1
Smoothed Analysis of the Simplex Method

Daniel Dadush and Sophie Huiberts

Abstract

In this chapter, we give a technical overview of smoothed analyses of the shadow vertex simplex method for linear programming (LP). We first review the properties of the shadow vertex simplex method and its associated geometry. We begin the smoothed analysis discussion with an analysis of the successive shortest path algorithm for the minimum-cost maximum-flow problem under objective perturbations, a classical instantiation of the shadow vertex simplex method. Then we move to general linear programming and give an analysis of a shadow vertex based algorithm for linear programming under Gaussian constraint perturbations.

1.1 Introduction

We recall that a linear program (LP) in n variables and m constraints is of the form:

$$\begin{aligned} \max\ & c^\mathsf{T} x \\ & Ax \le b, \end{aligned} \tag{1.1}$$

where $x \in \mathbb{R}^n$ are the decision variables. The data of the LP are the objective $c \in \mathbb{R}^n$, the constraint matrix $A \in \mathbb{R}^{m \times n}$ and the corresponding right-hand side vector $b \in \mathbb{R}^m$. We shall refer to $P = \{x \in \mathbb{R}^n : Ax \le b\}$ as the feasible polyhedron. Throughout the chapter, we will assume that the reader is familiar with the basics of linear programming and polyhedral theory (the reader may consult the excellent book by Matousek and Gärtner (2007) for a reference).

The simplex method, introduced by Dantzig in 1947, is the first procedure developed for algorithmically solving LPs. It is a class of local search based LP algorithms, which solve LPs by moving from vertex to vertex along edges of the feasible polyhedron until an optimal solution or unbounded ray is found. The methods differ by the rule they use for choosing the next vertex to move to, known as the pivot rule. Three popular pivot rules are Dantzig's rule, which chooses the edge for which the objective gain per unit of slack is maximized (with respect to the current tight constraints), and Goldfarb's steepest edge rule together with its approximate cousin, Harris' Devex rule, which chooses the edge whose angle to the objective is minimized.

Organization. In section 1.2, we give a detailed overview the shadow vertex simplex method and its associated geometry. In section 1.3, we analyze the successive shortest path algorithm for minimum-cost maximum-flow under objective perturbations. In section 1.4, we give an analysis for general LPs under Gaussian constraint perturbations.

1.2 The Shadow Vertex Simplex Method

The shadow vertex simplex algorithm is a simplex method which, given two objectives c, d and an initial vertex v maximizing c, computes a path corresponding to vertices that are optimal (maximizing) for any intermediary objective $\lambda c+(1-\lambda)d, \lambda \in [0,1]$.

While the shadow vertex rule is not generally used in practice, e.g. the steepest descent rule is empirically far more efficient, it is much easier to analyze from the theoretical perspective as it admits a tractable characterization of the vertices it visits. Namely, a vertex can only be visited if it optimizes an objective between c and d, which can be checked by solving an LP.

In what follows, we overview the main properties of the shadow vertex simplex method together with how to implement it algorithmically. For this purpose, we will need the following definitions.

Definition 1.1 (Optimal Face)**.** For $P \subseteq \mathbb{R}^n$ a polyhedron and $c \in \mathbb{R}^n$, define $P[c] := \{x \in P : c^\mathsf{T} x = \sup_{z\in P} c^\mathsf{T} z\}$ to be the face of P maximizing c. If $\sup_{z\in P} c^\mathsf{T} z = \infty$, then $P[c] = \emptyset$ and we say that P is unbounded w.r.t. c.

Note that, in this notation, if $P[c] = P[d] \neq \emptyset$, for $d \in \mathbb{R}^n$, then $P[c] = P[\lambda c + (1-\lambda)d]$ for all $\lambda \in [0,1]$.

Definition 1.2 (Tangent Cone)**.** Let $P = \{x \in \mathbb{R}^n : Ax \leq b\}$, $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, be a polyhedron. For $x \in P$, define $\mathrm{tight}_P(x) = \{i \in [m] : a_i^\mathsf{T} x = b_i\}$ to be the tight constraints at x. The tangent cone at x w.r.t. P is $T_P(x) := \{w \in \mathbb{R}^n : \exists \varepsilon > 0 \text{ s.t. } x + \varepsilon w \in P\}$, the set of movement directions around x in P. In terms of the inequality representation, $T_P(x) := \{w \in \mathbb{R}^n : A_B w \leq 0\}$ where $B = \mathrm{tight}_P(x)$.

The Structure of the Shadow Path. The following lemma provides the general structure of any shadow path, which will generically induce a valid simplex path.

Lemma 1.3 (Shadow Path)**.** *Let $P \subseteq \mathbb{R}^n$ be a polyhedron and $c, d \in \mathbb{R}^n$. Then there exists a unique sequence of faces $P(c,d) := (v_0, e_1, v_1, \cdots, e_k, v_k)$ of P, $k \geq 0$, known as the shadow path of P w.r.t. (c,d), and scalars $0 = \lambda_0 < \lambda_1 < \cdots < \lambda_k < \lambda_{k+1} = 1$ such that*

1. *For all $1 \leq i \leq k$, we have $e_i = P[(1-\lambda_i)c + \lambda_i d] \neq \emptyset$, and moreover $e_1, \ldots, e_k$ are distinct faces of P.*
2. *For all $0 \leq i \leq k$ and $\lambda \in (\lambda_i, \lambda_{i+1})$, $v_i = P[(1-\lambda)c + \lambda d]$.*
3. *For all $0 < i < k$, the faces satisfy $v_i = e_i \cap e_{i+1} \neq \emptyset$, and if $k \geq 1$ then $v_0 \subset e_1$ and $v_k \subset e_k$.*

Furthermore, the first face is $v_0 = P[c][d]$, the face of $P[c]$ maximizing d, and the last face is $v_k = P[d][c]$. For every $i \in [k]$, we have $v_{i-1} = e_i[c] = e_i[-d]$ and $v_i = e_i[-c] = e_i[d]$.

Note that, as a set, the shadow path $P(c,d)$ exactly corresponds to the set of faces $\{P[(1-\lambda)c + \lambda d] : \lambda \in (0,1)\}$ optimizing an objective in (c,d). Lemma 1.3 shows that these faces have a useful connectivity structure that we will exploit algorithmically.

Definition 1.4 (Shadow Path Properties)**.** Given a polyhedron P, $c, d \in \mathbb{R}^n$, letting $P(c,d) = (v_0, e_1, v_1, \ldots, e_k, v_k)$, we use $P_V(c,d)$ to denote the subsequence of non-empty faces of $(v_0, v_1, \ldots, v_k)$ and $P_E(c,d) = (e_1, e_2, \ldots, e_k)$. We call each face $F \in P(c,d)$ a shadow face. We define the shadow path $P(c,d)$ to be non-degenerate if $\dim(v_0) \leq 0$ and $e_1, \ldots, e_k$ are edges of P. Note that this automatically enforces that $v_1, \ldots, v_{k-1}$ are vertices of P and that $\dim(v_k) \leq 0$. We say that $P(c,d)$ is proper if $P[c][d] \neq \emptyset$.

We are interested in the case when shadow paths are proper and non-degenerate. For a proper non-degenerate path $P(c,d) = (v_0, \ldots, e_k, v_k)$, the set $v_0 \cup \cup_{i=1}^k e_i$ is a connected polygonal path that begins at the vertex $v_0 = P[c][d]$ and follows edges of P, and thus forms a valid simplex path. The final face v_k will be non-empty iff P is bounded w.r.t. d. In this case, $v_k = P[d][c]$ is the vertex of $P[d]$ maximizing c. If $v_k = \emptyset$, then e_k will be an unbounded edge of the form $e_k = v_{k-1} + [0, \infty) \cdot w_k$ for which $w_k^\mathsf{T} d > 0$, yielding a certificate of the unboundedness of P w.r.t. d.

A useful way to interpret the shadow path is via a two-dimensional projection induced by c, d. We index this projection by $\pi_{c,d}$, where $\pi_{c,d}(z) := (d^\mathsf{T} z, c^\mathsf{T} z)$, and define $e_x := (1,0), e_y := (0,1)$ to be the generators of the x and y axis in $\mathbb{R}^2$ respectively. Under this map, the faces of the shadow path trace a path along the upper hull of $\pi_{c,d}(P)$. The projection interpretation is the reason why Borgwardt (1977) called the parametric objective method the shadow vertex simplex method (*schatteneckenalgoritmus*), which is the most common name for it today.

Lemma 1.5. *Let P be a polyhedron, $c, d \in \mathbb{R}^n$. For $P(c,d) = (v_0, e_1, v_1, \ldots, e_k, v_k)$,*

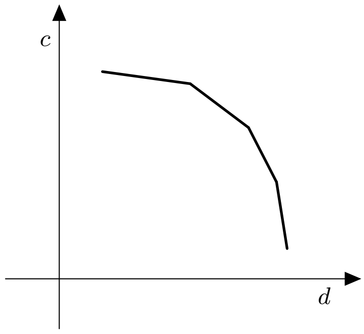

Figure 1.1 In (c,d) space, a shadow path starts at the highest vertex and moves to the rightmost vertex if they exist.

the shadow path satisfies $\pi_{c,d}(P)(e_y,e_x) = (\pi_{c,d}(v_0),\dots,\pi_{c,d}(e_k),\pi_{c,d}(v_k))$. *Furthermore, the shadow path* $\pi_{c,d}(P)(e_y,e_x)$ *is non-degenerate and* $P(c,d)$ *is non-degenerate iff* $\dim(v_0) = \dim(\pi_{c,d}(v_0))$ *and* $\dim(e_i) = \dim(\pi_{c,d}(e_i)) = 1$ *for all* $i \in [k]$.

Lemma 1.5 in fact implies that non-degeneracy can be restated as requiring $\pi_{c,d}$ to be a bijection between $S = v_0 \cup \bigcup_{i=1}^k e_i$ and its projection $\pi_{c,d}(S)$. Non-degeneracy of a shadow path is in fact a generic property. That is, given any pointed polyhedron $P \subseteq \mathbb{R}^n$ and objective d, the set of objectives c for which $P(c,d)$ is degenerate has measure 0 in $\mathbb{R}^n$. As a consequence, given any c and d, one may achieve non-degeneracy by infinitessimally perturbing either c or d.

Under the $\pi_{c,d}$ projection, the faces $v_0,\dots,v_k$, except possibly v_0, v_k which may be empty, always map to vertices of $\pi_{c,d}(P)$, and the faces $e_1,\dots,e_k$ always map to edges of $\pi_{c,d}(P)$. Assuming $v_0, v_k \neq \emptyset$, then $\pi_{c,d}(v_0), \pi_{c,d}(v_k)$ are the vertices of maximum y and x coordinate respectively in $\pi_{c,d}$, and the edges $\pi_{c,d}(e_1),\dots,\pi_{c,d}(e_k)$ follow the upper hull of $\pi_{c,d}(P)$ between $\pi_{c,d}(v_0)$ and $\pi_{c,d}(v_k)$ from left to right. In this view, one can interpret the multipliers $\lambda_1 < \cdots < \lambda_k \in (0,1)$ from Lemma 1.3 in terms of the slopes of $e_1,\dots,e_k$ under $\pi_{c,d}$. Precisely, if we define the c,d slope $s_{c,d}(e_i) := c^\mathsf{T}(x_1-x_0)/d^\mathsf{T}(x_1-x_0)$, $i \in [k]$, where $x_1, x_0 \in e_i$ are any two points with $d^\mathsf{T}x_1 \neq d^\mathsf{T}x_0$, then $s_{c,d}(e_i) = -\lambda_i/(1-\lambda_i)$. This follows directly from the fact that the objective $(1-\lambda_i)c + \lambda_i d$, $\lambda_i \in (0,1)$, is constant on e_i. From this, we also see that $0 > s_{c,d}(e_1) > \cdots > s_{c,d}(e_k)$, that is the slopes are negative and strictly decreasing.

The Shadow Vertex Simplex Algorithm. A shadow vertex pivot, i.e. a move across an edge of P, will correspond to moving in a direction of largest (c,d) slope from the current vertex. Computing these directions will be achieved by solving linear programs over the tangent cones. In the context of the successive shortest path algorithm, these LPs are solved via a shortest path computation, while in the

Gaussian constraint perturbation model, they are solved explicitly by computing the extreme rays of the tangent cone. An abstract implementation of the shadow vertex simplex method is provided in Algorithm 1. While there is technically freedom in the choice of the maximizer on line 3, under non-degeneracy the solution will in fact be unique. We state the main guarantees of the algorithm below.

Algorithm 1 The Shadow Vertex Simplex Algorithm

Require: $P = \{x \in \mathbb{R}^n : Ax \leq b\}$, $c, d \in \mathbb{R}^n$, initial vertex $v_0 \in P[c][d] \neq \emptyset$.
Ensure: Return vertex of $P[d][c]$ if non-empty or $e \in \text{edges}(P)$ unbounded w.r.t. d.

1: $i \leftarrow 0$
2: **loop**
3:   $w_{i+1} \leftarrow$ vertex of $\text{argmax}\{c^\mathsf{T} w : w \in T_P(v_i), d^\mathsf{T} w = 1\}$ or $\emptyset$ if infeasible
4:   **if** $w_{i+1} = \emptyset$ **then**
5:     **return** v_i
6:   **end if**
7:   $\lambda_{i+1} \leftarrow -w_{i+1}^\mathsf{T} c/(1 - w_{i+1}^\mathsf{T} c)$
8:   $s_{i+1} \leftarrow \sup\{s \geq 0 : v_i + s w_{i+1} \in P\}$
9:   $e_{i+1} \leftarrow v_{i+1} + [0, s_{i+1}] \cdot w_{i+1}$
10:   $i \leftarrow i + 1$
11:   **if** $s_i = \infty$ **then**
12:     $v_i \leftarrow \emptyset$
13:     **return** e_i
14:   **else**
15:     $v_i \leftarrow v_{i-1} + s_i w_i$
16:   **end if**
17: **end loop**

Theorem 1.6. *Algorithm 1 is correct and finite. On input P,c,d,$v_0 \in P[c][d] \neq \emptyset$, the vertex–edge sequence $v_0, e_1, v_1, \ldots, e_k, v_k$ computed by the algorithm visits every face of $P(c, d)$ and the computed multipliers $\lambda_1, \ldots, \lambda_k \in (0, 1)$ form a non-decreasing sequence which satisfies $e_i \subseteq P[(1-\lambda_i)c+\lambda_i d]$ for every $i \in [k]$. If $P(c, d)$ is non-degenerate, then $(v_0, e_1, v_1, \ldots, e_k, v_k) = P(c, d)$. Furthermore, the number of simplex pivots performed is then $|P_E(c, d)|$, and the complexity of the algorithm is that of solving $|P_V(c, d)|$ tangent cone programs.*

In regard to slopes, the value of the program on line 3 equals the (c, d)-slope $s_{c,d}(e_{i+1})$.

While Algorithm 1 still works in the presence of degeneracy, one can no longer characterize the number of pivots by $|P_E(c, d)|$, though this always remains a lower bound. This is because it may take multiple pivots to cross a single face of $P_E(c, d)$, or equivalently, there can be a consecutive block $[i, j]$ of iterations where $\lambda_i = \cdots = \lambda_j$.

As is evident from the theorem and the algorithm, the complexity of each iteration depends on the difficulty of solving the tangent cone programs on line 3. One instance in which this is easy, is when the inequality system is *non-degenerate.*

Definition 1.7 (Non-degenerate Inequality System)**.** We say that the system $Ax \leq b$, $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $m \geq n$, describing a polyhedron P is *non-degenerate* if P is pointed and if for every vertex $v \in P$ the set $\mathrm{tight}_P(v)$ is a *basis* of A.

When the description of P is clear, we say that P is non-degenerate to mean that its describing system is. We call $B \subseteq [m]$, $|B| = n$, a *basis* of A if A_B, the submatrix corresponding to the rows in B, is invertible. A basis B is feasible if $A_B^{-1} b_B$ is a vertex of P. For a non-degenerate polyhedron P and $v \in \mathrm{vertices}(P)$, the extreme rays of the tangent cone at v are simple to compute. More precisely, letting $B = \mathrm{tight}_P(v)$ denote the basis for v, the extreme rays of the tangent cone $T_P(v)$ are generated by the columns of $-A_B^{-1}$. Knowing this explicit description of the extreme rays of $T_P(v)$, the program on line 3 is easy to solve because w_{i+1} is always a scalar multiple of a generator of an extreme ray.

The Shadow Plane and the Polar. In the previous subsection, we examined the shadow path $P(c, d)$ induced by two objectives c, d. This is enough for the result we prove in section 1.3. For the sake of section 1.4, we generalize the shadow path slightly by examining the shadow on the plane $W = \mathrm{span}(c, d)$. Letting π_W denote the orthogonal projection onto W, we will work with $\pi_W(P)$, the shadow of P on W. This will be useful to capture somewhat more global shadow properties. In particular, it will allow us to relate to the geometry of the corresponding polar, and allow us to get bounds on the lengths of shadow paths having knowledge of W, but not of the exact objectives $c, d \in W$ whose shadow path we will follow.

Definition 1.8 (Shadow on W)**.** Let $P \subseteq \mathbb{R}^n$ be a polyhedron and let $W \subseteq \mathbb{R}^n$ be a 2-dimensional linear subspace. We define the shadow faces of P w.r.t. W by $P[W] = \{P[c] : c \in W \setminus \{0\}\}$, that is the set of faces of P optimizing a non-zero objective in W. Let $P_V[W]$,$P_E[W]$ denote the set of faces in $P[W]$ projecting to vertices and edges of $\pi_W(P)$ respectively. We define $P[W]$ to be non-degenerate if every face $F \in P[W]$ satisfies $\dim(F) = \dim(\pi_W(F))$.

The following lemma provides the straightforward relations between shadow paths on W and the number of vertices of $\pi_W(P)$.

Lemma 1.9. *Let $P \subseteq \mathbb{R}^n$ be a polyhedron, $W \subseteq \mathbb{R}^n$, $\dim(W) = 2$. Then for $c, d \in W$, if the path $P(c, d)$ is non-degenerate and proper, then the number of pivots performed by Algorithm 1 on input $P, c, d, P[c][d]$ is bounded by $|P_V[W]| = |\mathrm{vertices}(\pi_W(P))|$. Furthermore, if $P[W]$ is non-degenerate and $\mathrm{span}(c, d) = W$, then $P(c, d)$ is non-degenerate.*

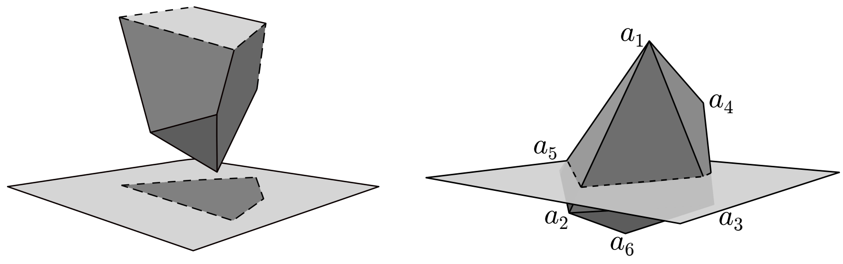

Figure 1.2 On the left, we see a polyhedron P projected on a plane W. The boundary of the projection uniquely lifts into the polyhedron. On the right, we see the corresponding polar polytope $Q = P^\circ$ with the intersection $Q \cap W$ marked. Every facet of Q intersected by W is intersected through its relative interior.

Moving to the polar view, we assume that we start with a polyhedron of the form $P = \{x \in \mathbb{R}^n : Ax \leq 1\}$. Define the polar polytope as $Q = \mathrm{conv}(a_1, \ldots, a_m)$, the convex hull of $a_1, \ldots, a_m$, where $a_1, \ldots, a_m$ are the rows of the constraint matrix A. We use a slightly different definition of the polar polytope than is common. The standard definition takes the polar to be

$$P^\circ := \{y \in \mathbb{R}^n : y^\mathsf{T} x \leq 1, \forall x \in P\} = \mathrm{conv}(Q, 0).$$

We have $P^\circ \neq Q$ exactly when P is unbounded. We depict a polyhedron and its polar polytope in Figure 1.2.

The following lemma, which follows from relatively standard polyhedral duality arguments, tells us that one can control the vertex count of the shadow using the corresponding slice of the polar. It provides the key geometric quantity we will bound in section 1.4. Proving the lemma is Exercise 1.2.

Lemma 1.10. *Let $P = \{x \in \mathbb{R}^n : Ax \leq 1\}$ be a polyhedron with a non-degenerate shadow on W and Q its polar polytope. Then*

$$|\mathrm{vertices}(\pi_W(P))| \leq |\mathrm{edges}(Q \cap W)|.$$

If P is bounded then the inequality is tight.

1.3 The Successive Shortest Path Algorithm

In this section, we will study the classical successive shortest path (SSP) algorithm for the minimum-cost maximum-flow problem under objective perturbations.

The Flow Polytope. Given a directed graph $G = (V, E)$ with source $s \in V$ and sink $t \in V$, a vector of positive arc capacities $u \in \mathbb{R}^E_+$, and a vector of arc costs $c \in (0, 1)^E$, we want to find a flow $f \in \mathbb{R}^E_+$ satisfying

$$\sum_{ij \in E} f_{ij} - \sum_{ji \in E} f_{ji} = 0, \forall i \in V \setminus \{s, t\} \qquad (1.2)$$
$$0 \leq f_{ij} \leq u_{ij}, \forall ij \in E$$

that maximizes the amount of flow shipped from s to t, and among such flows minimizes the cost $c^\mathsf{T} f$. We denote the set of feasible flows, that is, those satisfying (1.2), by P.

For simplicity of notation in what follows, we assume that G does not have bidirected arcs, that is E contains at most one of any pair $\{ij, ji\}$. To make the identification with the shadow vertex simplex method easiest, we consider only the case in which every shortest s-t path is unique.

The SSP Algorithm. We now describe the algorithm. For this purpose, we introduce some notation. Letting $\overleftarrow{ij} = ji$, we define the reverse arcs $\overleftarrow{E} := \{ji : ij \in E\}$, $\overleftrightarrow{E} = E \cup \overleftarrow{E}$, and extend c to $\overleftarrow{E}$ by letting $c_{ji} = -c_{ij}$ for $ji \in \overleftarrow{E}$. For $w \in \{-1, 0, 1\}^E$, we define its associated subgraph $R = \{a \in E : w_a = 1\} \cup \{\overleftarrow{a} : a \in E, w_a = -1\}$ and vice versa, noting that $c^\mathsf{T} w = \sum_{a \in R} c_a$. Given a feasible flow $f \in P$, the *residual graph* $N[f]$ has the same node set V and arc set $A[f] = F[f] \cup R[f] \cup B[f]$, where $F[f] = \{a \in E : f_a = 0\}$, $R[f] = \{\overleftarrow{a} : a \in E : f_a = u_a\}$, $B[f] = \{a, \overleftarrow{a} : a \in E, 0 < f_a < u_a\}$ are called forward, reverse and bidirected arcs w.r.t. f respectively. The combinatorial description of the SSP algorithm is provided below:

1. Initialize f to 0 on E.
2. While $N[f]$ contains an s-t path: compute a shortest s-t path R in $N[f]$ with respect to the costs c with associated vector $w_R \in \{-1, 0, 1\}^E$. Augment f along R until a capacity constraint becomes tight, that is update $f \leftarrow f + s_R w_R$ where $s_R = \max\{s \geq 0 : f + s_R w_R \in P\}$. Repeat.
3. Return f.

We recall that a shortest s-t path is well-defined if and only if $N[f]$ does not contain negative cost cycles.

For the SSP algorithm to take many iterations to find the optimum solution, the difference between the path lengths in each iteration should be very small. As long as the costs are not adversarially chosen, it seems unlikely that this should happen. That is what we formalize and prove in the remainder of this chapter.

The SSP as Shadow Vertex. We now show that the SSP algorithm corresponds to running the shadow vertex simplex algorithm on P applied to the starting objective being $-c$ and the target objective d being the flow from s to t, that is $d^\mathsf{T} f := \sum_{sj \in E} f_{sj}$. This correspondence will also show correctness of the SSP.

To see the link to the shadow vertex simplex algorithm, we reinterpret prior observations polyhedrally. Firstly, it is direct to check that the face $P[d]$ is the set of maximum s-t flows. In particular, the maximum-flow of minimum cost is then $P[d][-c]$. Since the arc costs are positive on E, any non-zero flow $f \in P$ must incur positive cost. Therefore, the zero flow is the unique cost minimizer, that is $\{0\} = P[-c] = P[-c][d]$. Thus, by Theorem 1.6, one can run the shadow vertex simplex algorithm on the flow polytope P, objectives $-c$,d and starting vertex 0 and get a vertex $v \in P[d][-c]$ as output.

To complete the identification, one need only show that the tangent cone LPs correspond to shortest s-t path computations. This is a consequence of the following lemma, whose proof is left as Exercise 1.3.

Lemma 1.11. *For $f \in P$ with residual graph $N[f]$, the following hold:*

1. *The tangent cone can be explicitly described using flow conservation and tight capacity constraints as $T_P(f) := \{w \in \mathbb{R}^A : \sum_{ij \in A} w_{ij} - \sum_{ji \in A} w_{ji} = 0 \ \forall i \in V \setminus \{s,t\}, w_a \geq 0 \ \forall a \in F[f], w_a \leq 0 \ \forall a \in R[f]\}$.*
2. *If $N[f]$ does not contain negative cost cycles, then any vertex solution to the program $\inf\{c^\mathsf{T} w : w \in T_P(f), d^\mathsf{T} w = \delta\}$, $\delta \in \{\pm 1\}$ corresponds to a minimum-cost s-t path for $\delta = 1$ and t-s path for $\delta = -1$, which by convention has cost ∞ if no such path exists.*
3. *If f is a shadow vertex and the shadow path is non-degenerate, the value of the above program for $\delta = 1$ equals the slope $s_{c,d}(e)$ of the shadow edge e leaving f and the value of the program for $\delta = -1$ equals the slope $s_{c,d}(e')$ of the shadow edge e' entering f.*

It will be useful to note here that since we interpolate from $-c$, that is minimizing cost, the shadow $P(-c,d)$ will in fact follow edges of the lower hull of $\pi_{c,d}(P)$ from left to right. In particular, the (c,d) slopes (cost per unit of flow) of the corresponding edges will all be positive and form an increasing sequence. The (c,d) slope of a shadow edge is always equal to the cost of some s-t path $\overleftrightarrow{E}$. Since any such path uses at most $n-1$ edges of cost between $(-1,1)$, the slope of any shadow edge is strictly less than $n-1$, which will be crucial to the analysis in the next section. By the correspondence of slopes with multipliers, the slope bound implies the rather strong property that any maximizer of $-c + \frac{n-1}{n} d$ in P, is already on the optimal face $P[d][-c]$.

1.3.1 Smoothed Analysis of the SSP

As shown by Zadeh (1973), there are inputs where the SSP algorithm requires an exponential number of iterations to converge. In what follows, we explain the main result of Brunsch et al. (2015), which shows that exponential behavior can be remedied by slightly perturbing the edge costs.

The perturbation model is known as the one-step model, which is a general model where we only control the support and maximum density of the perturbations. Precisely, each edge cost c_e will be a continuous random variable supported on $(0, 1)$, whose maximum density is upper bounded by a parameter $\phi \geq 1$. The upper bound on ϕ is equivalent to the statement that for any interval $[a, b] \subseteq [0, 1]$, the inequality $\Pr[c_e \in [a, b]] \leq \phi|b - a|$, known as the interval lemma, holds. Note that as $\phi \to \infty$, the cost vector c can concentrate on a single vector, and thus converge to a worst-case instance. The main result of this section is as follows.

Theorem 1.12 (Brunsch et al. (2015))**.** *Let $G = (V, E)$ be a graph with n nodes and m arcs, a source $s \in V$ and sink $t \in V$, and positive capacities $u \in \mathbb{R}^E_+$. Then for a random cost vector $c \in (0, 1)^E$ with independent coordinates having maximum density $\phi \geq 1$, the expected number of iterations of the SSP algorithm on G is bounded by $O(mn\phi)$.*

As with many smoothed analysis results, we want to quantify some form of "expected progress" per iteration, and the difficulty lies in identifying enough "independent randomness" such that not all randomness is used up in the first iteration.

To prove the theorem, we will upper bound the expected number of edges on the random shadow path followed by the SSP. The main idea will be to bound the expected number of times an arc of G can used by the s-t paths found by the SSP algorithm.

For the analysis, we maintain the notation from the previous section together with the following definitions. For $f \in P$, we identify the tight constraints $\mathrm{tight}_P(f)$ with arcs in $\overleftrightarrow{E}$, namely $a \in \mathrm{tight}_P(f)$ iff $a \in E$ and $f_{ij} = 0$ or $a \in \overleftarrow{E}$ and $f_{ij} = u_{ij}$. Similarly, we define $P_a = \{f \in P : a \in \mathrm{tight}_P(f)\}$. To identify (c, d) slopes, for any $f \in P$, we use $p_{s,t}(f), p_{t,s}(f) \in \mathbb{R} \cup \{\pm\infty\}$ to denote the cost of the shortest s-t and t-s path in $N[f]$. Similarly, for $a \in \overleftrightarrow{E}$, we use $p^{\pm a}_{s,t}(f), p^{\pm a}_{t,s}(f)$ to denote the corresponding minimum-cost paths not using arc a (superscript $-a$) and using arc a (superscript $+a$).

Proof of Theorem 1.12 To prove the theorem, we show that $\mathbb{E}_c[|P_E(-c, d)|]$, the expected shadow vertex count, is bounded by $O(mn\phi)$. Since the cost vector c is generic, the shadow path $P(-c, d)$ is non-degenerate with probability 1. By Theorem 1.6, this will establish the desired bound on the number of shadow vertex pivots.

Let $(v_0, e_1, v_1, \ldots, e_k, v_k)$ denote the random shadow path $P(-c, d)$, and similarly for $a \in \overleftrightarrow{E}$, let $(v^a_0, e^a_1, v^a_1, \ldots, e^a_{k_a}, v^a_{k_a})$ be the shadow path $P_a(-c, d)$, which we may assume to be non-degenerate with probability 1. Note that since P is a polytope, each shadow path is either $\emptyset$ (if the corresponding facet is infeasible) or contains no empty faces. By the natural extension of Lemma 1.11 to facets of P, we have that for $a \in \overleftrightarrow{E}$ and $i \in [k_a]$, the (c,d) slope of edge e_i^a is equal to $s_{c,d}(e_i^a) = p_{s,t}^{a-}(v_{i-1}^a) = -p_{t,s}^{a-}(v_i^a)$, i.e. the corresponding shortest path length restricted to not using arc a.

Since each vertex $v_{i-1} \subset e_i$, $i \in [k]$, is contained in its outgoing edge, there must exist a tight constraint $a \in \mathrm{tight}_P(v_{i-1})$ such that $a \notin \mathrm{tight}_P(e_i)$. This yields the following direct inequality:

$$|P_E(-c,d)| = \sum_{i=1}^{k} 1 \leq \sum_{a \in \overleftrightarrow{E}} \sum_{i=1}^{k} 1[a \in \mathrm{tight}_P(v_i), a \notin \mathrm{tight}_P(e_i)]. \tag{1.3}$$

Fixing $a \in \overleftrightarrow{E}$, we now show that the corresponding term in (1.3) is bounded on expectation over c by $O(n\phi)$. For $i \in [k]$, since the (c,d) slope satisfies $s_{c,d}(e_i) = p_{s,t}(v_{i-1})$, we know that $a \in \mathrm{tight}_P(v_{i-1}) \setminus \mathrm{tight}_P(e_i)$ implies that the minimum-cost s-t path in $N[v_i]$ uses arc a. In particular, $p_{s,t}(v_{i-1}) = p_{s,t}^{a+}(v_{i-1})$. Since $-p_{t,s}(v_{i-1})$ is the (c,d) slope of the incoming edge at v_{i-1}, by the increasing property of slopes we also have the inequality $-p_{t,s}(v_{i-1}) \leq p_{s,t}^{a+}(v_{i-1})$. Putting this information together,

$$\begin{aligned}
&\sum_{i=1}^{k} 1[a \in \mathrm{tight}_P(v_{i-1}), a \notin \mathrm{tight}_P(e_i)] \\
&\qquad \leq \sum_{i=0}^{k-1} 1[a \in \mathrm{tight}_P(v_i), -p_{t,s}(v_i) \leq p_{s,t}^{a+}(v_i) \leq p_{s,t}(v_i)] \\
&\qquad \leq \sum_{i=0}^{k-1} 1[a \in \mathrm{tight}_P(v_i), -p_{t,s}^{a-}(v_i) \leq p_{s,t}^{a+}(v_i) \leq p_{s,t}^{a-}(v_i)],
\end{aligned}$$

where last inequality follows from the trivial inequalities $p_{s,t}^{a-}(v_i) \geq p_{s,t}(v_i)$ and $p_{t,s}^{a-}(v_i) \geq p_{t,s}(v_i)$. We now make the link to the shadow on P_a. Since v_i is a shadow face in $P(-c,d)$, $a \in \mathrm{tight}_P(v_i)$ implies that v_i is also a shadow face of $P_a(-c,d)$. By this containment and the characterization of edge slopes in $P_a(-c,d)$ as shortest path lengths, we have that

$$\begin{aligned}
&\sum_{i=0}^{k-1} 1[a \in \mathrm{tight}_P(v_i), -p_{t,s}^{a-}(v_i) \leq p_{s,t}^{a+}(v_i) \leq p_{s,t}^{a-}(v_i)] \\
&\qquad = \sum_{i=0}^{k-1} 1[v_i \in P_a(-c,d), -p_{t,s}^{a-}(v_i) \leq p_{s,t}^{a+}(v_i) \leq p_{s,t}^{a-}(v_i)] \\
&\qquad \leq \sum_{i=0}^{k_a} 1[-p_{t,s}^{a-}(v_i^a) \leq p_{s,t}^{a+}(v_i^a) \leq p_{s,t}^{a-}(v_i^a)] \\
&\qquad \leq 2 + \sum_{i=1}^{k_a-1} 1[s_{c,d}(e_i^a) \leq p_{s,t}^{a+}(v_i^a) \leq s_{c,d}(e_{i+1}^a)].
\end{aligned}$$

We may now usefully take an expectation with respect to c_a. The crucial observa-

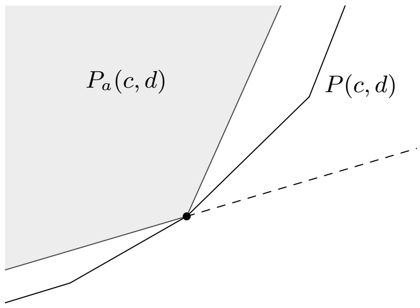

Figure 1.3 Any vertex of $P(c,d)$ is a vertex of some $P_a(c,d)$, and the outgoing edge on $P(c,d)$ has slope between the slopes of the adjacent edges of $P_a(c,d)$.

tion here is that by independence of the components of c, the shadow path $P_a(-c,d)$ is independent of the cost c_a, noting that the flow along arc a is fixed in P_a. Furthermore, expressing $a = pq \in \overleftrightarrow{E}$, we may usefully decompose $p^{a+}_{s,t}(v^a_i) = c_a + r^{a+}_{s,t}(v^a_i)$, where $r^{a+}_{s,t}(v^a_i)$ is the sum of the cost of the shortest s-p and q-t paths in $N[v^a_i]$. Noting that $N[v^a_i]$ does not contain $\overleftarrow{a}$, we see that $r^{a+}_{s,t}(v^a_i)$ is clearly independent of c_a. Using that edge slopes satisfy $0 < s_{c,d}(e^a_1) < \cdots < s_{c,d}(e^a_{k_a}) \leq n-1$, where the last inequality follows as before by the correspondence with s-t path lengths, together with the interval lemma, we bound the expectation as follows:

$$
\begin{aligned}
&\mathbb{E}_{c_a}[\sum_{i=1}^{k_a-1} 1[s_{c,d}(e^a_i) \leq p^{a+}_{s,t}(v^a_i) \leq s_{c,d}(e^a_{i+1})]] \\
&\qquad\qquad = \sum_{i=1}^{k_a-1} \Pr_{c_a}[c_a + r^{a+}_{s,t}(v^a_i) \in [s_{c,d}(e^a_i), s_{c,d}(e^a_{i+1})]] \\
&\qquad\qquad \leq \sum_{i=1}^{k_a-1} \phi\left(s_{c,d}(e^a_{i+1}) - s_{c,d}(e^a_i)\right) \\
&\qquad\qquad = \phi\left(s_{c,d}(e^a_{k_a}) - s_{c,d}(e^a_1)\right) \leq (n-1)\phi.
\end{aligned}
$$

Putting it all together, using that $|\overleftrightarrow{E}| = 2m$, we derive the desired expected bound

$$
\begin{aligned}
\mathbb{E}_c[|P_E(-c,d)|] &\leq \sum_{a\in\overleftrightarrow{E}} \mathbb{E}_c[\sum_{i=1}^{k} 1[a \in \mathrm{tight}_P(v_{i-1}), a \notin \mathrm{tight}_P(e_i)]] \\
&\leq 4m + \sum_{a\in\overleftrightarrow{E}} \mathbb{E}_c[\sum_{i=1}^{k_a-1} 1[s_{c,d}(e^a_i) \leq p^{a+}_{s,t}(v^a_i) \leq s_{c,d}(e^a_{i+1})]] \\
&\leq 4m + 2m\phi(n-1) = O(mn\phi). \qquad \square
\end{aligned}
$$

1.4 LPs with Gaussian Constraints

The Gaussian constraint perturbation model in this section was the first smoothed complexity model to be studied and was introduced by Spielman and Teng (2004). While not entirely realistic, since it does not preserve for example the sparsity structure seen in most real-world LPs, it does show that the worst-case behavior of the simplex method is very brittle. Namely, it shows that a shadow vertex simplex method efficiently solves most LPs in any big enough neighborhood around a base LP. At a very high level, this because an average shadow vertex pivot covers a significant fraction of the "distance" between the initial and target objective.

The Gaussian Constraint Perturbation Model. In this perturbation model, we start with any base LP

$$\max c^{\mathsf{T}} x,\ \bar{A}x \leq \bar{b}, \qquad \text{(Base LP)}$$

$\bar{A} \in \mathbb{R}^{m\times n}$, $\bar{b} \in \mathbb{R}^m$, $c \in \mathbb{R}^n \setminus \{0\}$, where the rows of $(\bar{A}, \bar{b})$ are normalized to have ℓ_2 norm at most 1. From the base LP, we generate the smoothed LP by adding Gaussian perturbations to both the constraint matrix $\bar{A}$ and the right-hand side $\bar{b}$. Precisely, the data of the smoothed LP is $A = \bar{A} + \hat{A}, b = \bar{b} + \hat{b},\ c$ where the perturbations $\hat{A}$,$\hat{b}$ have i.i.d. mean 0, variance σ^2 Gaussian entries. The goal is to solve

$$\max c^{\mathsf{T}} x,\ Ax \leq b. \qquad \text{(Smoothed LP)}$$

Note that we do not need to perturb the objective in this model, though we do require that $c \neq 0$. The base LP data must be normalized for this definition to make sense, since otherwise one could scale the base LP data up to make the effective perturbation negligible.

As noted earlier, the strength of the shadow vertex simplex algorithm lies in it being easy to characterize whether a basis is visited given the starting and final objective vectors. There is no dependence on decisions made in previous pivot steps. To preserve this independence, we have to be careful with how we find our initial vertex and objective. On the one hand, if we start out knowing a feasible basis $B \subset [m]$ of the smoothed LP, we cannot just set $d = \sum_{i\in B} a_i$, where $a_1, \ldots, a_m$ denote the rows of A. This would cause the shadow plane $\mathrm{span}(c, d)$ to depend on A and make our calculations rather more difficult. On the other hand, we cannot choose our starting objective d independently of A, b and find the vertex optimizing it, because that is the very problem that we aim to solve. We resolve this by analyzing the expected shadow vertex count on a plane that is independent of A, b and designing an algorithm that uses the shadow vertex simplex method as a subroutine only on objectives that lie inside such pre-specified planes.

Smoothed Unit LPs. As a further simplification of the probabilistic analysis, we restrict our shadow bounds to LP's with right-hand side equal to 1 and only A

perturbed with Gaussian noise:

$$\max c^\mathsf{T} x,\ Ax \le 1. \qquad \text{(Smoothed Unit LP)}$$

This assumption guarantees that 0 is a feasible solution. In the rest of this subsection, we reduce solving (Smoothed LP) to solving (Smoothed Unit LP) and show how to solve (Smoothed Unit LP).

The next theorem is the central technical result of this section and will be proven in subsection 1.4.2. The bound carries over to the expected number of pivot steps of the shadow vertex simplex method on a smoothed unit LP with c, d in a fixed plane using Lemma 1.9 and Lemma 1.10.

Theorem 1.13. *Let $W \subset \mathbb{R}^n$ be a fixed two-dimensional subspace, $m \ge n \ge 3$ and let $a_1, \dots, a_m \in \mathbb{R}^n$ be independent Gaussian random vectors with variance σ^2 and expectations of norm at most 1. Then the expected number of edges is bounded by*

$$\mathbb{E}[|\mathrm{edges}(\mathrm{conv}(a_1, \dots, a_m) \cap W)|] = O(n^2 \sqrt{\ln m}\ \sigma^{-2} + n^{2.5} \ln m\ \sigma^{-1} + n^{2.5} \ln^{1.5} m).$$

The linear programs we solve and their shadows are non-degenerate with probability 1, so the above theorem will also bound the expected number of pivot steps of a run of the shadow vertex simplex method.

First, we describe an algorithm that builds on this shadow path length bound to solve general smoothed LP's. After that, we will sketch the proof of Theorem 1.13.

Two-Phase Interpolation Method. Given data A, b, c, define the Phase I Unit LP:

$$\begin{aligned} \max\ & c^\mathsf{T} x \qquad \text{(Phase I Unit LP)} \\ & Ax \le 1 \end{aligned}$$

and the Phase II interpolation LP with parametric objective for $\gamma \in (-\infty, \infty)$

$$\begin{aligned} \max\ & c^\mathsf{T} x + \gamma\lambda \qquad \text{(Int. LP)} \\ & Ax + (1-b)\lambda \le 1 \\ & 0 \le \lambda \le 1. \end{aligned}$$

We claim that, if we can solve smoothed unit LP's, then we can use the pair (Phase I Unit LP) and (Int. LP) to solve general smoothed LP's.

Writing P for the feasible set of (Int. LP) and e_λ for the basis vector in the direction of increasing λ, the optimal solution to (Phase I Unit LP) corresponds to $P[-e_\lambda][c]$. Assuming that (Smoothed LP) is feasible, its optimal solution corresponds to $P[e_\lambda][c]$. Both (Phase I Unit LP) and (Int. LP) are unit LP's. We first describe how to solve (Smoothed LP) given a solution to (Phase I Unit LP).

If (Smoothed LP) is unbounded (i.e., the system $c^\mathsf{T} x > 0, Ax \le 0$ is feasible), this will be detected during Phase I as (Unit LP) is also unbounded.

Let us assume for the moment that (Smoothed LP) is bounded and feasible (i.e., has an optimal solution). We can start the shadow vertex simplex method from the vertex $P[-e_\lambda][c]$ at objective $\gamma e_\lambda + c$, for some $\gamma < 0$ small enough, and move to maximize e_λ to find $P[e_\lambda][c]$.

If (Smoothed LP) is infeasible but bounded, then the shadow vertex run will terminate at a vertex having $\lambda < 1$. Thus, all cases can be detected by the two-phase procedure.

We bound the number of pivot steps taken to solve (Int. LP) given a solution to (Unit LP), and after that we describe how to solve (Unit LP).

Consider polyhedron $P' = \{(x, \lambda) \in \mathbb{R}^{n+1} : Ax + (1-b)\lambda \leq 1\}$, the slab $H = \{(x, \lambda) \in \mathbb{R}^{d+1} : 0 \leq \lambda \leq 1\}$ and let $W = \mathrm{span}(c, e_\lambda)$. In this notation, $P = P' \cap H$ is the feasible set of (Int. LP) and W is the shadow plane of (Int. LP). We bound the number of vertices in the shadow $\pi_W(P)$ of (Int. LP) by relating it to $\pi_W(P')$.

The constraint matrix of P' is $(A, 1-b)$, so the rows are Gaussian distributed with variance σ^2 and means of norm at most 2. After rescaling by a factor 2 we satisfy all the conditions for Theorem 1.13 to apply.

Since the shadow plane contains the normal vector e_λ to the inequalities $0 \leq \lambda \leq 1$, these constraints intersect the shadow plane W at right angles. It follows that $\pi_W(P' \cap H) = \pi_W(P') \cap H$. Adding 2 constraints to a 2D polyhedron can add at most 2 new edges, hence the constraints on λ can add at most 4 new vertices. By combining these observations, we directly derive the following lemma.

Lemma 1.14. *If (Unit LP) is unbounded, then (Smoothed LP) is unbounded. If (Unit LP) is bounded, then given an optimal solution to (Unit LP) one can solve (Smoothed LP) using an expected* $O(n^2\sqrt{\ln m}\ \sigma^{-2} + n^{2.5}\ln m\ \sigma^{-1} + n^{2.5}\ln^{1.5} m)$ *shadow vertex simplex pivots over (Int. LP).*

Given the above, our main task is now to solve (Unit LP), i.e., either to find an optimal solution or to determine unboundedness. The simplest algorithm that can operate using only pre-determined shadow planes is Borgwardt's dimension-by-dimension (DD) algorithm.

DD Algorithm. The DD algorithm solves Unit LP by iteratively solving the restrictions:

$$\begin{aligned} \max\ & {c^k}^\mathsf{T} x \qquad\qquad (\text{Unit LP}_k) \\ & Ax \leq 1 \\ & x_i = 0,\ \forall i \in \{k+1, \dots, n\}, \end{aligned}$$

where $k \in \{1, \dots, n\}$ and $c^k := (c_1, \dots, c_k, 0, \dots, 0)$. We assume that $c_1 \neq 0$ without loss of generality. The crucial observation in this context is that the optimal vertex v^* of (Unit LP$_k$), $k \in \{1, \dots, n-1\}$, is generically on an edge w^* of the shadow of (Unit LP$_{k+1}$) with respect to c^k and e_{k+1}. To initialize the (Unit LP$_{k+1}$) solve, we move to a vertex v_0 of the edge w^* and compute an objective $d \in \mathrm{span}(c^k, e_{k+1})$

uniquely maximized by v_0. Noting that $c^{k+1} \in \mathrm{span}(c^k, e_{k+1})$, we then solve (Unit LP$_{k+1}$) by running the shadow vertex simplex method from v_0 with starting objective d and target objective c^{k+1}.

We note that Borgwardt's algorithm can be applied to any LP with a known feasible point as long as appropriate non-degeneracy conditions hold (which occur with probability 1 for smoothed LPs). Furthermore, (Unit LP$_1$) is trivial to solve since the feasible region is an interval whose endpoints are straightforward to compute. By combining the arguments above, we get the following theorem.

Theorem 1.15. *Let S_k, $k \in \{2, \dots, n\}$, denote the shadow of (Unit LP$_k$) on $W_k = \mathrm{span}(c_{k-1}, e_k)$. Then, if each (Unit LP$_k$) and shadow S_k is non-degenerate for $k \in \{2, \dots, n\}$, the DD algorithm solves (Unit LP) using at most $\sum_{k=2}^{n} |\mathrm{vertices}(S_k)|$ number of pivots.*

To bound the number of vertices of S_k, we first observe that the feasible set of (Unit LP$_k$) does not depend on coordinates $k+1, \dots, n$ of the constraints vectors. Ignoring those, it is clear that there is an equivalent unit LP to (Unit LP$_k$) in just k variables. This equivalent unit LP has Gaussian distributed rows with variance σ^2 and means of norm at most 1.

Using Theorem 1.15 with the shadow bounds in Theorem 1.13, for $k \geq 3$, and Theorem 1.18 (proven below), for $k = 2$, we get the following complexity estimate for solving (Smoothed Unit LP).

Corollary 1.16. *The program (Smoothed Unit LP) can be solved by the DD algorithm using an expected number of shadow vertex pivots bounded by*

$$\sum_{k=2}^{n} \mathbb{E}[|\mathrm{edges}(\mathrm{conv}(a_1, \dots, a_m) \cap W_k)|] = O(n^3 \sqrt{\ln m}\, \sigma^{-2} + n^{3.5} \sigma^{-1} \ln m + n^{3.5} \ln^{3/2} m).$$

1.4.1 The Shadow Bound in Two Dimensions

As a warm-up before the proof sketch of Theorem 1.13, we look at the easier two-dimensional case. We bound the expected complexity of the convex hull of Gaussian perturbed points. The proof is much simpler than the shadow bound in higher dimensions but it contains many of the key insights we need.

First, we state a simple lemma. Proving this lemma is Exercise 1.4.

Lemma 1.17. *Let $X \in \mathbb{R}$ be a random variable with $\mathbb{E}[X] = \mu$ and $\mathrm{Var}(X) = \tau^2$. Then X satisfies*

$$\frac{\mathbb{E}\left[X^2\right]}{\mathbb{E}\left[|X|\right]} \geq (|\mu| + \tau)/2.$$

Theorem 1.18. *For points $a_1, \dots a_m \in \mathbb{R}^2$ independently Gaussian distributed, each with covariance matrix $\sigma^2 I_2$ and $\|\mathbb{E}[a_i]\| \leq 1$ for all $i \in [m]$, the convex hull $Q := \mathrm{conv}(a_1, \dots, a_m)$ has $O(\sigma^{-1} + \sqrt{\ln m})$ edges in expectation.*

Proof We will prove that, on average, the edges of Q are long and the perimeter of Q is small. This is sufficient to bound the expected number of edges.

For $i, j \in [m], i \neq j$, let $E_{i,j}$ denote the event that a_i and a_j are the end points of an edge of Q. By linearity of expectation we have the following equality:

$$\mathbb{E}[\text{perimeter}(Q)] = \sum_{1 \leq i < j \leq m} \mathbb{E}[\|a_i - a_j\| \mid E_{i,j}] \Pr[E_{i,j}].$$

We lower bound the right-hand side by taking the minimum over all conditional expectations and get

$$\sum_{1 \leq i < j \leq m} \mathbb{E}[\|a_i - a_j\| \mid E_{i,j}] \Pr[E_{i,j}] \geq \min_{k \neq l} \mathbb{E}[\|a_k - a_l\| \mid E_{k,l}] \sum_{1 \leq i < j \leq m} \Pr[E_{i,j}].$$

Dividing on both sides, we can estimate the expected number of edges

$$\mathbb{E}[|\text{edges}(Q)|] = \sum_{1 \leq i < j \leq m} \Pr[E_{i,j}] \leq \frac{\mathbb{E}[\text{perimeter}(Q)]}{\min_{k \neq l} \mathbb{E}[\|a_k - a_l\| \mid E_{k,l}]}. \tag{1.4}$$

We are left to bound the numerator and denominator on the right-hand side. For the first, we observe that Q is convex and thus has perimeter at most that of any containing disc. This yields the bound

$$\mathbb{E}[\text{perimeter}(Q)] \leq \mathbb{E}[2\pi \max_i \|a_i\|] \leq 2\pi(1 + 6\sigma\sqrt{\ln m}), \tag{1.5}$$

using standard Gaussian tail bounds.

We are left to lower bound the denominator. Fix $k = 1, l = 2$ without loss of generality. The quantity of interest is

$$\mathbb{E}[\|a_1 - a_2\| \mid E_{1,2}] = \frac{\int_{\mathbb{R}^2} \int_{\mathbb{R}^2} \|a_1 - a_2\| \Pr[E_{1,2}] \mu_1(a_1)\mu_2(a_2) \mathrm{d}a_1 \mathrm{d}a_2}{\int_{\mathbb{R}^2} \int_{\mathbb{R}^2} \Pr[E_{1,2}] \mu_1(a_1)\mu_2(a_2) \mathrm{d}a_1 \mathrm{d}a_2}$$

where μ_i is the probability density of a_i and the probability of $E_{1,2} := E_{1,2}(a_1, \ldots, a_n)$ is taken over the randomness in $a_3, a_4, \ldots, a_m$. To get control on the event $E_{1,2}$, we perform a change of coordinates from $a_1, a_2 \in \mathbb{R}^2$ to $t \in [0, \infty], \theta \in \mathbb{S}^1, h_1, h_2 \in \mathbb{R}$ satisfying

$$a_1 = t\theta + R_\theta(h_1)$$
$$a_2 = t\theta + R_\theta(h_2)$$

where $R_\theta : \mathbb{R} \to \theta^\perp$ is the isometric linear embedding of $\mathbb{R}$ into the linear subspace orthogonal to θ with $R_\theta(1)$ having positive first coordinate. This transformation is uniquely defined and continuous whenever a_1 and a_2 are linearly independent and θ has non-zero first coordinate, which happens with probability 1. The Jacobian of this transformation is $|h_1 - h_2|$ and we can rewrite the above fraction as

$$\frac{\int_0^\infty \int_{\mathbb{S}^1} \int_{-\infty}^\infty \int_{-\infty}^\infty |h_1 - h_2|^2 \Pr[E_{1,2}] \mu_1(t\theta + R_\theta(h_1))\mu_2(t\theta + R_\theta(h_2)) \mathrm{d}h_1 \mathrm{d}h_2 \mathrm{d}\theta \mathrm{d}t}{\int_0^\infty \int_{\mathbb{S}^1} \int_{-\infty}^\infty \int_{-\infty}^\infty |h_1 - h_2| \Pr[E_{1,2}] \mu_1(t\theta + R_\theta(h_1))\mu_2(t\theta + R_\theta(h_2)) \mathrm{d}h_1 \mathrm{d}h_2 \mathrm{d}\theta \mathrm{d}t}.$$

The event $E_{1,2}$ is equivalent to asking that either $\theta^\mathsf{T} a_i \leq t$ for all $i = 3, 4, \ldots, m$ or $\theta^\mathsf{T} a_i \geq t$ for all $i = 3, 4, \ldots, m$. This makes $E_{1,2}$ a function of only $a_3, \ldots, a_n$ and θ and t, i.e. its value does not depend on h_1, h_2.

Now, we use that $\frac{\int g(p)h(p)\mathrm{d}p}{\int g(p)\mathrm{d}p} \geq \inf_p h(p)$ for any positive integrable g, h and find

$$\begin{aligned}\mathbb{E}[\|a_1 - a_2\| \mid E_{1,2}] &\geq \inf_{t,\theta} \frac{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |h_1 - h_2|^2 \mu_1(t\theta + R_\theta(h_1))\mu_2(t\theta + R_\theta(h_2))\mathrm{d}h_1\mathrm{d}h_2}{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |h_1 - h_2| \mu_1(t\theta + R_\theta(h_1))\mu_2(t\theta + R_\theta(h_2))\mathrm{d}h_1\mathrm{d}h_2} \\ &= \inf_{t,\theta} \frac{\int_{-\infty}^{\infty} z^2 \left(\int_{-\infty}^{\infty} \mu_1(R_\theta(h_1))\mu_2(R_\theta(h_1 - z))\mathrm{d}h_1\right)\mathrm{d}z}{\int_{-\infty}^{\infty} |z| \left(\int_{-\infty}^{\infty} \mu_1(R_\theta(h_1))\mu_2(R_\theta(h_1 - z))\mathrm{d}h_1\right)\mathrm{d}z},\end{aligned}$$

substituting $z = h_1 - h_2$ and simplifying. For fixed t, θ, we can reinterpret the last fraction as $\mathbb{E}[Z^2]/\mathbb{E}[|Z|]$ for Z a random variable with probability density proportional to

$$\int_{-\infty}^{\infty} \mu_1(R_\theta(h_1))\mu_2(R_\theta(h_1 - z))\mathrm{d}h_1.$$

This is the same probability density as that of the difference of two independent Gaussian random variables each of variance σ^2, which means that Z has variance $2\sigma^2$. If we apply Lemma 1.17 to Z, we deduce $\mathbb{E}[\|a_1 - a_2\| \mid E_{1,2}] \geq \sigma/\sqrt{2}$. We conclude that the expected total number of edges is bounded from above by

$$\mathbb{E}[\text{edges}(Q)] \leq 2\pi \frac{1 + 6\sigma\sqrt{\ln m}}{\sigma/\sqrt{2}} \leq 9\sigma^{-1} + 54\sqrt{\ln m}.$$

□

1.4.2 The Shadow Bound in Higher Dimensions

In this section we sketch the proof of Theorem 1.13. For the remainder of this section, let $a_1, \ldots, a_m \in \mathbb{R}^n$ be independent variance σ^2 Gaussian random vectors, $Q := \mathrm{conv}(a_1, \ldots, a_m)$ and $W \subseteq \mathbb{R}^n$ be a fixed 2D plane.

Our task is to bound $\mathbb{E}[|\text{edges}(Q \cap W)|]$. The strategy will be the same as in Theorem 1.18, namely to relate the perimeter and expected minimum edge length. A first observation is that an edge of $Q \cap W$ w.p. 1 takes the form $\mathrm{conv}(a_i : i \in B) \cap W$, where $B \subseteq [m]$, $|B| = n$, and $\mathrm{conv}(a_i : i \in B)$ is a facet of Q (see Figure 1.2). From here, an identical argument as for (1.4) yields the following edge counting lemma.

Lemma 1.19. *For a basis $B \subseteq [m]$, $|B| = n$, let E_B denote the event that* $\mathrm{conv}(a_i : i \in B) \cap W$ *is an edge of $Q \cap W$. Then, the following bound holds:*

$$\mathbb{E}[|\text{edges}(Q \cap W)|] \leq \frac{\mathbb{E}[\text{perimeter}(Q \cap W)]}{\min_{B \subseteq [m], |B| = n} \mathbb{E}[\text{length}(\mathrm{conv}(a_i : i \in B) \cap W) \mid E_B]}.$$

The numerator in Lemma 1.19 can be bounded along the same lines as in Theorem 1.18.

Lemma 1.20. $\mathbb{E}[\text{perimeter}(Q \cap W)] \leq \mathbb{E}[\text{perimeter}(\pi_W(Q))] \leq O(1 + \sigma\sqrt{\ln m})$.

We now restrict our attention to lower bounding $\mathbb{E}[\text{length}(\text{conv}(a_i : i \in B) \cap W) \mid E_B]$ for a fixed basis $B \subseteq [m]$, where w.l.o.g. we may assume that $B = \{1, \dots, n\}$.

Just like we did in the proof of Theorem 1.18, we perform a change of variables. The first part of the new parametrization of $a_1, \dots, a_n$ consists of their containing affine subspace H, described by $\theta \in \mathbb{S}^{n-1}, t \geq 0$ satisfying

$$\text{aff}(a_1, \dots, a_n) =: H = \{x \in \mathbb{R}^n : \theta^\mathsf{T} x = t \quad \text{for all } i \in [n]\}.$$

This is depicted in Figure 1.4, with $\text{conv}(a_i : i \in B) \cap W$ marked by the line segment K.

To describe the location of the points inside the hyperplane H, we use a family of orthonormal embeddings $R := R_\theta : \mathbb{R}^{n-1} \to \theta^\perp$, where the points $b_1, \dots, b_n$ satisfy $t\theta + R_\theta(b_i) = a_i$, $\forall i \in [n]$. A simple choice for R_θ is $R_\theta(b) := (b, 0) - (e_n + \theta)(\theta^\mathsf{T}(b, 0))/(1 + \theta_n)$, which first sends $b \to (b, 0) \in (e_n)^\perp$ and composes it with the rotation which sends e_n to θ and fixes $\text{span}(e_n, \theta)^\perp$. The properties of this change of variables are given below.

Theorem 1.21. *The change of variables is well-defined with probability* 1 *and has Jacobian* $(n-1)!\text{vol}(\text{conv}(b_1, \dots, b_n))$. *If we fix* θ, t *then the induced probability density function of* $b_1, \dots, b_n$ *is proportional to* $\text{vol}(\text{conv}(b_1, \dots, b_n)) \prod_{i=1}^n \mu_i(Rb_i)$, *where* μ_i *is the probability density function of* a_i *for each* $i \in [n]$.

Define the line $\ell \subset \mathbb{R}^{n-1}$ to satisfy $H \cap W = t\theta + R\ell$. In this notation we get $\text{conv}(a_1, \dots, a_n) \cap W = t\theta + R(\text{conv}(b_1, \dots, b_n) \cap \ell)$. The event E_B holds when $\theta^\mathsf{T} a_i > t$ for all $i = n+1, \dots, m$ or $\theta^\mathsf{T} a_i < t$ for all $i = n+1, \dots, m$ (i.e., $\text{conv}(a_1, \dots, a_n)$ is a facet of Q), which we denote by $E_{B,f}$, and $\text{conv}(b_i : i \in B) \cap \ell$ has positive length, which we denote by $E_{B,l}$. Just like in the two-dimensional case, after conditioning on θ, t, the events $E_{B,f}$ and $E_{B,l}$ become independent. In particular, after this conditioning, $E_{B,l}$ only depends on $b_1, \dots, b_n$ and $E_{B,f}$ is independent of $b_1, \dots, b_n$.

Given this independence, we may restrict our attention to proving a lower bound on $\mathbb{E}[\text{length}(\text{conv}(b_1, \dots, b_n) \cap \ell) \mid E_{B,l}]$, where $b_1, \dots, b_n$ are conditioned on a fixed θ and t. To analyze the expected edge length, we will need the following concepts.

Definition 1.22. Let $\omega \in \mathbb{R}^n$, $\|\omega\|_2 = 1$ and $p \in \omega^\perp$ such that $\ell = p + \mathbb{R}\omega$ and let $\mathcal{L} := \text{conv}(b_i : i \in B) \cap \ell$. For any $q \in \omega^\perp$, define the set of convex combinations

$$C(q) := \{\lambda \in \mathbb{R}^n_+ : \sum_{i=1}^n \lambda_i = 1, \sum_{i=1}^n \lambda_i \pi_\omega^\perp(b_i) = q\},$$

whose ℓ_1 diameter we denote by $\|C(q)\|_1$, which is 0 by convention if $C(q) = \emptyset$. Let $\gamma := \|C(p)\|_1$. Define $z \in \mathbb{R}^n$ to be the unique up to sign solution to $\sum_{i=1}^n z_i \pi_{\omega^\perp}(b_i) = 0$ with $\|z\|_1 = 1$ (uniqueness holds w.p. 1).

Some preliminary remarks on the above definitions. ω is the direction of the

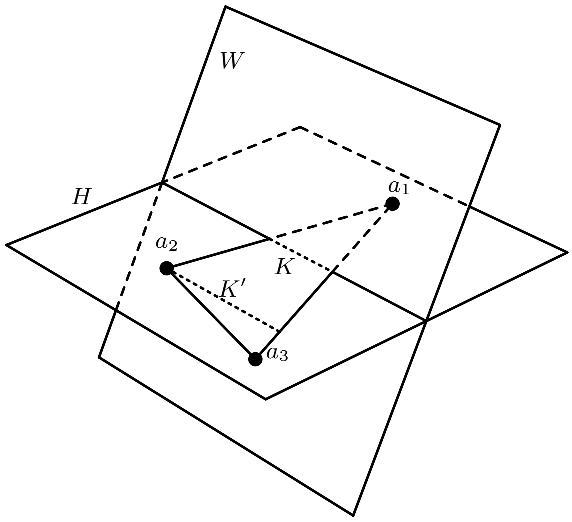

Figure 1.4 The vectors $a_1, \ldots, a_n$ are conditioned for $\mathrm{conv}(a_1, \ldots, a_n)$ to intersect W and lie in H. The short dotted line segment $K = W \cap H \cap \mathrm{conv}(a_1, a_2, a_3)$ is the edge of $Q \cap W$ induced by the basis and the longer dotted line segment K' is the longest chord of the simplex parallel to the line $H \cap W$. We aim to lower bound the expected length of the line segment K.

line ℓ and $\{p\} = \pi_{\omega^\perp}(\ell)$ is its intersection with $\omega^\perp$. $\mathcal{L}$ is the tentative edge whose expected length we wish to lower bound. The set $C(q)$, $q \in \omega^\perp$, is a line segment in the direction of z, noting that the difference of any two points in $C(q)$ must be a multiple of z. In particular, if $C(p) \neq \emptyset$, one may express $C(p) = [\lambda_0, \lambda_0 + \gamma z]$, for some convex combination λ_0, where $\gamma := \|C(p)\|_1$ as above. One may equivalently define

$$C(p) = \{\lambda \in \mathbb{R}^n_+ : \sum_{i=1}^n \lambda_i = 1, \sum_{i=1}^n \lambda_i b_i \in \mathcal{L}\},$$

that is, $C(p)$ is the set of convex combination representing the edge $\mathcal{L}$. It is now direct to see that $\mathcal{L}$ has positive length iff $\gamma > 0$, that is, $E_{B,l}$ is equivalent to $\gamma > 0$.

The following lemma, whose proof is Exercise 1.6, encapsulates the properties of $C(q)$ that we will need.

Lemma 1.23. *Let* $y := \sum_{i=1}^n |z_i| \pi_{\omega^\perp}(b_i)$ *and* $h_1 = \omega^\mathsf{T} b_1, \ldots, h_n = \omega^\mathsf{T} b_n$. *Then the following hold:*

1. $\|C(q)\|_1$ *is a non-negative concave function of* $q \in \mathrm{conv}(\pi_{\omega^\perp}(b_i) : i \in [n])$.
2. $\max_{q \in \mathrm{conv}(\pi_{\omega^\perp}(b_i) : i \in [n])} \|C(q)\|_1 = \|C(y)\|_1 = 2$.
3. $\mathrm{length}(\mathcal{L}) = \gamma |\sum_{i=1}^n z_i h_i|$.

The factors on the right-hand side in the last item of Lemma 1.23 have identifiable meanings. The sum $2|\sum_{i=1}^{n} z_i h_i|$ is the length of the longest chord of $\mathrm{conv}(b_1, \ldots, b_n)$ parallel to ℓ. In Figure 1.4, this longest chord is represented by the line segment K'. It is the analogue of $h_1 - h_2$ from the two-dimensional case. The remaining term, $\gamma/2$, is the ratio of the length of the edge L to the length of the longest chord. In Figure 1.4 this is the ratio of the length of the line segment K to the length of the line segment K'. We note that this term has no analogue in 2 dimensions and so lower bounding it will require new ideas. We can now lower bound the expected length of $\mathcal{L}$ as follows:

$$\mathbb{E}[\gamma|\sum_{i=1}^{n} z_i h_i| \mid \gamma > 0] \geq \mathbb{E}[\gamma \mid \gamma > 0] \inf_{\pi_{\omega^\perp}(b_i):i\in[n]} \mathbb{E}[|\sum_{i=1}^{n} z_i h_i| \mid \pi_{\omega^\perp}(b_i) : i \in [n]], \tag{1.6}$$

noting that $(\pi_{\omega^\perp}(b_i) : i \in [n])$ determine z and γ.

We first lower bound the latter term, the expected maximum chord length, for which we will need the induced probability density on $h_1, \ldots, h_n$. This is given by the following lemma, whose proof is a straightforward manipulation of the Jacobian in Theorem 1.21.

Lemma 1.24. *For any fixed values of the projections $\pi_{\omega^\perp}(b_1), \ldots, \pi_{\omega^\perp}(b_n)$, the inner products $h_1, \ldots, h_n$ have joint probability density proportional to*

$$|\sum_{i=1}^{n} z_i h_i| \prod_{i=1}^{n} \mu_i(R(h_i\omega)).$$

Using Lemma 1.24 and an analoguous argument to that in Theorem 1.18, we can express $\mathbb{E}[|\sum_{i=1}^{n} z_i h_i| \mid \pi_{\omega^\perp}(b_i) : i \in [n]]$ as the ratio $\mathbb{E}[(\sum_{i=1}^{n} z_i x_i)^2]/\mathbb{E}[|\sum_{i=1}^{n} z_i x_i|]$, where $x_1, \ldots, x_n$ are independent and each x_i is distributed according to $\mu_i(R(x_i\omega))$. Since $\sum_{i=1}^{n} z_i x_i$ has variance $\sigma^2\|z\|_2^2 \geq \sigma^2\|z\|_1^2/n = \sigma^2/n$, we may apply Lemma 1.17 to deduce the following lower bound.

Lemma 1.25. *Fixing $\pi_{\omega^\perp}(b_1), \ldots, \pi_{\omega^\perp}(b_n)$, we have $\mathbb{E}[|\sum_{i=1}^{n} z_i h_i|] \geq \sigma/(2\sqrt{n})$.*

The remaining task is to lower bound $\mathbb{E}[\gamma \mid \gamma > 0]$. This will require a number of new ideas and some simplifying assumptions, which we sketch below.

The main intuitive observation is that $\gamma > 0$ is small essentially only when $p \in \mathrm{conv}(\pi_{\omega^\perp}(b_i) : i \in [n])$ is close to the boundary of the convex hull. To show that this does not happen on average, the main idea will be to show that for any configuration $\pi_{\omega^\perp}(b_1), \ldots, \pi_{\omega^\perp}(b_n)$ for which γ is tiny, there is a nearly-equiprobable one for which γ is lower bounded a function of n, m and σ. Here the move to the improved configuration will correspond to pushing the "center" y of $\mathrm{conv}(\pi_{\omega^\perp}(b_i) : i \in [n])$ towards p, where y is as in Lemma 1.23.

To be able to argue near-equiprobability, we will make the simplifying assumption that the original densities $\mu_1, \ldots, \mu_m$ are *L-log-Lipschitz*, for $L = \Theta(\sqrt{n \ln m}/\sigma)$, where we recall that $f : \mathbb{R}^n \to \mathbb{R}_+$ is L-log-Lipschitz if $f(x) \leq f(y)e^{L\|x-y\|}$, $\forall x, y$.

While a variance σ^2 Gaussian is not *globally* log-Lipschitz, it can be checked that is L-log-Lipschitz within distance $\sigma^2 L$ of its mean. By standard Gaussian tail bounds the probability that any a_i is at distance $\sigma^2 L = \Omega(\sigma\sqrt{n \ln m})$ from its mean is at most $m^{-\Omega(n)}$. Since an event occurring w.p. less than $\binom{m}{n}^{-1}$ contributes at most 1 to $E[|\text{edges}(Q \cap W)|]$, noting that $\binom{m}{n}$ is a deterministic upper bound, it is intuitive that we can assume L-log-Lipschitzness "wherever it matters", though a rigorous proof of this is beyond the scope of this chapter.

Using log-Lipschitzness, we will only be able to argue that close-by configurations are equiprobable. For this to make a noticeable impact on γ, we will need $\pi_{\omega^\perp}(b_1), \ldots, \pi_{\omega^\perp}(b_n)$ to not be too far apart to begin with. For this purpose, we let E_D denote the event that $\max_{i,j} \|\pi_{\omega^\perp}(b_i) - \pi_{\omega^\perp}(b_j)\| \leq D$, for $D = \Theta(1 + \sigma\sqrt{n \ln m})$. It is useful to note that the original $a_1, \ldots, a_m$, which are farther apart, already satisfy this distance requirement w.p. $1 - m^{-\Omega(n)}$ using similar tail bound arguments as above.

With these concepts, we will be able to lower bound $\mathbb{E}[\gamma \mid \gamma > 0, E_D]$ in Lemma 1.26 below. For this to be useful, we would like

$$\mathbb{E}[\gamma \mid \gamma > 0] \geq \mathbb{E}[\gamma \mid \gamma > 0, E_D]/2. \tag{1.7}$$

While this may not be true in general, the main reason it can fail is if the starting basis B has probability less than $m^{-\Omega(n)}$ of forming an edge to begin with, in which case it can be safely ignored anyway. We henceforth assume inequality (1.7).

Lemma 1.26. *Letting $L = \Theta(\sqrt{n \ln m}/\sigma), D = \Theta(1 + \sigma\sqrt{n \ln m})$ be as above, we have that $\mathbb{E}[\gamma \mid \gamma > 0, E_D] \geq \Omega(\frac{1}{nDL})$.*

Proof sketch Let us start by fixing $s_i := \pi_{\omega^\perp}(b_i) - \pi_{\omega^\perp}(b_1)$ for all $2 \leq i \leq n$, for which the condition $\|s_i\| \leq D$, $\|s_i - s_j\| \leq D$, for all $i, j \in \{2, \ldots, n\}$ holds. Note that this condition is equivalent to E_D. Let $S = \text{conv}(0, s_2, \ldots, s_n)$ denote the resulting shape of the projected convex hull. Let us now additionally fix $h_1, \ldots, h_n$ arbitrarily.

At this point, the only degree of freedom left is in the position of $\pi_{\omega^\perp}(b_1)$. The condition $\gamma > 0$ is now equivalent to $p \in \pi_{\omega^\perp}(b_1) + S \Leftrightarrow \pi_{\omega^\perp}(b_1) \in p - S$. From here, the conditional density μ of $\pi_{\omega^\perp}(b_1)$ satisfies

$$\mu \propto \mu_1(R(\pi_{\omega^\perp}(b_1))) \prod_{i=2}^{n} \mu_i(R(\pi_{\omega^\perp}(b_1) + s_i)),$$

where we note that fixing $h_1, \ldots, h_n, s_2, \ldots, s_n$ makes the Jacobian in Theorem 1.21 constant.

As we mentioned above, we assume that $\mu_1, \ldots, \mu_n$ are L-log-Lipschitz everywhere. This makes μ be nL-log-Lipschitz. Since $p - S$ has diameter at most D and γ is a concave function of $\pi_{\omega^\perp}(b_1)$ with maximum 2 by Lemma 1.23, we can use Lemma 1.27 below to finish the sketch. $\square$

The final lemma is Exercise 1.7.

Lemma 1.27. *For a random variable $x \in S \subset \mathbb{R}^n$ having L-log-Lipschitz density supported on a convex set S of diameter D and $f : S \to \mathbb{R}_+$ concave, one has*

$$\mathbb{E}[f(x)] \geq e^{-2} \frac{\max_{y \in S} f(y)}{\max(DL, n)}.$$

Putting together Lemma 1.19, 1.20, 1.25, 1.26 and inequality 1.7, we get the desired result

$$\begin{aligned}\mathbb{E}[|\mathrm{edges}(Q \cap W)|] &\leq \frac{O(1 + \sigma\sqrt{\ln m})}{\frac{\sigma}{2\sqrt{n}} \cdot \Omega(\frac{1}{nDL})} \\ &= O(n^2 \sigma^{-2} \sqrt{\ln m}(1 + \sigma\sqrt{n \ln m})(1 + \sigma\sqrt{\ln m})).\end{aligned}$$

1.5 Discussion

We saw smoothed complexity results for linear programming in two different perturbation models. In the first model, the feasible region was highly structured and "well-conditioned", namely a flow polytope, and only the objective was perturbed. In the second model, the feasible region was a general linear program whose constraint data was perturbed by Gaussians.

While the latter model is the more general, the LP's it generates differ from real-world LP's in many ways. Real-world LP's are often highly degenerate, due to the combinatorial nature of many practical problems, and sparse, typically only one percent of the constraint matrix entries are non-zero. The Gaussian constraint perturbation model has neither of these properties. Second, it is folklore that the number of pivot steps it takes to solve an LP is roughly linear in m or n. At least from the perspective of the shadow vertex simplex method, this provably does not hold for the Gaussian constraint perturbation model. Indeed, Borgwardt (1987) proved that as $m \to \infty$ and n is fixed, the shadow bound for Gaussian unit LPs (where the means are all 0) is $\Theta(n^{1.5}\sqrt{\ln m})$.

There are plenty of concrete open problems in this area. The shadow bound of Theorem 1.13 is likely to be improvable, as it does not match the known $\Theta(n^{1.5}\sqrt{\ln m})$ bound for Gaussian unit LPs mentioned above. Already in two dimensions, the correct bound could be much smaller, as discussed in (Devillers et al., 2016). In the i.i.d. Gaussian case, the edge counting strategy in Lemma 1.19 is exact, but our lower bound on the expected edge length is much smaller than the true value. In the smoothed case, the edge counting strategy seems too lossy already when $n = 2$.

The proof of Theorem 1.13 also works for any log-Lipschitz probability distribution with sufficiently strong tail bounds. However, nothing is known for distributions with bounded support or distributions that preserve some meaningful structure of the LP, such as most zeroes in the constraint matrix. One difficulty in extending the current proof lies in it considering even very unlikely hyperplanes for the basis vectors to lie in.

In practice the shadow vertex pivot rule is outperformed by the commonly used most-negative reduced cost rule, steepest edge rule, and Devex rule. However, there are currently no theoretical explanations for why these rules would perform well. The analyses discussed here do not extend to such pivot rules, due to making heavy use of the local characterization of whether a given vertex is visited by the algorithm.

We note that a major reason for the popularity of the simplex method is its unparalleled effectiveness at solving sequences of related LPs, where after each solve a column or row may be added or deleted from the current program. In this context, the simplex method is easy to “warm start” from either the primal or dual side, and typically only a few additional pivots solve the new LP. This scenario occurs naturally in the context of integer programming, where one must solve many related LP relaxations within a branch and bound tree or during the iterations of a cutting plane method. Current theoretical analyses of the simplex method don’t say anything about this scenario.

1.6 Notes

The shadow vertex simplex method was first introduced by Gass and Saaty (1955) to solve bi-objective linear programming problems and is also known as the parametric simplex algorithm.

Families of LPs on which the shadow vertex simplex method takes an exponential number of steps were constructed by Murty (1980); Goldfarb (1983, 1994); Amenta and Ziegler (1998); Gärtner et al. (2013). One such construction is the subject of Exercise 1.1. A very interesting construction was given by Disser and Skutella (2018), who gave a flow network on which it is NP-complete to decide whether the SSP algorithm will ever use a given edge. Hence, the shadow vertex simplex algorithm implicitly spends its exponential running time to solve hard problems.

The first probabilistic analysis of the simplex method is due Borgwardt, see (Borgwardt, 1987), who studied the complexity of solving $\max c^\mathsf{T} x, Ax \leq 1$ when the rows of A are sampled from a spherically symmetric distribution. He proved a tight shadow bound of $\Theta(n^2 m^{1/(n-1)})$, which is valid for any such distribution, as well as the tight limit for the Gaussian distribution mentioned earlier. Both of these bounds can be made algorithmic, losing a factor n, using Borgwardt’s DD algorithm.

The smoothed analysis of the SSP algorithm is due to Brunsch et al. (2015). They also proved that the running time bound holds for the SSP algorithm as applied to the minimum-cost flow problem, and they showed a nearly matching lower bound.

The first smoothed analysis of the simplex method was by Spielman and Teng

(2004), who introduced the concept of smoothed analysis and the perturbation model of section 1.4. They achieved a bound of $O(n^{55}m^{86}\sigma^{-30} + n^{70}m^{86})$. This bound was subsequently improved by Deshpande and Spielman (2005); Vershynin (2009); Schnalzger (2014); Dadush and Huiberts (2018).

In this chapter, we used the DD algorithm for the Phase I unit LP, traversing $n-1$ shadow paths. Another algorithm for solving (Phase I Unit LP), which traverses an expected $O(1)$ shadow paths, can bring the smoothed complexity bound down to $O(n^2\sigma^{-2}\sqrt{\ln m} + n^3 \ln^{3/2} m)$. This procedure, which is a variant of an algorithm of Vershynin (2009), as well as a rigorous proof of Theorem 1.13, can be found in Dadush and Huiberts (2018).

The two-dimensional convex hull complexity of Gaussian perturbed points from Theorem 1.18 was studied before by Damerow and Sohler (2004); Schnalzger (2014); Devillers et al. (2016). The best general bound among them is $O(\sqrt{\ln n} + \sigma^{-1}\sqrt{\ln n})$, asymptotically slightly worse than the bound in Theorem 1.18.

The DD algorithm was first used for smoothed analysis by Schnalzger (2014). The edge counting strategy based on the perimeter and minimum edge length is due to Kelner and Spielman (2006). They proved that an algorithm based on the shadow vertex simplex method can solve linear programs in weakly polynomial time. The two-phase interpolation method used here was first introduced and analyzed in the context of smoothed analysis by Vershynin (2009). The coordinate transformation in Theorem 1.21 is called a Blaschke-Petkantschin identity. It is a standard tool in the study of random convex hulls.

The number of pivot steps in practice is surveyed by Shamir (1987). More recent experiments such as (Makhorin, 2017) remain bounded by a small linear function of $n + m$, though a slightly super-linear function better fits the data according to Andrei (2004).

References

Amenta, Nina, and Ziegler, Günter M. 1998. Deformed Products and Maximal Shadows. *Contemporary Math.*, **223**, 57–90.

Andrei, Neculai. 2004. On the complexity of MINOS package for linear programming. *Studies in Informatics and Control*, **13**(1), 35–46.

Borgwardt, Karl-Heinz. 1977. *Untersuchungen zur Asymptotik der mittleren Schrittzahl von Simplexverfahren in der linearen Optimierung.* Ph.D. thesis, Universität Kaiserslautern.

Borgwardt, Karl-Heinz. 1987. *The simplex method: A probabilistic analysis.* Algorithms and Combinatorics: Study and Research Texts, vol. 1. Berlin: Springer-Verlag.

Brunsch, Tobias, Cornelissen, Kamiel, Manthey, Bodo, Röglin, Heiko, and Rösner, Clemens. 2015. Smoothed analysis of the successive shortest path algorithm. *SIAM J. Comput.*, **44**(6), 1798–1819. Preliminary version in SODA '13.

Dadush, Daniel, and Huiberts, Sophie. 2018. A friendly smoothed analysis of the simplex method. Pages 390–403 of: *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*. ACM.

Damerow, Valentina, and Sohler, Christian. 2004. Extreme points under random noise. Pages 264–274 of: *European Symposium on Algorithms*. Springer.

Deshpande, Amit, and Spielman, Daniel A. 2005. Improved Smoothed Analysis of the Shadow Vertex Simplex Method. Pages 349–356 of: *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*. FOCS '05.

Devillers, Olivier, Glisse, Marc, Goaoc, Xavier, and Thomasse, Rémy. 2016. Smoothed complexity of convex hulls by witnesses and collectors. *Journal of Computational Geometry*, **7**(2), 101–144.

Disser, Yann, and Skutella, Martin. 2018. The simplex algorithm is NP-mighty. *ACM Transactions on Algorithms (TALG)*, **15**(1), 5.

Gärtner, Bernd, Helbling, Christian, Ota, Yoshiki, and Takahashi, Takeru. 2013. Large shadows from sparse inequalities. *arXiv preprint arXiv:1308.2495*.

Gass, Saul, and Saaty, Thomas. 1955. The computational algorithm for the parametric objective function. *Naval Res. Logist. Quart.*, **2**, 39–45.

Goldfarb, Donald. 1983. *Worst case complexity of the shadow vertex simplex algorithm*. Tech. rept. Columbia University, New York.

Goldfarb, Donald. 1994. *On the Complexity of the Simplex Method*. Dordrecht: Springer Netherlands. Pages 25–38.

Kelner, Jonathan A., and Spielman, Daniel A. 2006. A randomized polynomial-time simplex algorithm for linear programming. Pages 51–60 of: *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*. STOC '06. ACM, New York.

Makhorin, Andrew. 2017. GLPK (GNU Linear Programming Kit) documentation.

Matousek, Jiri, and Gärtner, Bernd. 2007. *Understanding and using linear programming*. Springer Science & Business Media.

Murty, Katta G. 1980. Computational complexity of parametric linear programming. *Math. Programming*, **19**(2), 213–219.

Schnalzger, Emanuel. 2014. *Lineare Optimierung mit dem Schatteneckenalgorithmus im Kontext probabilistischer Analysen*. Ph.D. thesis, Universität Augsburg. Original in German. English translation by K.H. Borgwardt available at `https://www.math.uni-augsburg.de/prof/opt/mitarbeiter/Ehemalige/borgwardt/Downloads/Abschlussarbeiten/Doc_Habil.pdf`.

Shamir, Ron. 1987. The efficiency of the simplex method: a survey. *Management Sci.*, **33**(3), 301–334.

Spielman, Daniel A., and Teng, Shang-Hua. 2004. Smoothed analysis of algorithms: why the simplex algorithm usually takes polynomial time. *J. ACM*, **51**(3), 385–463 (electronic).

Vershynin, Roman. 2009. Beyond Hirsch conjecture: walks on random polytopes and smoothed complexity of the simplex method. *SIAM J. Comput.*, **39**(2), 646–678. Preliminary version in FOCS '06.

Zadeh, Norman. 1973. A bad network problem for the simplex method and other minimum cost flow algorithms. *Math. Programming*, **5**, 255–266.

Exercises

1.1 In this exercise we show that the projection of an LP can have 2^n vertices on instances with n variables and $2n$ constraints. The Goldfarb cube in dimension n is the LP

$$\begin{aligned}
&\max x_n \\
&0 \leq x_1 \leq 1 \\
&\alpha x_1 \leq x_2 \leq 1 - \alpha x_1 \\
&\alpha(x_{k-1} - \beta x_{k-2}) \leq x_k \leq 1 - \alpha(x_{k-1} - \beta x_{k-2}) \text{ for } 3 \leq k \leq n
\end{aligned}$$

where $\alpha < 1/2$ and $\beta < \alpha/4$.

(a) Prove that the LP has 2^n vertices.

(b) Prove that every vertex is optimal for some range of linear combinations $\alpha e_{n-1} + \beta e_n$. Hint: a vertex maximizes an objective if that objective can be written as a non-negative linear combination of the constraint vectors of tight constraints.

(c) Show that it follows that the shadow vertex simplex method has worst-case running time exponential in n.

(d) Can you adapt the instance such that the expected shadow vertex count remains exponential when the shadow plane is randomly perturbed?

(e) Define *zero-preserving* perturbations to perturb only the non-zero entries of the constraint matrix. Do the worst-case instances still have shadows with exponentially many vertices after applying Gaussian zero-preserving perturbations of variance $O(1)$?

1.2 Prove Lemma 1.10. Specifically, show that if a basis $B \subset [m]$ induces the optimal vertex of P for some objective c, then B induces a facet of Q intersecting the ray $c\mathbb{R}_{++}$. Then, prove that this fact implies the lemma.

1.3 Prove Lemma 1.11.

1.4 Prove Lemma 1.17.

1.5 Verify that the Jacobian of the coordinate transformation in Theorem 1.18 is $|h_1 - h_2|$.

1.6 Prove Lemma 1.23.

1.7 Prove Lemma 1.27. Hint: let $y = \operatorname{argmax}_{y \in S} f(y)$ and define $S' := y + \alpha(S - y)$. Prove that $\Pr[x \in S'] \geq e^{-2}$ for $\alpha = 1 - \frac{1}{\max(DL, n)}$, and that $f(x) \geq (1-\alpha) f(y)$ for all $x \in S'$.