/~/media/Files/Funding-Announcements/Special-Notice/2012/12-SN-0006.ashx`, Online; accessed 13-Mar-2013].

[92] U.S Government Printing Office. Faa modernization and reform act of 2012, 2012. [See: `http://www.gpo.gov/fdsys/pkg/CRPT-112hrpt381/pdf/CRPT-112hrpt381.pdf`, Online; accessed 13-Mar-2013].

[93] C. Ogilvy. *Excursions in Geometry*. Dover Publications, New York, 1990.

[94] C.S. Ogilvy. *Excursions in Geometry*. Dover Classics of Science and Mathematics. Dover Publications, 1990.

[95] A. Okabe and A. Suzuki. Locational optimization problems solved through Voronoi diagrams. *Eur. J. of Oper. Res.*, 98(3):445 – 456, 1997.

[96] Atsuyuki Okabe, Barry Boots, Kokichi Sugihara, and Sung Nok Chiu. *Spatial tessellations: concepts and applications of Voronoi diagrams*, volume 501. John Wiley & Sons, 2009.

[97] M. O'Leary. Modeling criminal distance decay. *Cityscape: A Journal of Policy Development and Research*, 13:161–198, 2011.

[98] C. H. Papadimitriou. Worst-case and probabilistic analysis of a geometric location problem. *SIAM Journal on Computing*, 10:542, 1981.

[99] M. Pavone, A. Arsie, E. Frazzoli, and F. Bullo. Equitable partitioning policies for robotic networks. In *ICRA'09: Proceedings of the 2009 IEEE international conference on Robotics and Automation*, pages 3979–3984, Piscataway, NJ, USA, 2009. IEEE Press.

[100] M. Pavone, A. Arsie, E. Frazzoli, and F. Bullo. Distributed algorithms for environment partitioning in mobile robotic networks. *Automatic Control, IEEE Transactions on*, 56(8):1834–1848, 2011.

[101] M. Pavone, A. Arsie, E. Frazzoli, and F. Bullo. Distributed algorithms for environment partitioning in mobile robotic networks. *Automatic Control, IEEE Transactions on*, 56(8):1834–1848, 2011.

[102] M. Pavone, N. Bisnik, E. Frazzoli, and V. Isler. Decentralized vehicle routing in a stochastic and dynamic environment with customer impatience. In *RoboComm '07: Proceedings of the 1st international conference on Robot communication and coordination*, pages 1–8. IEEE Press, 2007.

[103] L. Pimenta, V. Kumar, R.C. Mesquita, and G. Pereira. Sensing and coverage for a network of heterogeneous robots. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pages 3947 –3952, December 2008.

[104] L. C. A. Pimenta, V. Kumar, R. C. Mesquita, and G. A. S. Pereira. Sensing and coverage for a network of heterogeneous robots. In *IEEE CDC'08*, pages 3947–3952, 2008.

[105] R. Poulin. The decay of similarity with geographical distance in parasite communities of vertebrate hosts. *Journal of Biogeography*, 30(10):1609–1615, 2003.

[106] Franco P. Preparata and Michael I. Shamos. *Computational geometry: an introduction*. Springer-Verlag New York, Inc., New York, NY, USA, 1985.

[107] A. Quarteroni, R. Sacco, and F. Saleri. *Numerical Mathematics*. Texts in applied mathematics. Springer, 2007.

[108] René Reitsma, Stanislav Trubin, and Eric Mortensen. Weight-proportional space partitioning using adaptive voronoi diagrams. *Geoinformatica*, 11(3):383–405, 2007.

[109] J.P. Rodrigue, C. Comtois, and B. Slack. *The Geography of Transport Systems*. Routledge, 2009.

[110] Toshinori Sakai. Balanced convex partitions of measures in ?2. *Graphs and Combinatorics*, 18(1):169–192, 2002.

[111] E. Sheppard. Theoretical underpinnings of the gravity hypothesis. *Geographical Analysis*, 10(4):386–402, 1978.

[112] E. Sheppard. Geographic potentials. *Annals of the Association of American Geographers*, 69(3):438–447, September 1979.

[113] A. Skupin. A cartographic approach to visualizing conference abstracts. *IEEE Computer Graphics and Application*, 22:50–58, 2002.

[114] R. Stock. Distance and the utilization of health facilities in rural Nigeria. *Social Science & Medicine*, 17(9):563 – 570, 1983.

[115] A. Suzuki and Z. Drezner. The p-center location problem in an area. *Location Sci.*, 4(1-2):69 – 82, 1996.

[116] A. Suzuki and Z. Drezner. The minimum equitable radius location problem with continuous demand. *Eur. J. of Oper. Res.*, 195(1):17 – 30, 2009.

[117] Xuehou Tan and Bo Jiang. A new approach to the upper bound on the average distance from the fermat-weber center of a convex body. In *Combinatorial Optimization and Applications*, pages 250–259. Springer, 2013.

[118] W. Tobler. Spatial interaction patterns. *Journal of Environmental Systems*, 6:271–301, 1976.

[119] W. Tobler. Thirty five years of computer cartograms. *Annals of the Association of American Geographers*, 94:58–73, 2004.

[120] R.S. Varga. *Matrix Iterative Analysis*. Springer Series in Computational Mathematics. Springer, 2009.

[121] C. Villani. *Topics in optimal transportation*. Graduate studies in mathematics. American Mathematical Society, 2003.

[122] Wikipedia. Lloyd's algorithm, 2012. [Online; accessed 05-Feb-2012].

[123] KF Wong and John E Beasley. Vehicle routing using fixed delivery areas. *Omega*, 12(6):591–600, 1984.

[124] Hongsheng Zhong, Randolph W Hall, and Maged Dessouky. Territory planning and vehicle dispatching with driver learning. *Transportation Science*, 41(1):74–89, 2007.

Appendices

Appendix A

Proofs of Theorem 2.3.1 and 2.3.3

In order to prove Theorems 2.3.1 and 2.3.3, we find it helpful to first state an important result from Section 8.6 of [83]:

Theorem A.0.1. *(Lagrange Duality) Let* $\mathfrak{f}$ *be a real-valued convex functional defined on a convex subset* Ω *of a vector space* $\mathfrak{X}$*, and let* $\mathfrak{G}$ *be a convex mapping of* $\mathfrak{X}$ *into a normed space* $\mathfrak{Z}$*. Suppose there exists* $\mathfrak{x}_1 \in \mathfrak{X}$ *such that* $\mathfrak{G}(\mathfrak{x}_1) < \theta$*, where* θ *denotes the zero element, and that* $\mu_0 := \inf\{\mathfrak{f}(\mathfrak{x}) : \mathfrak{G}(\mathfrak{x}) \leq \theta, \mathfrak{x} \in \Omega\}$ *is finite. Then*

$$\inf_{\mathfrak{x}\in\Omega,\, \mathfrak{G}(\mathfrak{x})\leq\theta} \mathfrak{f}(\mathfrak{x}) = \max_{\mathfrak{z}^*\geq\theta} \varphi(\mathfrak{z}^\star)$$

where

$$\varphi(\mathfrak{z}^\star) = \inf_{\mathfrak{x}\in\Omega} \mathfrak{f}(\mathfrak{x}) + \langle \mathfrak{G}(\mathfrak{x}), \mathfrak{z}^\star \rangle \ ,$$

and the maximum on the right is achieved by some $\mathfrak{z}_0^\star \geq \theta$*.*

We will assume in this section, without loss of generality, that $u_i(x) > 0$ for all i and all $x \in R$.

A.1 Proof of Theorem 2.3.1

We find it helpful to begin our proof by first considering the *dual* problem (2.7), which we will prove is equivalent to the original problem (2.1). The reason that we prefer to do things in this order is because it is easier to verify that a bounded optimal solution to (2.7) actually exists, as demonstrated by the following:

Lemma A.1.1. *A bounded optimal solution $\boldsymbol{\lambda}^*$ to problem (2.7) exists.*

Proof. Let $S = \iint_R f(x) \max_i u_i(x)\, dA$ denote the objective function value of (2.7) at $\boldsymbol{\lambda} = \mathbf{0}$ and let $Q = -2 \max_i \iint_R f(x)|u_i(x)|\, dA$, which implies that $Q \leq S$. Note that for any indices j and k there exists a finite threshold m_{jk} such that, if $\lambda_j - \lambda_k \geq m_{jk}$, then $\iint_{R_j} f(x)\, dA \leq \epsilon$, where $\epsilon = \frac{1}{2n}$. Let M' be the maximum of all such thresholds m_{jk}, let $M'' = 4(S - Q)$, and let $M = \max\{M', M''\}$.

Suppose that $\boldsymbol{\lambda}$ satisfies $\mathbf{q}^T\boldsymbol{\lambda} = 0$ and $\|\boldsymbol{\lambda}\|_\infty > (n-1)M$. Let $\lambda_j = \max_i \lambda_i > 0$ and $\lambda_k = \min_i \lambda_i < 0$; by definition, we must have $\lambda_j \leq (n-1)|\lambda_k|$ and $|\lambda_k| > M$. Let $R_j := \{x \in R : u_j(x) - \lambda_j \geq u_k(x) - \lambda_k\}$ and let $R_k = R \setminus R_j$. (Note that this is different from our usual definition of the regions R_i because we are disregarding all indices other than j and k.) The objective function value of (2.7) is then

$$\iint_R f(x) \max_i \{u_i(x) - \lambda_i\}\, dA$$

$$\begin{aligned}
&\geq \iint_R f(x) \max\{u_j(x) - \lambda_j, u_k(x) - \lambda_k\}\, dA \\
&= \iint_{R_j} f(x)(u_j(x) - \lambda_j)\, dA + \iint_{R_k} f(x)(u_k(x) - \lambda_k)\, dA \\
&= \iint_{R_j} f(x)u_j(x)\, dA - \lambda_j \underbrace{\iint_{R_j} f(x)\, dA}_{\leq \frac{1}{2n}} + \iint_{R_k} f(x)u_k(x)\, dA - \lambda_k \underbrace{\iint_{R_k} f(x)\, dA}_{\geq 1 - \frac{1}{2n}} \\
&\geq \underbrace{\iint_{R_j} f(x)u_j(x)\, dA}_{\geq -\iint_R f(x)|u_j(x)|\, dA} + \underbrace{\iint_{R_k} f(x)u_k(x)\, dA}_{\geq -\iint_R f(x)|u_k(x)|\, dA} - \frac{\lambda_j}{2n} + |\lambda_k| \left(1 - \frac{1}{2n}\right) \\
&\geq Q - \frac{\lambda_j}{2n} + |\lambda_k| \left(1 - \frac{1}{2n}\right) \geq Q - \frac{n-1}{2n}|\lambda_k| + |\lambda_k| \left(1 - \frac{1}{2n}\right) \\
&= Q + |\lambda_k|/2 \geq S
\end{aligned}$$

and therefore $\boldsymbol{\lambda}$ has an objective value no better than that induced by the zero vector. Thus, we can assume without loss of generality that problem (2.7) is restricted to the compact set $\|\boldsymbol{\lambda}\|_\infty \leq (n-1)M$, which completes the proof. □

To prove Theorem 2.3.1, we find it helpful to use the alternate formulation

(2.6) of (2.7), reproduced below:

$$\begin{aligned} \underset{\boldsymbol{\lambda},\sigma(\cdot)}{\text{minimize}} \sum_{i=1}^{n} q_i\lambda_i + \iint_R f(x)\sigma(x)\,dA \qquad & s.t. \\ \sigma(x) \;&\geq\; u_i(x) - \lambda_i \quad \forall i, x\,. \end{aligned} \tag{A.1}$$

We now apply Theorem A.0.1: in problem (A.1), the optimization variables are $\boldsymbol{\lambda}$ and $\sigma(\cdot)$, so we let $\mathfrak{X} = \Omega = \mathbb{R}^n \oplus L_1$, where L_1 represents all functions $h(\cdot)$ defined on R such that $|h(x)|$ is Lebesgue integrable on R. We let $\mathfrak{f}(\mathfrak{x})$ be defined by

$$\mathfrak{f} : \begin{pmatrix} \boldsymbol{\lambda} \\ \sigma(\cdot) \end{pmatrix} \mapsto \mathbf{q}^T\boldsymbol{\lambda} + \iint_R f(x)\sigma(x)\,dA$$

and we let $\mathfrak{G} : \mathfrak{X} \to \mathfrak{Z}$ be defined by

$$\mathfrak{G} : \begin{pmatrix} \boldsymbol{\lambda} \\ \sigma(\cdot) \end{pmatrix} \mapsto \begin{pmatrix} \xi_1(\cdot) - \lambda_1 \\ \vdots \\ \xi_n(\cdot) - \lambda_n \end{pmatrix}$$

where $\xi_i(x) := u_i(x) - \sigma(x)$, so that $\mathfrak{Z} = \underbrace{L_1 \oplus \cdots \oplus L_1}_{n}$. By the preceding existence argument for $\boldsymbol{\lambda}^*$, we can replace the infimum operator of Theorem A.0.1 with the minimum operator. From basic functional analysis, we have $\mathfrak{Z}^\star = \underbrace{L_\infty \oplus \cdots \oplus L_\infty}_{n}$, where L_∞ denotes all *bounded* functions on R. Let $(J_1(\cdot), \ldots, J_n(\cdot))$ denote an element of $\mathfrak{Z}^\star$. Theorem A.0.1 says that

$$\begin{aligned} &\min_{\mathfrak{x}\in\Omega,\, \mathfrak{G}(\mathfrak{x})\leq\theta} \mathfrak{f}(\mathfrak{x}) \\ &= \max_{\mathfrak{z}^*\geq\theta} \left\{ \inf_{\mathfrak{x}\in\Omega} \mathfrak{f}(\mathfrak{x}) + \langle \mathfrak{G}(\mathfrak{x}), \mathfrak{z}^\star \rangle \right\} \\ &= \max_{J_i(\cdot)\geq 0} \left\{ \inf_{\boldsymbol{\lambda},\sigma(\cdot)} \mathbf{q}^T\boldsymbol{\lambda} + \iint_R f(x)\sigma(x)\; dA + \iint_R \sum_{i=1}^{n} J_i(x)\left(\xi_i(x) - \lambda_i\right) dA \right\} \end{aligned}$$

$$= \max_{J_i(\cdot)\geq 0} \left\{ \inf_{\boldsymbol{\lambda},\sigma(\cdot)} \left[\sum_{i=1}^{n} \left(q_i - \iint_R J_i(x)\, dA \right) \lambda_i \right] ... \right.$$
$$\left. + \iint_R \left(\sum_{i=1}^{n} J_i(x)\, u_i(x) - J_i(x)\, \sigma(x) \right) + f(x)\sigma(x)\, dA \right\}$$

$$= \max_{J_i(\cdot)\geq 0} \left\{ \inf_{\boldsymbol{\lambda},\sigma(\cdot)} \left[\sum_{i=1}^{n} \left(q_i - \iint_R J_i(x)\, dA \right) \lambda_i \right] ... \right.$$
$$\left. + \iint_R \sigma(x) \left(f(x) - \sum_{i=1}^{n} J_i(x) \right) dA + \iint_R \sum_{i=1}^{n} J_i(x)\, u_i(x)\, dA \right\}$$

Clearly we need $\iint_R J_i(x)\, dA = q_i$ for all i and $\sum_i J_i(x) = f(x)$ for all $x \in R$ (the infimum term over all $\boldsymbol{\lambda}$ and $\sigma(\cdot)$ is unbounded below otherwise). Introducing new functions $I_i(x) := J_i(x)/f(x)$, the above is equivalent to

$$\max_{I_i(\cdot)\geq 0} \left\{ \inf_{\boldsymbol{\lambda},\sigma(\cdot)} \left[\sum_{i=1}^{n} \left(q_i - \iint_R f(x) I_i(x)\, dA \right) \lambda_i \right] ... \right.$$
$$\left. + \iint_R f(x)\sigma(x) \left(1 - \sum_{i=1}^{n} I_i(x) \right) dA + \iint_R \sum_{i=1}^{n} f(x) u_i(x) I_i(x)\, dA \right\}$$

so that $\iint_R f(x) I_i(x)\, dA = q_i$ for all i and $\sum_i I_i(x) = 1$ for all $x \in R$. Thus, the problem (2.7) and the problem

$$\begin{aligned} \underset{I_1(\cdot),\ldots,I_n(\cdot)}{\text{maximize}} \iint_R \sum_{i=1}^{n} f(x) u_i(x) I_i(x)\, dA \qquad & s.t. \\ \iint_R f(x) I_i(x)\, dA &= q_i \quad \forall i \\ \sum_{i=1}^{n} I_i(x) &= 1 \quad \forall x \\ I_i(x) &\geq 0 \quad \forall i, x\,. \end{aligned}$$

are primal-dual pairs as desired, and by Theorem A.0.1 we know that an optimal solution $I_1^*(\cdot), \ldots, I_n^*(\cdot)$ to problem (2.1) exists. This completes the proof.

A.2 Proof of Theorem 2.3.3

To prove Theorem 2.3.3, we will again consider the dual problem (2.10) first and show that this is equivalent to problem (2.2). Using a very similar argument to that of Lemma A.1.1, which we omit for brevity, it is not hard to verify that there must exist an optimal solution $\boldsymbol{\lambda}^*$ to problem (2.10). In order to apply Theorem A.0.1 directly, we will alter problem (2.10) without loss of generality by substituting an inequality in the linear constraint on $\boldsymbol{\lambda}$ and by writing the problem as a linear program:

$$\begin{aligned} \underset{\boldsymbol{\lambda},\sigma(\cdot)}{\text{minimize}} \iint_R f(x)\sigma(x)\,dA \qquad & s.t. \\ \sigma(x) &\geq \lambda_i u_i(x) \quad \forall i, x \\ \sum_{i=1}^{n} \lambda_i &\geq 1 \\ \lambda_i &\geq 0 \quad \forall i \end{aligned} \tag{A.2}$$

In problem (A.2), the optimization variables are $\boldsymbol{\lambda}$ and $\sigma(\cdot)$, so we let $\mathfrak{X} = \mathbb{R}^n \oplus L_1$, where L_1 represents all functions $h(\cdot)$ defined on R such that $|h(x)|$ is Lebesgue integrable on R. Let Ω denote the positive orthant, i.e. $\lambda_i \geq 0$ and $\sigma(x) \geq 0$ for all $x \in R$. We let $\mathfrak{f}(\mathfrak{x})$ be defined by

$$\mathfrak{f} : \begin{pmatrix} \boldsymbol{\lambda} \\ \sigma(\cdot) \end{pmatrix} \mapsto \iint_R f(x)\sigma(x)\,dA$$

and we let $\mathfrak{G} : \mathfrak{X} \to \mathfrak{Z}$ be defined by

$$\mathfrak{G} : \begin{pmatrix} \boldsymbol{\lambda} \\ \sigma(\cdot) \end{pmatrix} \mapsto \begin{pmatrix} \xi_1(\cdot) \\ \vdots \\ \xi_n(\cdot) \\ 1 - \sum_i \lambda_i \end{pmatrix}$$

where $\xi_i(x) := \lambda_i u_i(x) - \sigma(x)$, so that $\mathfrak{Z} = \underbrace{L_1 \oplus \cdots \oplus L_1}_{n} \oplus \mathbb{R}$. We can again replace the infimum operator in Theorem A.0.1 with the minimum operator. Let $(J_1, \ldots, J_n, t) \in \underbrace{L_\infty \oplus \cdots \oplus L_\infty}_{n} \oplus \mathbb{R}$ denote an element of the dual space $\mathfrak{Z}^\star$,

and as before define $I_i(x) := J_i(x)/f(x)$. We then find that

$$\min_{\mathfrak{x}\in\Omega,\,\mathfrak{G}(\mathfrak{x})\leq\theta} \mathfrak{f}(\mathfrak{x}) = \max_{\mathfrak{z}^*\geq\theta} \left\{ \inf_{\mathfrak{x}\in\Omega} \mathfrak{f}(\mathfrak{x}) + \langle \mathfrak{G}(\mathfrak{x}), \mathfrak{z}^\star \rangle \right\}$$

$$= \max_{J_i(\cdot),t\geq 0} \left\{ \inf_{\boldsymbol{\lambda}\geq\mathbf{0},\sigma(\cdot)\geq 0} \iint_R f(x)\sigma(x)\, dA + \iint_R \sum_{i=1}^n J_i(x)\,\xi_i(x)\, dA \ldots \right.$$
$$\left. + t\left(1 - \sum_{i=1}^n \lambda_i\right) \right\}$$

$$= \max_{J_i(\cdot),t\geq 0} \left\{ \inf_{\boldsymbol{\lambda}\geq\mathbf{0},\sigma(\cdot)\geq 0} \iint_R f(x)\sigma(x) + \sum_{i=1}^n J_i(x)\,\xi_i(x)\, dA \ldots \right.$$
$$\left. + \iint_R f(x)t\left(1 - \sum_{i=1}^n \lambda_i\right) dA \right\}$$

$$= \max_{J_i(\cdot),t\geq 0} \left\{ \inf_{\boldsymbol{\lambda}\geq\mathbf{0},\sigma(\cdot)\geq 0} \iint_R \left(\sum_{i=1}^n J_i(x)\,\xi_i(x) - f(x)t\lambda_i\right) dA \ldots \right.$$
$$\left. + f(x)\sigma(x) + f(x)t\, dA \right\}$$

$$= \max_{J_i(\cdot),t\geq 0} \left\{ \inf_{\boldsymbol{\lambda}\geq\mathbf{0},\sigma(\cdot)\geq 0} \iint_R \sigma(x)\left(f(x) - \sum_{i=1}^n J_i(x)\right) dA \ldots \right.$$
$$\left. + \left[\sum_{i=1}^n \lambda_i \left(J_i(x)\,u_i(x) - f(x)t\right)\right] + f(x)t\, dA \right\}$$

$$= \max_{I_i(\cdot),t\geq 0} \left\{ \inf_{\boldsymbol{\lambda}\geq\mathbf{0},\sigma(\cdot)\geq 0} \iint_R \sigma(x)\, f(x)\left(1 - \sum_{i=1}^n I_i(x)\right) dA \ldots \right.$$
$$\left. + \left[\sum_{i=1}^n \lambda_i \left(\iint_R f(x)u_i(x)I_i(x)\, dA - t\right)\right] + t \right\}$$

which implies that $\sum_i I_i(x) \leq 1$ for all $x \in R$ and $\iint_R f(x) I_i(x)\, u_i(x)\, dA \geq t$ for all i. Thus, problem (A.2) and the problem

$$\begin{aligned} \underset{t, I_1(\cdot), \ldots, I_n(\cdot)}{\text{maximize}}\ & t \qquad s.t. \qquad (A.3) \\ t \ &\leq \ \iint_R f(x) u_i(x) I_i(x)\, dA \quad \forall i \\ \sum_{i=1}^{n} I_i(x) \ &\leq \ 1 \quad \forall x \\ I_i(x) \ &\geq \ 0 \quad \forall i, x \end{aligned}$$

are primal-dual pairs. We see that problem (A.3) differs from the linear relaxation of problem (2.8) only by the inequality $\sum_{i=1}^{n} I_i(x) \leq 1$ for all x; it is of course trivial to see that equality must hold at optimality for problem (A.3) which confirms that the two problems are equivalent. This completes the proof.

Appendix B

Ambiguities Arising Due to Duality

Section 2.4, and Algorithms 2.5.1 and 2.5.2, show how to solve problems (2.1) and (2.2) by way of complementary slackness: given the optimal Lagrange multiplier $\boldsymbol{\lambda}^*$ to either problem, we define each optimal region R_i^* to be those points $x \in R$ such that either $u_i(x) - \lambda_i^*$ or $\lambda_i^* u_i(x)$ is maximal over all i (depending on what problem we want to solve). For most of the examples used in this thesis, this characterization is sufficient, because the set of "ambiguous points" x where this maximal index is not unique has measure zero. In this section, we show how to define the optimal partition when there exists a set with positive measure on which the maximal index i is not unique. As we noted in Section 4.3, one case where this arises is when R contains a set of obstacles and we set $u_i(x) = -d(x, p_i)$, where $d(x, p_i)$ is the length of the shortest path between x and p_i.

B.1 Ambiguities in (2.1)

Let $\boldsymbol{\lambda}^*$ denote an optimal Lagrange multiplier for problem (2.7), the dual of (2.4). Let $R_1^+, \ldots, R_n^+$ denote the *strict dominance regions* where $u_i(x) - \lambda_i^*$ is strictly maximal for some i, and let $R_1^-, \ldots, R_k^-$ denote the *ambiguous dominance regions* where strict optimality does not hold. Associated with each ambiguous dominance region R_j^- is an index set $\mathcal{I}_j \subseteq \{1, \ldots, n\}$ that indicates the set of indices i for which $u_i(x) - \lambda_i^*$ is maximal.

Recall that Theorem A.0.1 guarantees that an optimal solution to the linear relaxation of (2.1), i.e. (2.4), must exist. Let $I_1^*(\cdot), \ldots, I_n^*(\cdot)$ denote this optimal solution, which would of course be unknown to us. By Theorem A.0.1, we are guaranteed that strong duality holds, i.e. that

$$\begin{aligned} \texttt{OPT}: &= \iint_R f(x) \max_i \{u_i(x) - \lambda_i^*\}\, dA = \sum_{i=1}^n \iint_R f(x) u_i(x) I_i^*(x)\, dA \\ &= \left(\sum_{i=1}^n \iint_{R_i^+} f(x) u_i(x)\, dA \right) + \left(\sum_{j=1}^k \iint_{R_j^-} f(x) \sum_{i \in \mathcal{I}_j} u_i(x) I_i^*(x)\, dA \right). \end{aligned}$$

Consider a particular ambiguous dominance region R_j^-; by construction, we are guaranteed that $u_i(x) - \lambda_i^* = u_{i'}(x) - \lambda_{i'}^*$ for all $i, i' \in \mathcal{I}_j$ and all $x \in R_j^-$. Let $\bar{u}(x) = u_i(x) - \lambda_i^*$ for any (equivalently, all) $i \in \mathcal{I}_j$, so that

$$\begin{aligned} \iint_{R_j^-} f(x) \sum_{i \in \mathcal{I}_j} u_i(x) I_i^*(x)\, dA &= \iint_{R_j^-} f(x) \sum_{i \in \mathcal{I}_j} (\bar{u}(x) + \lambda_i^*) I_i^*(x)\, dA \\ &= \iint_{R_j^-} f(x) \bar{u}(x) \underbrace{\sum_{i \in \mathcal{I}_j} I_i^*(x)}_{=1}\, dA + \iint_{R_j^-} f(x) \sum_{i \in \mathcal{I}_j} \lambda_i^* I_i^*(x)\, dA \\ &= \underbrace{\iint_{R_j^-} f(x) \bar{u}(x)\, dA}_{=:a_j \text{ (known)}} + \sum_{i \in \mathcal{I}_j} \lambda_i^* \iint_{R_j^-} f(x) I_i^*(x)\, dA\,. \end{aligned}$$

Observe that the second term in the above, $\sum_i \lambda_i^* \iint_{R_j^-} f(x) I_i^*(x)\, dA$, does not actually depend on the *functions* $I_i^*(\cdot)$, but merely the amount of the mass in R_j^- that is allocated to i. In other words, setting $b_j = \iint_{R_j^-} f(x)\, dA$, we see that

$$\sum_{i \in \mathcal{I}_j} \lambda_i^* \iint_{R_j^-} f(x) I_i^*(x)\, dA = \sum_{i \in \mathcal{I}_j} \lambda_i^* \alpha_{ij} b_j$$

for some coefficients $\alpha_{ij} \geq 0$ such that $\sum_{i \in \mathcal{I}_j} \alpha_{ij} = 1$ for all j. Given an optimal Lagrange multiplier $\boldsymbol{\lambda}^*$, and the dual optimal objective value $\texttt{OPT}$, it is therefore

easy to find the optimal coefficients α_{ij} because we can write

$$\begin{aligned}
\texttt{OPT} &= \iint_R f(x) \max_i \{u_i(x) - \lambda_i^*\}\, dA = \sum_{i=1}^{n} \iint_R f(x) u_i(x) I_i^*(x)\, dA \\
&= \underbrace{\left(\sum_{i=1}^{n} \iint_{R_i^+} f(x) u_i(x)\, dA \right)}_{=:c \text{ (known)}} + \left(\sum_{j=1}^{k} \iint_{R_j^-} f(x) \sum_{i \in \mathcal{I}_j} u_i(x) I_i^*(x)\, dA \right) \\
&= c + \sum_{j=1}^{k} \left(a_j + \sum_{i \in \mathcal{I}_j} \lambda_i^* \alpha_{ij} b_j \right)
\end{aligned}$$

and solve for the terms α_{ij} using linear programming. It is then a trivial matter to divide each ambiguous region R_j^- into components with mass $\alpha_{ij} b_j$ in whatever manner we like.

B.2 Ambiguities in (2.2)

Let $\boldsymbol{\lambda}^*$ denote an optimal Lagrange multiplier for problem (2.10), the dual of the linear relaxation of (2.8). Define regions $R_1^+, \ldots, R_n^+$, $R_1^-, \ldots, R_k^-$, and $\mathcal{I}_1, \ldots, \mathcal{I}_k$ analogously as in Section B.1. Theorem A.0.1 again guarantees that an optimal solution to the linear relaxation of (2.8) must exist, so that

$$\begin{aligned}
\texttt{OPT}: &= \iint_R f(x) \max_i \{\lambda_i^* u_i(x)\}\, dA = t^* = \iint_R f(x) u_i(x) I_i^*(x)\, dA \quad \forall i \\
&= \iint_{R_i^+} f(x) u_i(x)\, dA + \sum_{j : i \in \mathcal{I}_j} \iint_{R_j^-} f(x) u_i(x) I_i^*(x)\, dA \quad \forall i
\end{aligned}$$

where we have used the fact that, at optimality, it must be true that $t^* = \iint_R f(x) u_i(x) I_i^*(x)\, dA$ for all i (this occurs because we have assumed that $f(x) > 0$ and $u_i(x) > 0$ for all i and $x \in R$, and thus $\lambda_i^* > 0$ for all i as well). Consider a particular ambiguous dominance region R_j^-; by construction, we are guaranteed that $\lambda_i^* u_i(x) = \lambda_{i'}^* u_{i'}(x)$ for all $i, i' \in \mathcal{I}_j$ and all $x \in R_j^-$. Let $\bar{u}(x) = \lambda_i^* u_i(x)$ for any (equivalently, all) $i \in \mathcal{I}_j$, so that

$$\iint_{R_j^-} f(x) u_i(x) I_i^*(x)\, dA = \frac{1}{\lambda_i^*} \iint_{R_j^-} f(x) \bar{u}(x) I_i^*(x)\, dA\,.$$

We again observe that the term $\iint_{R_j^-} f(x)\bar{u}(x)I_i^*(x)\,dA$ does not actually depend on the *function* $I_i^*(\cdot)$ but merely the amount of the mass in R_j^- that is allocated to i. In other words, setting $b_j = \iint_{R_j^-} f(x)\bar{u}(x)\,dA$, we see that

$$\frac{1}{\lambda_i^*}\iint_{R_j^-} f(x)\bar{u}(x)I_i^*(x)\,dA = \frac{\alpha_{ij}b_j}{\lambda_i^*}$$

for some coefficients $\alpha_{ij} \geq 0$ such that $\sum_{i\in\mathcal{I}_j}\alpha_{ij} = 1$ for all j. Thus, given an optimal Lagrange multiplier $\boldsymbol{\lambda}^*$, and the dual optimal objective value $\texttt{OPT}$, it is therefore easy to find the optimal coefficients α_{ij} because we can write

$$\begin{aligned}\texttt{OPT} &= \underbrace{\iint_{R_i^+} f(x)u_i(x)\,dA}_{c_i \text{ (known)}} + \sum_{j:i\in\mathcal{I}_j}\iint_{R_j^-} f(x)u_i(x)I_i^*(x)\,dA \quad \forall i \\ &= c_i + \frac{1}{\lambda_i^*}\sum_{j:i\in\mathcal{I}_j}\alpha_{ij}b_j \quad \forall i\end{aligned}$$

and solve for the terms α_{ij} using linear programming. It is then a trivial matter to divide each ambiguous region R_j^- into components with mass $\alpha_{ij}b_j$ in whatever manner we like.