

The Minimum Euclidean-Norm Point in a Convex Polytope: Wolfe’s Combinatorial Algorithm Is Exponential

Jesús A. De Loera
Department of Mathematics
University of California, Davis
USA
deloera@math.ucdavis.edu

Jamie Haddock
Department of Mathematics
University of California, Davis
USA
jhaddock@math.ucdavis.edu

Luis Rademacher
Department of Mathematics
University of California, Davis
USA
lrademac@ucdavis.edu

ABSTRACT

The complexity of Philip Wolfe’s method for the minimum Euclidean-norm point problem over a convex polytope has remained unknown since he proposed the method in 1974. The method is important because it is used as a subroutine for one of the most practical algorithms for submodular function minimization. We present the first example that Wolfe’s method takes exponential time. Additionally, we improve previous results to show that linear programming reduces in strongly-polynomial time to the minimum norm point problem over a simplex.

CCS CONCEPTS

• **Mathematics of computing** → **Quadratic programming**; • **Theory of computation** → **Linear programming**; **Quadratic programming**;

KEYWORDS

convex quadratic optimization, Wolfe’s method, linear programming, strongly polynomial time algorithms, lower bounds

ACM Reference Format:

Jesús A. De Loera, Jamie Haddock, and Luis Rademacher. 2018. The Minimum Euclidean-Norm Point in a Convex Polytope: Wolfe’s Combinatorial Algorithm Is Exponential. In *Proceedings of 50th Annual ACM SIGACT Symposium on the Theory of Computing (STOC’18)*. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3188745.3188820>

1 INTRODUCTION

The fundamental algorithmic problem we consider here is: Given a convex polytope $P \subset \mathbb{R}^d$, to find the point $\mathbf{x} \in P$ of minimum Euclidean norm, i.e., the *closest point to the origin* or what we call its *minimum norm point* for short. We assume P is presented as the convex hull of finitely many points $\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_n$ (not necessarily in convex position). We wish to find

$$\begin{aligned} & \operatorname{argmin} && \|\mathbf{x}\|_2 \\ & \text{subject to} && \mathbf{x} = \sum_{k=1}^n \lambda_k \mathbf{p}_k, \\ & && \sum_{k=1}^n \lambda_k = 1, \lambda_k \geq 0, \text{ for } k = 1, \dots, n. \end{aligned}$$

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

STOC’18, June 25–29, 2018, Los Angeles, CA, USA

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-5559-9/18/06...\$15.00

<https://doi.org/10.1145/3188745.3188820>

Finding the minimum norm point in a polytope is a basic auxiliary step in several algorithms arising in many areas of optimization and machine learning; a subroutine for solving the minimum norm point problem can be used to compute the projection of an arbitrary point to a polytope (indeed, $\operatorname{argmin}_{\mathbf{x} \in P} \|\mathbf{x} - \mathbf{a}\|_2$ is the same as $\mathbf{a} + \operatorname{argmin}_{\mathbf{y} \in P - \mathbf{a}} \|\mathbf{y}\|_2$). The minimum norm problem additionally appears in combinatorial optimization, e.g., for the nearest point problem for transportation polytopes [Bachem and Korte 1980; Calvillo and Romero 2016] and as a vital subroutine in Barany and Onn’s approximation algorithm to solve the colorful linear programming problem [Barany and Onn 1997]. One of the most important reasons to study this problem is because the minimum norm problem can be used as a subroutine for submodular function minimization through projection onto the base polytope, as proposed by Fujishige [Fujishige 1980]. Submodular minimization is useful in machine learning, where applications such as large scale learning and vision require efficient and accurate solutions [Bach et al. 2013; Nagano et al. 2011]. The problem also appears in optimal loading of recursive neural networks [Chandru et al. 1995]. The Fujishige-Wolfe algorithm is currently considered an important practical algorithm in applications [Chakrabarty et al. 2014; Fujishige et al. 2006; Fujishige and Isotani 2011]. Furthermore, Fujishige et al. first observed that linear programs may be solved by solving the minimum norm point problem [Fujishige et al. 2006], so this simple geometric problem is also relevant to the theory of algorithmic complexity of linear optimization.

One may ask about the complexity of other closely related problems. First, it is worth remembering that L_p norm minimization over a polyhedron is NP-hard for $0 \leq p < 1$ (see [Ge et al. 2011] and the references therein), while for $p \geq 1$ the convexity of the norm allows for computation of an ϵ -approximate solution in time polynomial in $\log(1/\epsilon)$. When $p = 1, 2$, the solution to L_p norm minimization over a polyhedron given by rational data is rational (see [Schrijver 1998] for details), so this solution may be computed exactly in polynomial time. Meanwhile, one can prove that the seemingly similar problem of finding the closest *vertex* of a convex polytope given by inequalities is NP-hard. The reduction for hardness is to the *directed Hamiltonian path problem*: Given a directed graph $G = (V, A)$ and two distinct vertices $s, t \in V$, one aims to decide whether G contains a directed Hamiltonian path from s to t . It is well-known there is a polytope represented by inequalities which has some vertices corresponding to the characteristic vectors of directed paths joining s to t in G . See e.g., [Fukuda et al. 1997] for the details, including the explicit inequality description of this polytope. Finally, by a change of variable $y_i = 1 - x_i$, changing zeros to ones and vice versa, and an appropriate selection of the

upper bound on the value of x_i , the minimum Euclidean norm vertex becomes precisely the “longest path from s to t ”, solving the directed Hamiltonian path problem.

Since the Euclidean norm is a convex quadratic form, the minimum norm point problem is a special case of convex quadratic optimization problem. Indeed, it is well-known that a convex quadratic programming problem can be approximately solved in polynomial time; that is, some point \mathbf{y} within distance ε of the desired minimizing point \mathbf{x} may be found in polynomial time with respect to $\log \frac{1}{\varepsilon}$. This can be done either through several iterative (convergent) algorithms, such as the Ellipsoid method [Kozlov et al. 1980] and interior-point method techniques [Boyd and Vandenberghe 2004]. Each of these are methods whose complexity depends upon the desired accuracy. However, an approximate numerical solution is inconvenient when the application requires more information, e.g., if we require to know the face that contains the minimum norm point. Numeric methods that converge to a solution and require further rounding are not as convenient for this need.

In this paper, we focus on combinatorial algorithms that rely on the structure of the polytope. There are several reasons to study the complexity of combinatorial algorithms for the minimum norm problem. On the one hand, the minimum norm problem can indeed be solved in strongly-polynomial time for some polytopes; most notably in network-flow and transportation polytopes (see [Bachem and Korte 1980; Calvillo and Romero 2016; Végé 2016], and references therein, for details). On the other hand, while linear programming reduces to the minimum norm problem, it is unknown whether linear programming can be solved in strongly-polynomial time [Smale 2000], thus the complexity of the minimum norm point problem could also impact the algorithmic efficiency of linear programming and optimization in general. For all these reasons it is natural to ask whether a strongly-polynomial time algorithm exists for the minimum norm problem for general polytopes.

Our contributions:

- In 1974, Philip Wolfe proposed a combinatorial method that can solve the minimum-norm point problem exactly [Wolfe 1974, 1976]. Since then, the complexity of Wolfe’s method was not understood. In Section 1 we present our main contribution and give the first example that Wolfe’s method has exponential behavior. This is akin to the well-known Klee-Minty examples showing exponential behavior for the simplex method [Klee and Minty 1972].
- As we mentioned earlier, an enticing reason to explore the complexity of the minimum norm problem is its intimate link to the complexity of linear programming. It is known that linear programming can be polynomially reduced to the minimum norm point problem [Fujishige et al. 2006]. In Section 2, we strengthen earlier results showing that linear optimization is strongly-polynomial time reducible to the minimum norm point problem on a simplex.

Proofs omitted from this extended abstract can be found in the full version [De Loera et al. 2017].

2 WOLFE’S METHOD EXHIBITS EXPONENTIAL BEHAVIOR

For convenience of the reader and to set up notation we start with a brief description of Wolfe’s method; however, for efficiency of presentation, we refer the reader to [Schrijver 1998] for relevant definitions and preliminary results in convex analysis. We will then describe our exponential example in detail, proving the exponential behavior of Wolfe’s method. First, we review two important definitions. Given a set of points $S \subseteq \mathbb{R}^d$, we have two minimum-norm points to consider. One is the *affine minimizer* which is the point of minimum norm in the affine hull of S , $\operatorname{argmin}_{\mathbf{x} \in \operatorname{aff}(S)} \|\mathbf{x}\|_2$. The second is the *convex minimizer* which is the point of minimum norm in the convex hull of S , $\operatorname{argmin}_{\mathbf{x} \in \operatorname{conv}(S)} \|\mathbf{x}\|_2$. Note that solving for the convex minimizer of a set of points is exactly the problem we are solving, while solving for the affine minimizer of a set of points is easily computable by solving a system of linear equations and thus may be computed in strongly-polynomial time; see [Schrijver 1998, Section 3.3] and references therein.

2.1 A Brief Review of Wolfe’s Combinatorial Method

Wolfe’s combinatorial method solves the minimum norm point problem over a polytope, $P = \operatorname{conv}(\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_n) \subset \mathbb{R}^d$, and was introduced by P. Wolfe in [Wolfe 1976]. The method iteratively solves the minimum norm point problem over a sequence of subsets of no more than $d + 1$ affinely independent points from $\mathbf{p}_1, \dots, \mathbf{p}_n$ and it checks to see if the solution to the subproblem is a solution to the problem over P using the following lemma due to Wolfe. We call this *Wolfe’s criterion*.

LEMMA 2.1 (WOLFE’S CRITERION [WOLFE 1976]). *Let $P \subset \mathbb{R}^d$ be the convex hull of finitely many points, $P := \operatorname{conv}(\mathbf{p}_1, \dots, \mathbf{p}_n)$, then $\mathbf{x} \in P$ is the minimum norm point in P if and only if $\mathbf{x}^T \mathbf{p}_j \geq \|\mathbf{x}\|_2^2$ for all $j \in [n]$.*

Note that this tells us that if there exists a point p_j so that $\mathbf{x}^T \mathbf{p}_j < \|\mathbf{x}\|_2^2$ (i.e., the hyperplane $\{\mathbf{y} : \mathbf{x}^T \mathbf{y} = \|\mathbf{x}\|_2^2\}$ does not weakly separate P from $\mathbf{0}$), then \mathbf{x} is not the minimum norm point in P . We say that p_j violates Wolfe’s criterion and using this point should decrease the norm of the minimum norm point of the current subproblem.

It should be observed that just as Wolfe’s criterion is a rule to decide optimality over $\operatorname{conv}(P)$, one has a very similar rule for deciding optimality over the affine hull, $\operatorname{aff}(P)$.

LEMMA 2.2 (WOLFE’S CRITERION FOR THE AFFINE HULL). *Let $P = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_n\} \subseteq \mathbb{R}^d$ be a non-empty finite set of points. Then $\mathbf{x} \in \operatorname{aff} P$ is the minimum norm point in $\operatorname{aff} P$ iff for all $\mathbf{p}_i \in P$ we have $\mathbf{p}_i^T \mathbf{x} = \|\mathbf{x}\|_2^2$.*

We say a set of affinely independent points S is a *corral* if the affine minimizer of S lies in the relative interior of $\operatorname{conv} S$. Requiring the affine minimizer to lie in the *relative interior* of the convex hull ensures that corrals are of minimal size and without points unnecessary for expressing the affine minimizer as a convex combination of the corral. Note that singletons are always corrals. Carathéodory’s theorem implies that the minimum norm point of P will lie in the convex hull of some corral of points among $\mathbf{p}_1, \dots, \mathbf{p}_n$. The goal of

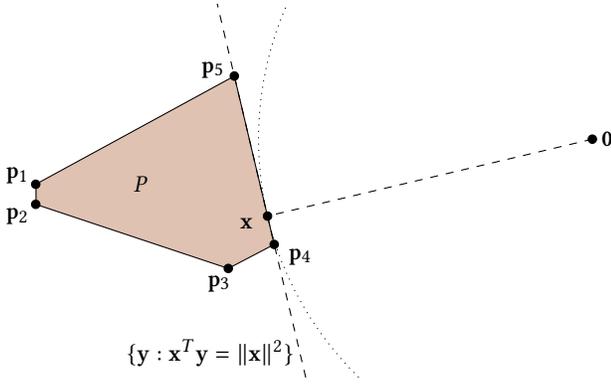


Figure 1: A visualization of Wolfe's criterion. Note that $\{y : x^T y = \|x\|^2\}$ weakly separates P from 0 , so x is the minimum norm point in P .

Wolfe's method is to search for a corral containing the (unique) minimizing point.

The pseudo-code in Method 1 below presents the iterations of Wolfe's method. It is worth noticing that some steps of the method can be implemented in more than one way and Wolfe proved that all of them lead to a correct, terminating algorithm (for example, the choice of the initial point in line 2). We therefore use the word *method* to encompass all these variations and we discuss specific choices when they are relevant to our analysis of the method.

Method 1 Wolfe's Method [Wolfe 1976]

```

1: procedure WOLFE( $p_1, p_2, \dots, p_n$ )
2:   Initialize  $x = p_i$  for some  $i \in [n]$ , initial corral
    $C = \{p_i\}$ ,  $I = \{i\}$ ,  $\lambda = e_i$ ,  $\alpha = 0$ .
3:   while  $x \neq 0$  and there exists  $p_j$  with  $x^T p_j < \|x\|_2^2$  do
4:     Add  $p_j$  to the potential corral:  $C = C \cup \{p_j\}$ ,
      $I = I \cup \{j\}$ .
5:     Find the affine minimizer of  $C$ ,
      $y = \operatorname{argmin}_{y \in \operatorname{aff}(C)} \|y\|_2$ , and the affine
     coefficients,  $\alpha$ .
6:     while  $y$  is not a strict convex combination of
     points in  $C$ ;  $\alpha_i \leq 0$  for some  $i \in I$  do
7:       Find  $z$ , closest point to  $y$  on  $[x, y] \cap \operatorname{conv}(C)$ ;
        $z = \theta y + (1 - \theta)x$ ,  $\theta = \min_{i \in I: \alpha_i \leq 0} \frac{\lambda_i}{\lambda_i - \alpha_i}$ .
8:       Select  $p_i \in \{p_j \in C : \theta \alpha_j + (1 - \theta) \lambda_j = 0\}$ .
9:       Remove this point from  $C$ ;  $C = C - \{p_i\}$ ,
        $I = I - \{i\}$ ,  $\alpha_i = 0$ ,  $\lambda_i = 0$ .
10:      Update  $x = z$  and the convex coefficients,  $\lambda$ , of
        $x$  for  $C$ ; solve  $x = \sum_{p_i \in C} \lambda_i p_i$  for  $\lambda$ .
11:      Find the affine minimizer of  $C$ ,
        $y = \operatorname{argmin}_{y \in \operatorname{aff}(C)} \|y\|_2$  and the affine
       coefficients,  $\alpha$ .
12:     end while
13:     Update  $x = y$  and  $\lambda = \alpha$ .
14:   end while
15:   Return  $x$ .
16: end procedure

```

The subset of points being considered as the *potential corral* is maintained in the set C . Iterations of the outer-loop, where points are added to C , are called *major cycles* and iterations of the inner-loop, where points are removed from C , are called *minor cycles*. The potential corral, C , is named so because at the beginning of a major cycle it is guaranteed to be a corral, while within the minor cycles it may or may not be a corral. Intuitively, a major cycle of Wolfe's method inserts an *improving point* which violates Wolfe's criterion (p_j so that $x^T p_j < \|x\|_2^2$) into C , then the minor cycles remove points until C is a corral, and this process is repeated until no points are improving and C is guaranteed to be a corral containing the minimizer. It can be shown that this method terminates because the norm of the convex minimizer of the corrals visited monotonically decreases and thus, no corral is visited twice [Wolfe 1976].

Within the method, there are two moments at which one may choose which points to add to the potential corral. Observe that at line 2 of the pseudocode, one may choose which initial point to add to the potential corral. In this paper we will only consider one *initial rule*, which is to initialize with the point of minimum norm. Observe that at line 4 of the pseudocode, there are several potential choices of which point to add to the potential corral. Two important examples of *insertion rules* are, first, the *minnorm rule* which dictates that one chooses, out of the improving points for the potential corral, to add the point p_j of minimum norm. Second, the *linopt rule* dictates that one chooses, out of the improving points for the potential corral, to add the point p_j minimizing $x^T p_j$. Notice that insertion rules are to Wolfe's method what *pivot rules* are to the Simplex Method (see [Terlaky and Zhang 1993] for a summary).

As with pivot rules, there are advantages and disadvantages of insertion rules. For example, the minnorm rule has the advantage that its implementation only requires an initial ordering of the points, then in each iteration it need only to search for an improving point in order of increasing norm and to add the first found. However, the linopt insertion rule has the advantage that, if the polytope is given in H-representation (intersection of halfspaces) rather than V-representation (convex hull of points), one may still perform Wolfe's method by using linear programming to find p_j minimizing $x^T p_j$ over the polytope. In other words, Wolfe's method does not need to have the list of vertices explicitly given, but suffices to have a linear programming oracle that provides the new vertex to be inserted. This feature of Wolfe's method means that each iteration can be implemented efficiently even for certain polyhedra having too many vertices and facets: specifically, over zonotopes (presented as a Minkowski sum of segments) [Fujishige et al. 2006] and over the base polyhedron of a submodular function [Fujishige 1980].

Wolfe's method with the linopt insertion rule (otherwise called the Lawson-Hanson algorithm for non-negative least squares [Lawson and Hanson 1995, Section 23.3]) is similar to other active-set methods in convex optimization. In particular, von Neumann's algorithm for determining whether the origin lies in a convex polytope [Dantzig 1992], Gilbert's procedure for computing the minimum of a quadratic form on a convex set [Gilbert 1966], and the Frank-Wolfe method for convex optimization [Frank and Wolfe 1956] make use of the same active-set selection criterion, the linopt insertion rule. Several of these methods and their variants have been shown to

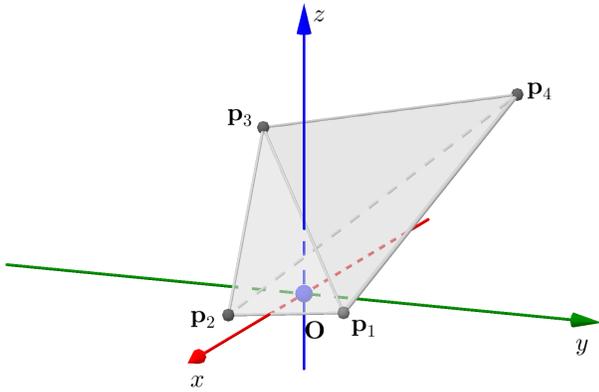


Figure 2: The simplex $P = \text{conv}\{p_1, p_2, p_3, p_4\} \subset \mathbb{R}^3$ where $p_1 = (0.8, 0.9, 0)$, $p_2 = (1.5, -0.5, 0)$, $p_3 = (-1, -1, 2)$ and $p_4 = (-4, 1.5, 2)$.

have sublinear or linear convergence (with appropriate assumptions on the location of the minimizer); see e.g., [Lacoste-Julien and Jaggi 2013, 2015; Peña et al. 2016]. In [Lacoste-Julien and Jaggi 2015], the authors additionally studied Wolfe’s method with the linopt insertion rule, showing that the method converges linearly with a rate $O(e^{-\rho t})$ where ρ is an eccentricity parameter of the polytope. This improved upon prior results by [Chakrabarty et al. 2014], which had provided a sublinear ($O(1/t)$) rate. Note, however, that the parameter ρ defined in [Lacoste-Julien and Jaggi 2015] may be exponentially small in the encoding length of the problem, so both results give a pseudo-polynomial time bound for Wolfe’s method with the linopt insertion rule.

We present examples that show that the optimal choice of insertion rule depends on the input data. First, we present a simple example where the minnorm rule outperforms the linopt rule. That is, the minnorm insertion rule is not in obvious disadvantage to the linopt rule. This is in contrast to the family of examples we present in Section 2.2 where the minnorm rule takes exponential time, while we expect the linopt rule to take polynomial time.

Consider the simplex P shown in Figure 2 (we present the coordinates of vertices in the figure’s caption). We list the steps of Wolfe’s method on P for the minnorm and linopt insertion rules in Tables 1 and 2 and demonstrate a single step from each set of iterations in Figure 3. Each row lists major cycle and minor cycle iteration number, the vertices in the potential corral, and the value of x and y at the end of the iteration (before $x = y$ for major cycles). Note that the vertex p_4 is added to the potential corral twice with the linopt insertion rule, as evidenced in Table 2.

Currently, there are examples of exponential behavior for the simplex method for all known deterministic pivot rules. It is our aim to provide the same for insertion rules on Wolfe’s method. In the next subsection we will present the first exponential time example using the minnorm insertion rule.

Table 1: iterations for *minnorm* insertion rule on simplex P . Cycle i, j denotes the i th major cycle and the j th minor cycle within.

Cycle	C	x	y
0,0	$\{p_1\}$	p_1	
1,0	$\{p_1, p_2\}$	p_1	(1, 0.5, 0)
2,0	$\{p_1, p_2, p_3\}$	(1, 0.5, 0)	(0.398, 0.199, 0.547)
3,0	$\{p_1, p_2, p_3, p_4\}$	(0.398, 0.199, 0.547)	(0, 0, 0)
3,1	$\{p_1, p_2, p_4\}$	(0.288, 0.144, 0.396)	(0.198, 0.099, 0.446)

Table 2: iterations for *linopt* insertion rule on simplex P . Cycle i, j denotes the i th major cycle and the j th minor cycle within.

Cycle	C	x	y
0,0	$\{p_1\}$	p_1	
1,0	$\{p_1, p_4\}$	p_1	(0.222, 0.972, 0.241)
2,0	$\{p_1, p_4, p_3\}$	(0.222, 0.972, 0.241)	(0.285, 0.342, 0.581)
2,1	$\{p_1, p_3\}$	(0.286, 0.355, 0.574)	(0.277, 0.348, 0.581)
3,0	$\{p_1, p_3, p_2\}$	(0.277, 0.348, 0.581)	(0.398, 0.199, 0.547)
4,0	$\{p_1, p_2, p_3, p_4\}$	(0.398, 0.199, 0.547)	(0, 0, 0)
4,1	$\{p_1, p_2, p_4\}$	(0.288, 0.144, 0.396)	(0.198, 0.099, 0.446)

2.2 An Exponential Lower Bound for Wolfe’s Method

To understand our hard instance, it is helpful to consider first a simple instance that shows an inefficiency of Wolfe’s method. The example is a set of points where a point leaves and reenters the current corral: 4 points in \mathbb{R}^3 , $(1, 0, 0)$, $(1/2, 1/4, 1)$, $(1/2, 1/4, -1)$, $(-2, 1/4, 0)$. If one labels the points 1, 2, 3, 4, the sequence of corrals with the minnorm rule is 1, 12, 23, 234, 14, where point 1 enters, leaves and reenters (For succinctness, sets of points like $\{a, b, c\}$ may be denoted abc). The idea now is to recursively replace point 1 (that reenters) in this construction by a recursively constructed set of points whose corrals are then considered twice by Wolfe’s method. To simplify the proof, our construction uses a variation of this set of 4 points with an additional point and modified coordinates. This modified construction is depicted in Figure 4, where point 1 corresponds to a set of points $P(d - 2)$, points 2,3 correspond to points p_d, q_d and point 4 corresponds to points r_d, s_d .

The high-level idea of our exponential lower bound example is the following. We will inductively define a sequence of instances of increasing dimension of the minimum norm point problem. Given an instance in dimension $d - 2$, we will add a few dimensions and points so that, when given to Wolfe’s method, the number of corrals of the new augmented instance in dimension d has about twice the number of corrals of the input instance in dimension $d - 2$. More precisely, our augmentation procedure takes an instance $P(d - 2)$ in

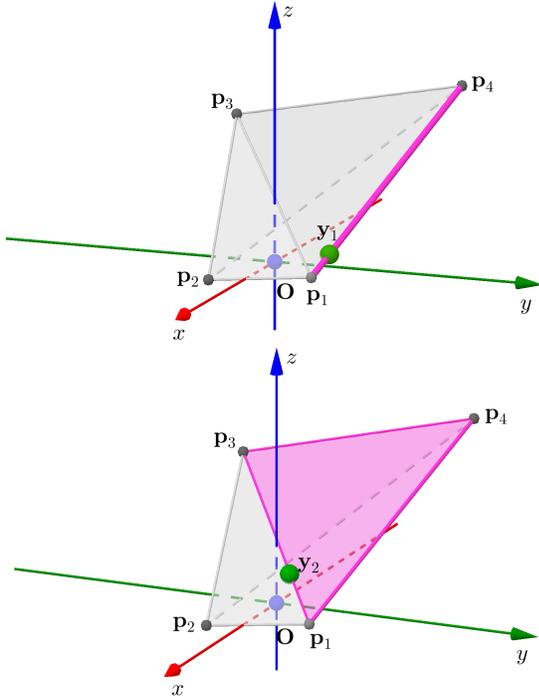


Figure 3: Top: Major cycle 1, minor cycle 0 for the linopt rule on P illustrates the end of a major cycle; the affine minimizer $y_1 \in \text{relint}(\text{conv}\{C\}) = \text{relint}(\text{conv}\{p_1, p_4\})$. **Bottom:** Major cycle 2, minor cycle 0 for the linopt rule on P illustrates the beginning of a minor cycle; the affine minimizer $y_2 \notin \text{relint}(\text{conv}\{C\}) = \text{relint}(\text{conv}\{p_1, p_4, p_3\})$ and the vertex p_4 will be removed in the next minor cycle.

\mathbb{R}^{d-2} , adds two new coordinates and adds four points, p_d, q_d, r_d, s_d , to get an instance $P(d)$ in \mathbb{R}^d .

Points p_d, q_d are defined so that the method on instance $P(d)$ goes first through every corral given by the points in the prior configuration $P(d-2)$ and then goes to corral $p_d q_d$. To achieve this under the minimum norm rule, the four new points have greater norm than any point in $P(d-2)$ and they are in the geometric configuration sketched in Figure 4.

At this time, no point in $P(d-2)$ is in the current corral and so, if a point in $P(d-2)$ is part of the optimal corral, it will have to reenter, which is expensive. Points r_d, s_d are defined so that $r_d s_d$ is a corral after $p_d q_d$, but now every point in $P(d-2)$ is improving according to Wolfe's criterion and may enter again. Specifically, every corral in $P(d-2)$, with $r_d s_d$ appended, is visited again.

Before we start describing the exponential example in detail, we wish to review preliminary lemmas of independent interest which will be used in the arguments. The first lemma demonstrates that orthogonality between finite point sets allows us to easily describe the affine minimizer of their union. Figure 5 shows two such situations, one in which the affine hull of the union of the point sets span all of \mathbb{R}^3 and one in which it does not.

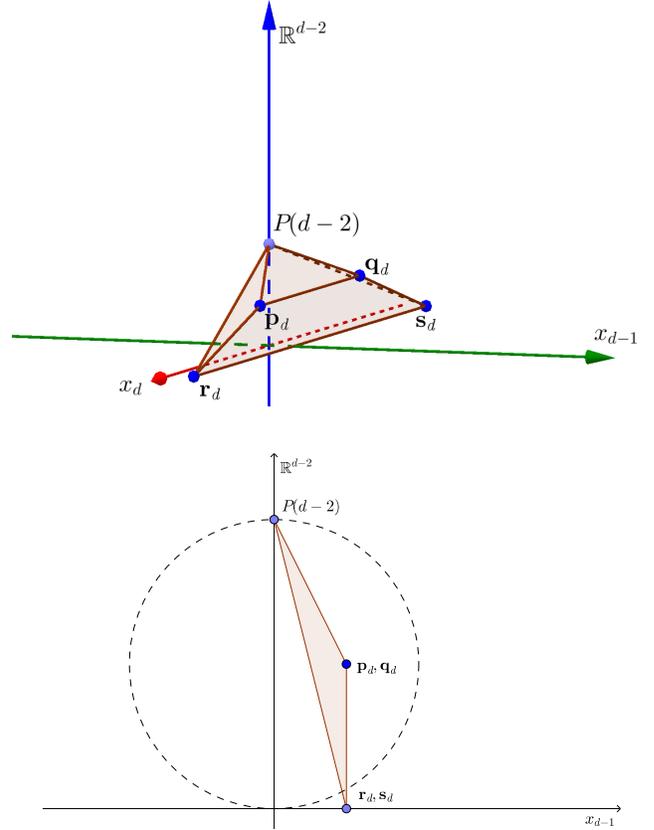


Figure 4: Top: In this view of $P(d)$, the point labeled $P(d-2)$ represents all points from $P(d-2)$ embedded into \mathbb{R}^d . The axis labeled \mathbb{R}^{d-2} represents the $(d-2)$ -dimensional subspace, $\text{span}(P(d-2))$ projected into $\text{span}(\mathbf{o}_{d-2}^*)$. **Bottom:** A two-dimensional view of $P(d)$ projected along the x_d coordinate axis.

LEMMA 2.3. Let $A \subseteq \mathbb{R}^d$ be a proper linear subspace. Let $P \subseteq A$ be a non-empty finite set. Let $Q \subseteq A^\perp$ be another non-empty finite set. Let \mathbf{x} be the minimum norm point in $\text{aff } P$. Let \mathbf{y} be the minimum norm point in $\text{aff } Q$. Let \mathbf{z} be the minimum norm point in $\text{aff}(P \cup Q)$. We have:

- (1) \mathbf{z} is the minimum norm point in $[\mathbf{x}, \mathbf{y}]$ and therefore, if $\mathbf{x} \neq \mathbf{0}$ or $\mathbf{y} \neq \mathbf{0}$, then $\mathbf{z} = \lambda \mathbf{x} + (1 - \lambda) \mathbf{y}$ with $\lambda = \frac{\|\mathbf{y}\|_2^2}{\|\mathbf{x}\|_2^2 + \|\mathbf{y}\|_2^2}$.
- (2) If $\mathbf{x} \neq \mathbf{0}$ and $\mathbf{y} \neq \mathbf{0}$, then \mathbf{z} is a strict convex combination of \mathbf{x} and \mathbf{y} .
- (3) If $\mathbf{x} \neq \mathbf{0}$, $\mathbf{y} \neq \mathbf{0}$ and P and Q are corrals, then $P \cup Q$ is also a corral.

The following lemma shows conditions under which, if we have a corral and a new point that only has components along the minimum norm point of the corral and along new coordinates, then the corral with the new point added is also a corral. Moreover, the new minimum norm point is a convex combination of the old minimum norm point and the added point. Figure 6 gives an example of such a situation in \mathbb{R}^3 . Denote by $\text{span}(M)$ the linear span of the set M .

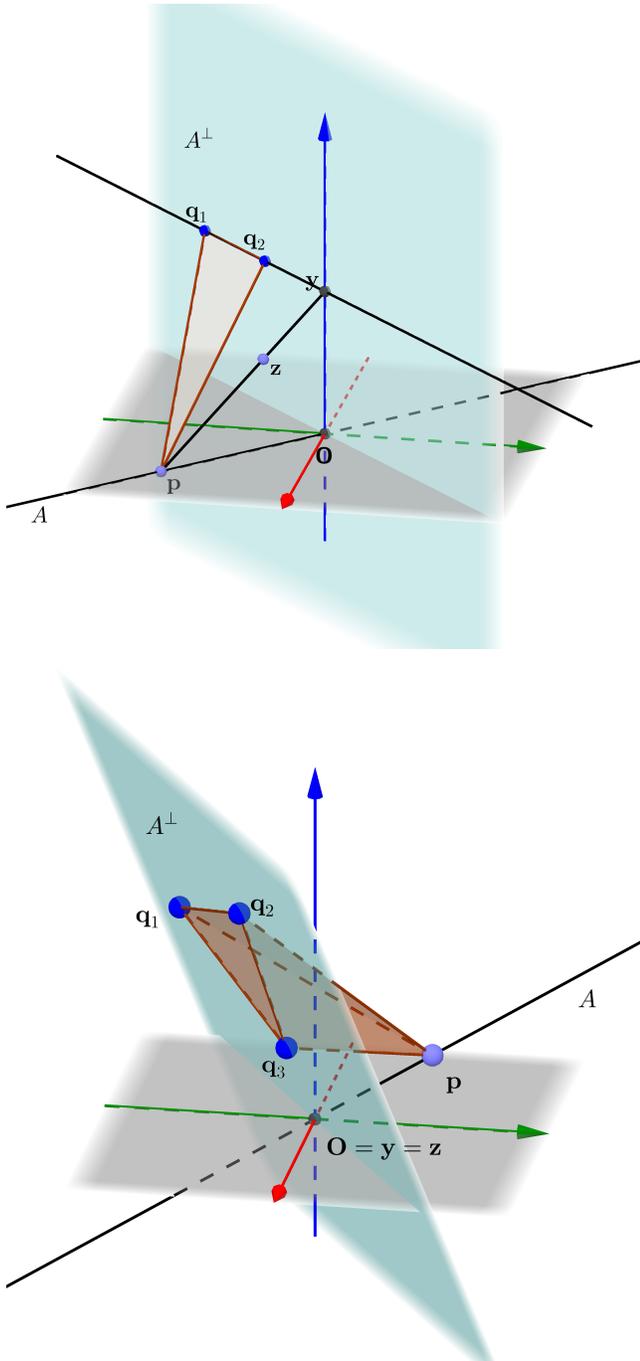


Figure 5: Examples of Lemma 2.3. Top: the affine hull of $P \cup Q$ is not full dimensional, and thus the affine minimizer lies at z along the line segment connecting $x = p$ and y . Bottom: the convex hull of $P \cup Q$ is full-dimensional and thus the affine hull of $P \cup Q$ includes O which is the affine minimizer.

LEMMA 2.4. Let $P \subseteq \mathbb{R}^d$ be a finite set of points that is a corral. Let x be the minimum norm point in $\text{aff } P$. Let $q \in \text{span}(x, \text{span}(P)^\perp)$, and

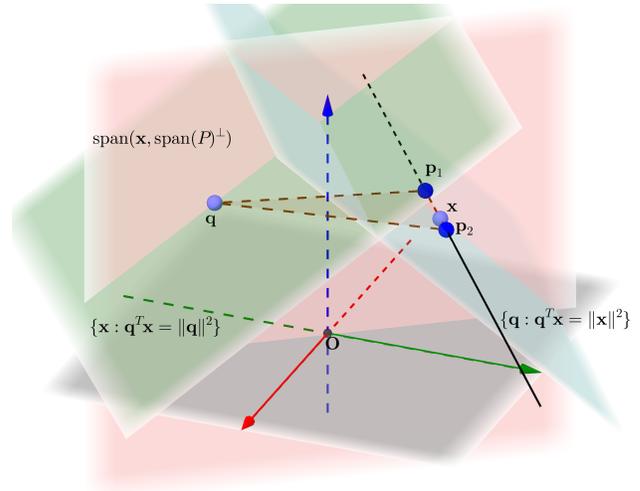


Figure 6: An example of Lemma 2.4 in which point q satisfies all assumptions and $P \cup \{q\}$ is a corral. The hyperplanes are labeled with their defining properties and demonstrate that $q^T x < \min\{\|x\|^2, \|q\|^2\}$. The minimizer of $P \cup \{q\}$ lies at the intersection of the blue, vertical axis and $\text{conv}(P \cup \{q\})$.

assume $q^T x < \min\{\|q\|_2^2, \|x\|_2^2\}$. Then $P \cup \{q\}$ is a corral. Moreover, the minimum norm point y in $\text{conv}(P \cup \{q\})$ is a (strict) convex combination of q and the minimum norm point of P : $y = \lambda x + (1-\lambda)q$ with $\lambda = q^T(q-x)/\|q-x\|_2^2$.

Our last lemma shows that if we have points in two orthogonal subspaces, A and A^\perp , then adding a point from A^\perp to a set from A does not cause any points from A that previously did not violate Wolfe’s criterion (for the affine minimizer) to violate it. Figure 7 demonstrates this situation.

LEMMA 2.5. For a point z define $H_z = \{w \in \mathbb{R}^n : w^T z < \|z\|_2^2\}$. Suppose that we have an instance of the minimum norm point problem in \mathbb{R}^d as follows: Some points, P , live in a proper linear subspace A and some, Q , in A^\perp . Let x be the minimum norm point in $\text{aff } P$ and y be the minimum norm point in $\text{aff}(P \cup Q)$. Then $H_y \cap A = H_x \cap A$.

We will now describe our example in detail. The simplest version of our construction uses square roots and real numbers. We present instead a version with a few additional tweaks so that it only involves rational numbers.

Let $P(1) = \{1\} \subseteq \mathbb{Q}$. For odd $d > 1$, let $P(d)$ be a list of points in \mathbb{Q}^d defined inductively as follows: Let o_d^* denote the minimum norm point in $\text{conv } P(d)$. Let $M_d := \max_{p \in P(d)} \|p\|_1$, which is a rational upper bound to the maximum 2-norm among the points in $P(d)$. (For a first reading one can let M_d be the maximum 2-norm among points in $P(d)$, which leads to an essentially equivalent instance except that it is not rational.) Similarly, let $m_d = \|o_d^*\|_\infty$, which is a rational lower bound to the minimum norm among points in $\text{conv } P(d)$. (Again, for a first reading one can define $m_d = \|o_d^*\|_2$ which leads to an essentially equivalent instance, except that it is not rational.)

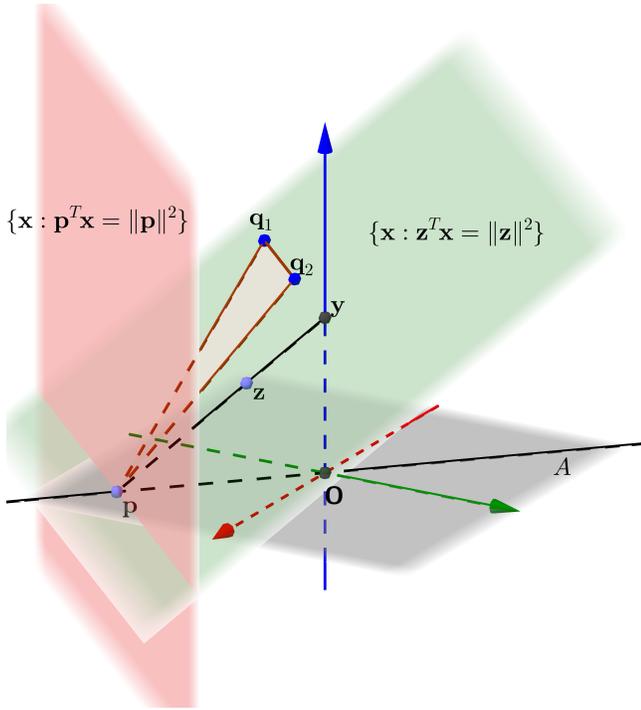


Figure 7: An example of Lemma 2.5 in which adding points Q from A^\perp to points P from A create a new affine minimizer, z , but the points satisfying Wolfe’s criterion in A remain the same. Note that both hyperplanes intersect at the affine minimizer of P , so the halfspace intersections with A are the same.

We finally present the example. If we identify $P(d)$ with a matrix where the points are rows, then the points in $P(d)$ are given by the following block matrix:

$$P(d) = \begin{pmatrix} P(d-2) & 0 & 0 \\ \frac{1}{2} \mathbf{o}_{d-2}^* & \frac{m_{d-2}}{4} & M_{d-2} \\ \frac{1}{2} \mathbf{o}_{d-2}^* & \frac{m_{d-2}}{4} & -(M_{d-2} + 1) \\ 0 & \frac{m_{d-2}}{4} & M_{d-2} + 2 \\ 0 & \frac{m_{d-2}}{4} & -(M_{d-2} + 3) \end{pmatrix}.$$

The last four rows of the matrix $P(d)$ are the points $\mathbf{p}_d, \mathbf{q}_d, \mathbf{r}_d, \mathbf{s}_d$ of the configuration. For a picture of the case of $P(3)$ see Figure 8. We also present the sets C , and points \mathbf{x} and \mathbf{y} defined throughout the iterations of Wolfe’s method with the minnorm insertion rule on $P(3)$ in Table 3. For comparison only, we include the same for the iterations of Wolfe’s method with the linopt insertion rule on $P(3)$ in Table 4.

Remark: First note that strictly speaking $P(d-2) \subset \mathbb{Q}^{d-2}$, and that we are defining an embedding of it into \mathbb{Q}^d , for which we have to use a recursive process. To avoid unnecessary notation, we will abuse the notation. The point \mathbf{v}_{d-2} denotes both a point of $P(d-2)$ and of the subsequent $P(d)$, i.e., $\mathbf{v}_{d-2} = (\mathbf{v}, 0, 0)$ will be the identical copy of \mathbf{v}_{d-2} within $P(d)$, but we add two extra zero coordinates. Depending on the context \mathbf{v}_{d-2} will be understood

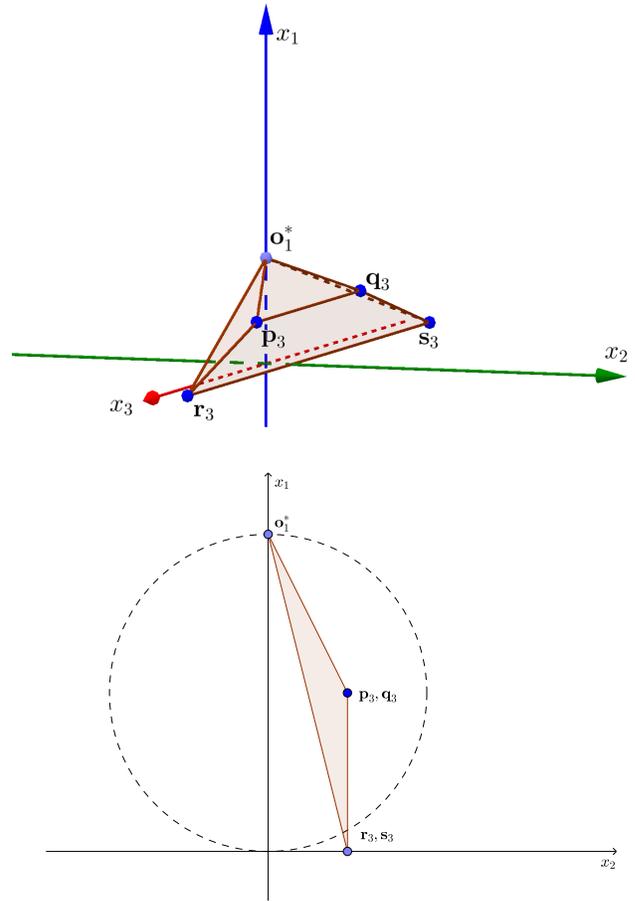


Figure 8: Top: Three-dimensional view of $P(3)$. Bottom: A two-dimensional view of $P(3)$ projected along the x_3 coordinate axis.

as both a $(d-2)$ -dimensional vector or as a d -dimensional vector (e.g., when doing dot products). The points of $P(d-2)$ become a subset of the point configuration $P(d)$ by padding extra zeros. See Figures 4 and 9 which illustrate this embedding and address our visualizations of these sets in three dimensions.

THEOREM 2.6. *Consider the execution of Wolfe’s method with the minnorm point rule on input $P(d)$ where $d = 2k-1$. Then the sequence of corrals has length $5 \cdot 2^{k-1} - 4$.*

The inductive proof of this result relies upon the three lemmas stated above. Each of these are applied in various steps that characterize the sequence of corrals visited for $P(d)$ in terms of the sequence of corrals visited for $P(d-2)$. Moreover, this characterization shows that the length of the sequence of corrals at least doubles in length for each step from $P(d-2)$ to $P(d)$.

Table 3: iterations for *minnorm* insertion rule on $P(3)$. Cycle i, j denotes the i th major cycle and the j th minor cycle within. The point $(1, 0, 0)$ is denoted \mathbf{o}^* .

Cycle	C	\mathbf{x}	\mathbf{y}
0, 0	$\{\mathbf{o}^*\}$	\mathbf{o}^*	
1, 0	$\{\mathbf{o}^*, \mathbf{p}_3\}$	\mathbf{o}^*	(0.810, 0.095, 0.381)
2, 0	$\{\mathbf{o}^*, \mathbf{p}_3, \mathbf{q}_3\}$	(0.810, 0.095, 0.381)	(0.2, 0.4, 0)
2, 1	$\{\mathbf{p}_3, \mathbf{q}_3\}$	(0.5, 0.25, 0.1875)	(0.5, 0.25, 0)
3, 0	$\{\mathbf{p}_3, \mathbf{q}_3, \mathbf{r}_3\}$	(0.5, 0.25, 0)	(0, 0.25, 0)
3, 1	$\{\mathbf{q}_3, \mathbf{r}_3\}$	(0.3, 0.25, 0)	(0.297, 0.25, 0.0297)
4, 0	$\{\mathbf{q}_3, \mathbf{r}_3, \mathbf{s}_3\}$	(0.297, 0.25, 0.0297)	(0, 0.25, 0)
4, 1	$\{\mathbf{r}_3, \mathbf{s}_3\}$	(0, 0.25, 0)	(0, 0.25, 0)
5, 0	$\{\mathbf{r}_3, \mathbf{s}_3, \mathbf{o}^*\}$	(0, 0.25, 0)	(0.059, 0.235, 0)

Table 4: iterations for *linopt* insertion rule on $P(3)$. Cycle i, j denotes the i th major cycle and the j th minor cycle within. The point $(1, 0, 0)$ is denoted \mathbf{o}^* .

Cycle	C	\mathbf{x}	\mathbf{y}
0, 0	$\{\mathbf{o}^*\}$	\mathbf{o}^*	
1, 0	$\{\mathbf{o}^*, \mathbf{r}_3\}$	\mathbf{o}^*	(0.901, 0.025, 0.298)
2, 0	$\{\mathbf{o}^*, \mathbf{r}_3, \mathbf{s}_3\}$	(0.901, 0.025, 0.298)	(0.059, 0.235, 0)

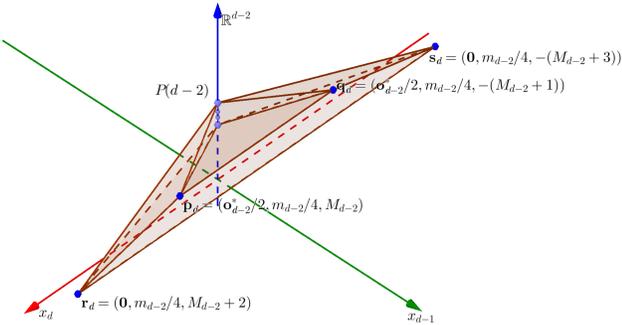


Figure 9: As described in Figure 4, the axis labeled \mathbb{R}^{d-2} represents the $(d - 2)$ -dimensional subspace $\text{span}(P(d - 2))$ projected onto the one dimensional subspace $\text{span}(\mathbf{o}_{d-2}^*)$. Here we illustrate that the projection of the set $P(d - 2)$ forms a ‘cloud’ of points and the convex hull of this projection has many fewer faces than the unprojected convex hull. For simplicity, we will visualize $P(d - 2)$ and subsets of $P(d - 2)$ as a single point in $\text{span}(\mathbf{o}_{d-2}^*)$ as in Figure 4.

3 LINEAR OPTIMIZATION REDUCES TO MINIMUM-NORM PROBLEMS ON SIMPLICES

We reduce linear programming to the minimum norm point problem over a simplex via a series of strongly polynomial time reductions. The algorithmic problems we consider are defined below.

Definition 3.1. Consider the following computational problems:

- **LP:** Given a rational matrix A , a rational column vector \mathbf{b} , and a rational row vector \mathbf{c}^T , output rational $\mathbf{x} \in \text{argmax}\{\mathbf{c}^T \mathbf{x} : A\mathbf{x} \leq \mathbf{b}\}$ if $\max\{\mathbf{c}^T \mathbf{x} : A\mathbf{x} \leq \mathbf{b}\}$ is finite, otherwise output INFEASIBLE if $\{\mathbf{x} : A\mathbf{x} \leq \mathbf{b}\}$ is empty and else output INFINITE.
- **DVS:** Given $n \leq d + 1$ affinely independent rational points $\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_n \in \mathbb{R}^d$ defining $(n - 1)$ -dimensional simplex $P = \text{conv}\{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_n\}$, output $d(\mathbf{0}, P)^2$.

The main result in this section reduces linear programming to finding the minimum norm point in a (vertex-representation) simplex.

THEOREM 3.2. LP reduces to DVS in strongly-polynomial time.

This result is proved via a series of strongly-polynomial reductions which may be found in the full version [De Loera et al. 2017]. The first reductions, which reduce LP to the problem of deciding if the origin lies in a polytope given in vertex-representation (which we call ZVPM), are classically known, although we do not believe they have all been written elsewhere. The last reduction, which reduces ZVPM to DVS, is novel and relies upon a careful application of Gram-Schmidt orthogonalization to construct a lift of the polytope to a simplex in higher dimension. We comment that the application of Gram-Schmidt without normalization causes the bit-size of the numbers involved in the problem to grow, but only polynomially, thus preserving the strong-polynomiality of the reduction; see [Grötschel et al. 1988, Section 1.4], [Schrijver 1998, Section 3.3].

4 CONCLUSIONS AND OPEN QUESTIONS

We have seen that Wolfe’s method using a natural point insertion rule exhibits exponential behavior. We have also shown that the minimum norm point problem for simplices is intimately related to the complexity of linear programming. Our work raises several very natural questions:

- Are there exponential examples for other insertion rules for Wolfe’s method? Also, at the moment, the ordering of the points starts with the closest point to the origin, but one could also consider a randomized initial rule or a randomized insertion rule.
- For applications in submodular function minimization, the polytopes one considers are base polytopes and our exponential example is not of this kind. Could there be hope that for base polytopes Wolfe’s method performs better?
- It would be interesting to understand the average performance of Wolfe’s method. How does it behave for random data? Further randomized analysis of this method would include the smoothed analysis of Wolfe’s method or at least the behavior for data following a prescribed distribution.

- We have seen that it is already quite interesting to study the minimum norm point problem for simplices, when we discussed the connection with linear programming. Is there a family of simplices where Wolfe's method takes exponential time?
- Can Wolfe's method be extended to other convex L_p norms for $p \geq 1$? Can we identify the types of objective functions for which computing the affine minimizer is easy?

ACKNOWLEDGMENTS

We thank Gilberto Calvillo, Deeparnab Chakrabarty, Antoine Deza, Stephanie Jegelka, Matthias Köppe, Tamon Stephen, and John Sullivan for useful suggestions, conversations and comments about this project. This work was partially supported by NSF grant DMS-1440140, while the authors were in residence at the Mathematical Sciences Research Institute in Berkeley, California, during the Fall 2017 semester. The first and second author were also partially supported by NSF grant DMS-1522158. The second author was also partially supported by the University of California, Davis Dissertation Fellowship. The third author was also partially supported by NSF grants CCF-1657939 and CCF-1422830 while the third author was in residence at the Simons Institute for the Theory of Computing.

REFERENCES

- Francis Bach et al. 2013. Learning with submodular functions: A convex optimization perspective. *Foundations and Trends® in Machine Learning* 6, 2-3 (2013), 145–373.
- Achim Bachem and Bernhard Korte. 1980. Minimum norm problems over transportation polytopes. *Linear Algebra Appl.* 31 (1980), 103–118.
- Imre Bárány and Shmuel Onn. 1997. Colourful linear programming and its relatives. *Mathematics of Operations Research* 22, 3 (1997), 550–567.
- Stephen Boyd and Lieven Vandenbergh. 2004. *Convex optimization*. Cambridge university press.
- Gilberto Calvillo and David Romero. 2016. On the closest point to the origin in transportation polytopes. *Discrete Applied Mathematics* 210 (2016), 88–102.
- Deeparnab Chakrabarty, Prateek Jain, and Pravesh Kothari. 2014. Provable submodular minimization using Wolfe's algorithm. In *Advances in Neural Information Processing Systems*. 802–809.
- Vijay Chandru, Abhi Dattasharma, S Sathiyaraj Keerthi, NK Sancheti, and V Vinay. 1995. Algorithms for the optimal loading of recursive neural nets. In *SODA*. 342–349.
- George Bernard Dantzig. 1992. An ϵ -precise feasible solution to a linear program with a convexity constraint in $1/\epsilon^2$ iterations independent of problem size. Technical Report. Stanford University.
- Jesus A. De Loera, Jamie Haddock, and Luis Rademacher. 2017. The Minimum Euclidean-Norm Point on a Convex Polytope: Wolfe's Combinatorial Algorithm is Exponential. arXiv:arXiv:1710.02608
- Marguerite Frank and Philip Wolfe. 1956. An algorithm for quadratic programming. *Naval Research Logistics (NRL)* 3, 1-2 (1956), 95–110.
- Satoru Fujishige. 1980. Lexicographically Optimal Base of a Polymatroid with Respect to a Weight Vector. *Math. Oper. Res.* 5, 2 (1980), 186–196. <https://doi.org/10.1287/moor.5.2.186>
- Satoru Fujishige, Takumi Hayashi, and Shiguelo Isotani. 2006. *The minimum-norm-point algorithm applied to submodular function minimization and linear programming*. Kyoto University. Research Institute for Mathematical Sciences [RIMS].
- Satoru Fujishige and Shiguelo Isotani. 2011. A submodular function minimization algorithm based on the minimum-norm base. *Pacific Journal of Optimization* 7 (2011), 3–17.
- Komei Fukuda, Thomas M. Liebling, and François Margot. 1997. Analysis of backtrack algorithms for listing all vertices and all faces of a convex polyhedron. *Computational Geometry* 8, 1 (1997), 1–12. [https://doi.org/10.1016/0925-7721\(95\)00049-6](https://doi.org/10.1016/0925-7721(95)00049-6)
- Dongdong Ge, Xiaoye Jiang, and Yinyu Ye. 2011. A note on the complexity of L_p minimization. *Mathematical Programming* 129, 2 (01 Oct 2011), 285–299. <https://doi.org/10.1007/s10107-011-0470-2>
- Elmer G. Gilbert. 1966. An iterative procedure for computing the minimum of a quadratic form on a convex set. *SIAM J. Control* 4 (1966), 61–80.
- Martin Grötschel, László Lovász, and Alexander Schrijver. 1988. *Geometric Algorithms and Combinatorial Optimization*. Springer-Verlag, Berlin.
- Victor Klee and George J. Minty. 1972. How good is the simplex algorithm? In *Inequalities, III (Proc. Third Sympos., Univ. California, Los Angeles, Calif., 1969; dedicated to the memory of Theodore S. Motzkin)*. Academic Press, New York, 159–175.
- Mikhail K. Kozlov, Sergei P. Tarasov, and Leonid G. Khachiyan. 1980. The polynomial solvability of convex quadratic programming. *U. S. S. R. Comput. Math. and Math. Phys.* 20, 5 (1980), 223–228.
- Simon Lacoste-Julien and Martin Jaggi. 2013. An affine invariant linear convergence analysis for Frank-Wolfe algorithms. *arXiv preprint arXiv:1312.7864* (2013).
- Simon Lacoste-Julien and Martin Jaggi. 2015. On the global linear convergence of Frank-Wolfe optimization variants. In *Advances in Neural Information Processing Systems*. 496–504.
- Charles L. Lawson and Richard J. Hanson. 1995. *Solving least squares problems*. Classics in Applied Mathematics, Vol. 15. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA. xiii+337 pages. <https://doi.org/10.1137/1.9781611971217> Revised reprint of the 1974 original.
- Kiyohito Nagano, Yoshinobu Kawahara, and Kazuyuki Aihara. 2011. Size-constrained submodular minimization through minimum norm base. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*. 977–984.
- Javier Peña, Daniel Rodríguez, and Negar Soheili. 2016. On the von Neumann and Frank-Wolfe algorithms with away steps. *SIAM J. Optim.* 26, 1 (2016), 499–512. <https://doi.org/10.1137/15M1009937>
- Alexander Schrijver. 1998. *Theory of Linear and Integer Programming*. John Wiley & Sons, Chichester, England.
- Steve Smale. 2000. Mathematical problems for the next century. *The Mathematical Intelligencer* 20, 2 (2000), 7–15.
- Tamás Terlaky and Shuzhong Zhang. 1993. Pivot rules for linear programming: A survey on recent theoretical developments. *Annals of Operations Research* 46, 1 (1993), 203–233. <https://doi.org/10.1007/BF02096264>
- László A. Végh. 2016. A Strongly Polynomial Algorithm for a Class of Minimum-Cost Flow Problems with Separable Convex Objectives. *SIAM J. Comput.* 45, 5 (2016), 1729–1761. <https://doi.org/10.1137/140978296>
- Philip Wolfe. 1974. Algorithm for a least-distance programming problem. *Math. Programming Stud.* 1 (1974), 190–205. Pivoting and extensions.
- Philip Wolfe. 1976. Finding the nearest point in a polytope. *Mathematical Programming* 11, 1 (1976), 128–149.