Deformed Products and Maximal Shadows of Polytopes

Nina Amenta*
Computer Sciences Department
University of Texas
Austin, TX 78712, USA
`amenta@cs.utexas.edu`
`http://www.cs.utexas.edu/users/amenta`

Günter M. Ziegler**
Dept. Mathematics 7-1
Technische Universität Berlin
10623 Berlin, Germany
`ziegler@math.tu-berlin.de`
`http://www.math.tu-berlin.de/~ziegler`

Abstract

We present a construction of *deformed products* of polytopes that has as special cases all the known constructions of linear programs with "many pivots," starting with the famous Klee-Minty cubes from 1972.

Thus we obtain sharp estimates for the following geometric quantities for d-dimensional simple polytopes with at most n facets:

- the maximal number of vertices on an increasing path,
- the maximal number of vertices on a "greedy" greatest increase path, and
- the maximal number of vertices of a 2-dimensional projection.

This, equivalently, provides good estimates for the worst-case behaviour of the simplex algorithm on linear programs with these parameters with the worst-possible, the greatest increase, and the shadow vertex pivot rules.

The bounds on the maximal number of vertices on an increasing path or a greatest increase path unify and slightly improve a number of known results. The bound on the maximal number of vertices of a 2-dimensional projection is new: we show that a 2-dimensional projection of a d-dimensional polytope with n facets may have as many as $\Theta(n^{\lfloor d/2 \rfloor})$ vertices for fixed d. This provides the same bound for the worst-case behaviour of the simplex algorithm with the shadow vertex pivot rule. The maximal complexity of shadows in fixed dimension is also relevant for problems of Computational Geometry. We give a new algorithm for the construction of the shadow of a d-dimensional polytope.

However, we find that for even $d \geq 4$ the polars of cyclic polytopes, $C_d(n)^\Delta$, which have the maximal number of vertices for any given n, do not maximize the shadow: for example, any 2-dimensional projection of $C_4(n)^\Delta$ has not more than $3n$ vertices.

*Most of this work was done at The Geometry Center, supported by NSF/DMS-8920161, at the Freie Universität Berlin, supported by the DAAD, and at Xerox PARC, partially supported by NSF/CCR-9404113

**Supported by a DFG Gerhard-Hess-Forschungsförderungspreis (Zi 475/1-1) and by the German-Israeli Foundation grant I-0309-146.06/93.

1 Introduction

In 1965, Victor Klee wrote:

> "Before attacking a linear programming problem with the simplex algorithm, it can be very comforting to have a good estimate of the number of iterations which may be required in order to reach a solution. [...] In addition to the primary interest in the expected number of iterations, there is a strong secondary interest in the *maximum number* of iterations for the problem of a given size. Here [...] the still unsolved problem of determining the maximum number has been of central interest since the inception of the simplex method." (Klee [26, p. 313])

In other words, Klee asked for the largest possible number $H_{\mathrm{Da}}(d,n)$ of bases that might be visited by the simplex algorithm (with Dantzig's largest coefficient rule) on a linear programming problem that defines a (simple, bounded) d-dimensional polytope with at most n facets. This problem is still not solved. However, obvious upper bounds are given by

$$H_{\mathrm{Da}}(d,n) \quad \leq \quad H(d,n) \quad \leq \quad M(d,n), \tag{1}$$

where $H(d,n)$ is the maximal number of vertices in an increasing path on such a polytope, and $M(d,n)$ is the maximal number of vertices that such a polytope may have.

Quite a different line of study, questions of Computational Geometry that arise, for example, in Robotics, lead one to ask for the maximal number of vertices that the k-dimensional projection ("shadow") of a d-dimensional polytope with n facets may have. If the maximal quantity is denoted by $H_{k\text{-sh}}(d,n)$, then again it is clear that one has upper bounds given by

$$H_{2\text{-sh}}(d,n) \quad \leq \quad H_{k\text{-sh}}(d,n) \quad \leq \quad M(d,n). \tag{2}$$

This also provides an upper bound for the worst-case behaviour of the simplex algorithm with the shadow vertex pivot rule, which is also given by $H_{2\text{-sh}}(d,n)$. Thus it complements Borgwardt's (much lower) linear bounds for the average case behaviour of this pivot rule.

Already in 1957, T. S. Motzkin had formulated that $M(d,n)$ is given by the number of vertices of the polars (or duals) of cyclic polytopes $C_d(n)$; this conjecture, which became known as the Upper Bound Conjecture, was proved in 1970 by McMullen.

Theorem 1.1 (Upper Bound Theorem, McMullen [31] [40, Sect. 8.4]**)**
A d-dimensional polytope with n facets has no more than

$$M(d,n) \quad := \quad \binom{n-\lceil\frac{d}{2}\rceil}{\lfloor\frac{d}{2}\rfloor} + \binom{n-1-\lceil\frac{d-1}{2}\rceil}{\lfloor\frac{d-1}{2}\rfloor} \quad = \quad \begin{cases} \frac{n}{n-k}\binom{n-k}{k} & \text{for } d=2k \text{ even,} \\ 2\binom{n-k-1}{k} & \text{for } d=2k+1 \text{ odd.} \end{cases}$$

vertices, where equality is attained by all polars of neighborly polytopes and only by those (in particular, by the polars of cyclic polytopes).

This upper bound is a polynomial in n of degree $\lfloor\frac{d}{2}\rfloor$ in the case of *constant dimension*:

$$M(d,n) = O(n^{\lfloor\frac{d}{2}\rfloor}) \qquad \text{for fixed } d.$$

It is exponential in the *diagonal case* (for constant n/d); for example, for $n = 2d$ it grows like

$$M(d,2d) \approx \left(\tfrac{27}{4}\right)^{\lfloor d/2\rfloor}$$

The Upper Bound Theorem does not, however, answer the original questions: it only provides a (large) upper bound. Klee & Minty's classic 1972 paper [27] established that the quantities $H_{\mathrm{Da}}(d,n)$ and $H(d,n)$, at least asymptotically, grow as badly as $M(d,n)$ — both in the *diagonal case* where n/d is constant, and in the case of *constant dimension.*

Theorem 1.2 (Long increasing paths, Klee & Minty [27]**)**
Increasing paths, even those followed by the simplex algorithm with Dantzig's rule, on a d-dimensional polytope with $2d$ facets may visit as many as

$$2^d \quad \leq \quad H_{\mathrm{Da}}(d,2d) \quad \leq \quad M(d,2d)$$

distinct vertices (that is, exponentially many in the diagonal case), and they have a bound

$$\frac{1}{2^{\lfloor d/2 \rfloor^2}} \quad \leq \quad \liminf_{n\longrightarrow\infty} \frac{H_{\mathrm{Da}}(d,n)}{n^{\lfloor \frac{d}{2} \rfloor}} \quad \leq \quad \liminf_{n\longrightarrow\infty} \frac{M(d,n)}{n^{\lfloor \frac{d}{2} \rfloor}}$$

for the case of constant dimension d.

Analogous results were subsequently obtained by various authors, in particular

- by Jeroslow [24] and Blair [6] for the greatest increase rule,
- by Goldfarb & Sit [20] for the diagonal case of the steepest increase rule,
- by Avis & Chvátal [4] for the diagonal case of Bland's least index rule,
- and by Murty [32] and Goldfarb [18, 19] for the diagonal case of the shadow vertex (Gass-Saaty) rule.

It turns out that *all* these "bad examples" were constructed by (clever) variations of the Klee-Minty method of "tilting facets" or of "deforming products":

> "Then Q is obtained from the prism $[0,1] \times P$ by tilting the left base $\{0\} \times P$ in one way and the right base $\{1\} \times P$ in the opposite way." (Klee & Minty [27, p. 162])

> "the main difficulty to overcome is to insure that the deformed polytope is combinatorially equivalent to $V \times P$." (Jeroslow [24, p. 370])

In view of this, it is quite surprising that a proper definition of "deformed products" apparently has never been given. The main point of the present article is to provide such a definition — see Section 3 — and to show that it subsumes *all* the main constructions of linear programs on which the simplex algorithm with various pivot rules needs an exponential number of steps: they *are* deformed products. In the process of recreating these "old examples" as deformed products, we also sharpen the analysis, obtaining somewhat better lower bounds than were previously available (Section 4).

As a new result, we produce, also as deformed products, d-dimensional polytopes for which some 2-dimensional projection has as many as $H_{2\text{-sh}}(d,n) = \Theta(n^{\lfloor d/2 \rfloor})$ vertices, for fixed d. In Section 5.2 we also outline the relevance of this for questions of Computational Geometry. The computation of k-shadows is a very natural problem also because it interpolates between linear programming and the convex hull problem, where linear programming is concerned with computing a 1-shadow, while a convex hull computation determines the d-shadow.

As a surprise in the investigation of shadows we find that the 2-shadows of the polars $C_d(n)^{\Delta}$ of cyclic polytopes have $O(n^{\lfloor (d-1)/2 \rfloor})$ vertices. Thus, in even dimensions, the polars of cyclic polytopes, which provide the maximal numbers $M(d,n)$ of vertices for any given n, do not maximize the shadow. This suggests that the answer to Klee's initial question will not be quite as simple as stating $H_{\mathrm{Da}}(d,n) = M(d,n)$ for all $n > d \geq 0$, although that has not been disproved.

2 Some Preliminaries

2.1 Polytopes

A *polytope* is a finite intersection of closed half-spaces in $\mathbb{R}^d$, that is, a bounded set of the form $P = \{x \in \mathbb{R}^d\colon Ax \le b\}$ with $b \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n\times d}$, where $Ax \le b$ is the matrix notation for a system of n linear inequalities. Equivalently [40, Chap. 1], a polytope is the convex hull of a finite set of points in $\mathbb{R}^d$, written $P = \mathrm{conv}\{p_1, \dots, p_m\}$ for points $p_i \in \mathbb{R}^d$. The *dimension* of a polytope is the dimension of its affine hull. A *d-polytope* is a polytope of dimension d.

An inequality "$a^t x \le \alpha$" (where a and x are column vectors in $\mathbb{R}^d$, a^t is the transpose of a, and α is a scalar) is *valid* for $P \subseteq \mathbb{R}^d$ if it is satisfied for all $x \in P$. The *faces* of P are all the subsets of the form $F = \{x \in P\colon a^t x = \alpha\}$, where $a^t x \le \alpha$ is a valid inequality. The faces of a polytope are themselves polytopes. (This includes $\emptyset$ and P as the *trivial* faces of P.) The faces of a d-polytope of dimensions 0, 1, $d-2$ and $d-1$ are called the *vertices*, *edges*, *ridges* and *facets* of P, respectively. A *k-face* is a k-dimensional face.

Lemma 2.1 *Let $P = \mathrm{conv}\{p_i, \dots, p_m\} \subseteq \mathbb{R}^d$. Then for any $I \subseteq \{1, \dots, m\}$, the subset $F := \mathrm{conv}\{p_i : i \in I\}$ is a face of P if and only if there is a linear inequality $a^t x \le \alpha$ such that $a^t p_i \le \alpha$ holds for all i, with equality if and only if $i \in I$.*

Proof. See e.g. [40, Prop. 2.3]. □

A d-polytope is *simple* if every vertex lies on exactly d facets, or (equivalently) on exactly d edges. A *polar* of a d-polytope $P \subseteq \mathbb{R}^d$ is given by

$$P^\Delta \quad := \quad \{a \in \mathbb{R}^d\colon a^t x \le 1 \text{ for all } x \in P'\},$$

where P' denotes a translate of P that has the origin 0 in its interior. Polars of simple polytopes are *simplicial*, that is, all their proper faces are simplices.

In the following, a polytope *of type* (d, n) is a simple d-dimensional polytope with at most n facets. We will usually write $n > d \ge 0$, where it is understood that the parameters n and d satisfy $n > d > 1$, or $(d, n) = (1, 2)$, or $(d, n) = (0, 0)$. We refer, for example, to [40] for other basic combinatorial properties of convex polytopes.

Definition 2.2 (Combinatorially equivalent polytopes)
Two polytopes P and Q are *combinatorially equivalent* if there is a bijection between their vertex sets $\mathrm{vert}(P) = \{p_1, \dots, p_m\}$ and $\mathrm{vert}(Q) = \{q_1, \dots, q_m\}$ such that for any subset $I \subseteq \{1, \dots, m\}$, the convex hull $\mathrm{conv}\{p_i : i \in I\}$ is a face of P if and only if the convex hull $\mathrm{conv}\{q_i : i \in I\}$ is a face of Q.

Definition 2.3 (Normally equivalent polytopes)
Two combinatorially equivalent d-polytopes $P, Q \subseteq \mathbb{R}^d$ are *normally equivalent* if additionally the unit facet normals of their corresponding facets coincide, that is, if each facet $\mathrm{conv}\{p_i : i \in I\}$ of P is parallel to the corresponding facet $\mathrm{conv}\{q_i : i \in I\}$ of Q.

An Isomorphism Lemma of the type below was already used by Klee & Minty [27, p. 167] [24, p. 371] [20, p. 278]. Our version adds a dimension condition (which is essential: otherwise see [40, p. 71] for a counterexample!), but slightly weakens the conditions otherwise.

Lemma 2.4 (Isomorphism Lemma)
Let $P = \mathrm{conv}\{p_1, \dots, p_m\}$ *and* $Q = \mathrm{conv}\{q_1 \dots, q_m\}$ *be two polytopes with* $\dim(Q) \le \dim(P)$ *such that for each index set* $I \subseteq \{1, 2, \dots, m\}$ *one has*

$$\mathrm{conv}\{p_i \colon i \in I\} \text{ is a facet of } P \qquad \Longrightarrow \qquad \mathrm{conv}\{q_i \colon i \in I\} \text{ is a face of } Q.$$

Then the correspondence $p_i \longleftrightarrow q_i$ *defines a combinatorial isomorphism between* P *and* Q, *that is,* $\mathrm{conv}\{p_i \colon i \in I\}$ *is a face (resp. facet) of* P if and only if $\mathrm{conv}\{q_i \colon i \in I\}$ *is a face (resp. facet) of* Q.

Proof. The faces are exactly the intersections of facets. Hence

$$\mathrm{conv}\{p_i \colon i \in I\} \text{ is a face of } P \qquad \Longrightarrow \qquad \mathrm{conv}\{q_i \colon i \in I\} \text{ is a face of } Q.$$

Since the dimension of a face G of Q is the largest l such that there is a chain of faces of Q of the form $\emptyset \subset F_0 \subset F_1 \subset \dots \subset F_{l-1} \subset F_l = G$, this implies $\dim(Q) \ge \dim(P)$. Hence $\dim(P) = \dim(Q)$, and

$$\mathrm{conv}\{p_i \colon i \in I\} \text{ is a } k\text{-face of } P \qquad \Longrightarrow \qquad \mathrm{conv}\{q_i \colon i \in I\} \text{ is a } k\text{-face of } Q.$$

In particular, all facets of P correspond to facets of Q.

Now we argue by induction on the dimension, where the situation is trivial for $\dim(P) = \dim(Q) \le 1$. Assume that some facets of Q are induced by P, some are not. Since the facet graph is connected, we can find adjacent facets G, G' of Q, with $G = \mathrm{conv}\{q_j : j \in J\}$ and $G' = \mathrm{conv}\{q_j : j \in J'\}$, such that $F := \mathrm{conv}\{p_j : j \in J\}$ is a facet of P, but $\mathrm{conv}\{p_j : j \in J'\}$ is not. Then the ridge

$$H := G \cap G' = \mathrm{conv}\{q_j : j \in J \cap J'\}$$

is a facet of G, hence (by induction applied to G) we know that $K := \mathrm{conv}\{p_j : j \in J \cap J'\}$ is a $(\dim(P) - 2)$-face of P, a facet of F. Let $F'' = \mathrm{conv}\{p_j : j \in J''\}$ be the second facet of P that contains K. Thus $G'' := \mathrm{conv}\{q_j : j \in J''\}$ is different from G and G', and it contains the ridge H: contradiction. $\square$

A *cyclic polytope* $C_d(n)$ is a polytope that is combinatorially equivalent to the polytope

$$\mathrm{conv}\{\gamma(1), \gamma(2), \dots, \gamma(n)\} \ \subseteq \ \mathbb{R}^d,$$

where $\gamma(t) := (t, t^2, \dots, t^d)$ denotes the *moment curve* in $\mathbb{R}^d$. The cyclic polytopes $C_d(n)$ are simplicial, and they are *neighborly*: any subset of $\lfloor \frac{d}{2} \rfloor$ of its vertices is the vertex set of a face.

The *product* of two polytopes $P \subseteq \mathbb{R}^d$ and $Q \subseteq \mathbb{R}^e$ is given by

$$P \times Q \quad = \quad \left\{ \begin{pmatrix} x \\ u \end{pmatrix} \colon \begin{array}{c} x \in P \\ u \in Q \end{array} \right\}.$$

The vertex set of a product is given by

$$\mathrm{vert}(P \times Q) \quad = \quad \left\{ \begin{pmatrix} p_i \\ q_j \end{pmatrix} \colon \begin{array}{c} p_i \in \mathrm{vert}(P) \\ q_j \in \mathrm{vert}(Q) \end{array} \right\}$$

while the facet-defining inequalities for $P \times Q$ are just the inequalities of P together with those of Q. (Thus products of simple polytopes are simple as well.) Here $\binom{x}{u}$ denotes the concatenation of the column vectors x and u, and $\mathrm{vert}(P)$ is the set of vertices of P. Thus taking products of polytopes the numbers of vertices are multiplied, while the numbers of facets are just added.

Example 2.5 (Polytopes with many vertices)
Polytopes with "many vertices and few facets" can be constructed by taking products of 2-polytopes, as follows.

If d is even, assume that n is a multiple of $\frac{d}{2}$. Then the product of $\frac{d}{2}$ copies of a $(\frac{2n}{d})$-gon $C_2(\frac{2n}{d})$ is a simple d-polytope $C_2(\frac{2n}{d})^{d/2}$ with n facets and $(\frac{2n}{d})^{d/2}$ vertices.

If d is odd, assume that $n-2$ is a multiple of $\frac{d-1}{2}$. Then the product of an interval I with $\frac{d-1}{2}$ copies of a $(\frac{2(n-2)}{d-1})$-gon $C_2(\frac{2(n-2)}{d-1})$ is a simple d-polytope $I \times C_2(\frac{2(n-2)}{d-1})^{(d-1)/2}$ with n facets and $2(\frac{2(n-2)}{d-1})^{(d-1)/2}$ vertices.

For instance, for $n = 2d$ the two constructions in Example 2.5 yield the d-cubes C_d: simple polytopes of type $(d, 2d)$, with 2^d vertices. Similarly, we get an exponential number of vertices (in d) if n is any larger fixed multiple of d.

Although for fixed dimension d the products in Example 2.5 have $O(n^{\lfloor d/2 \rfloor})$ vertices, equality for the upper bound theorem is attained only for the polar polytopes of neighborly polytopes.

2.2 Linear Programming

In geometric terms, linear programming (via the simplex algorithm) deals with edge paths on polyhedra that are strictly increasing with respect to some linear function $c\colon \mathbb{R}^d \longrightarrow \mathbb{R}$. Thus we study linear programming problems of the type

$$\max c^t x: \quad Ax \le b,$$

where $c \in \mathbb{R}^d$, $b \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n \times d}$. Writing $a_1^t, \ldots, a_n^t$ for the rows of A, the i-th inequality of the system is given by $a_i^t x \le b_i$. The assignment $x \longmapsto c^t x$ represents a linear objective function. Our constructions below maintain all the following nondegeneracy assumptions, although none of them is essential for what we do.

Feasibility: P is non-empty.

Full dimension: $P := \{x \in \mathbb{R}^d \colon Ax \le b\}$ is a d-dimensional polyhedron with n (distinct) facets (that is, each of the inequalities in the system $Ax \le b$ defines a facet of P, and all these facets are distinct).

Boundedness: P is a polytope. Hence $c^t x$ is bounded on P for *every* choice of $c \in \mathbb{R}^d$.

Phase I: A start vertex is given. Usually, we furthermore assume that the start vertex is the vertex of P that *minimizes* the objective function.

Primal nondegeneracy: The polytope P is *simple*, that is, every vertex of P lies on exactly d facets (equivalently, every vertex is adjacent to exactly d edges).

Dual nondegeneracy: No two vertices have the same objective function value $c^t v$. In particular, there are no horizontal edges with respect to the objective function.

With the primal nondegeneracy assumption, a *vertex* of the linear program is a point $v \in \mathbb{R}^d$ that satisfies $a_i^t v = b_i$ for exactly d indices $i \in I(v) := \{i_1, i_2, \ldots, i_d\}_<$, while $a_i^t v < b_i$ holds for all $i \in \{1, 2, \ldots, n\} \backslash I(v)$.

The (geometric) simplex algorithm [40, Sect. 3.2] proceeds from any vertex v to a *better adjacent* vertex, that is, to a vertex v' such that $I(v)$ and $I(v')$ differ in exactly one element, and such that $c^t v < c^t v'$. A *pivot rule* determines which vertex to choose. Here we mostly concentrate on geometrically defined rules. We do not consider randomized pivot rules, although they are very interesting [25] [30]. The same applies for "affirmative action" pivot rules, where exponential lower bounds in the worst case are not available [39] [15]. We do consider the following deterministic and non-deterministic pivot rules.

Greatest increase rule: Choose the vertex v' that gives the greatest increase in the objective function, that is, such that $c^t v' - c^t v$ is maximal.

Dantzig's largest coefficient rule: Choose the vertex v' such that $I(v) \backslash I(v') = \{i\}$, where γ_i is the largest (positive!) coefficient in the (unique) representation of the objective function as $c^t x = c^t v + \sum_{i \in I(v)} \gamma_i a_i^t x$. (The coefficients γ_i in this representation are known as the *reduced costs* or *nonbasic gradients*. The positive reduced cost coefficients correspond to the legal pivots at v [26, Sect. I] [27, Sect. 7].)

Bland's least index rule: Choose v' such that the (unique) index in $I(v) \backslash I(v')$ is minimal, that is, choose the edge that leaves the facet with the smallest number (according to the fixed numbering of the facets resp. inequalities) [7].

The Gass-Saaty shadow vertex rule: For the starting vertex one constructs a linear functional $\widetilde{c} \in \mathbb{R}^d$ such that v_0 (uniquely) maximizes $\widetilde{c}^t x$ over P. (E. g., take $\widetilde{c} := \sum_{i \in I(v_0)} a_i$). Then given that v_{i-1} maximizes $(\widetilde{c} + \lambda c)^t x$ exactly for $\lambda \in [\lambda_{i-1}, \lambda_i]$, the next vertex v_i is chosen to maximize $(\widetilde{c} + \lambda c)^t x$ for all $\lambda \in [\lambda_i, \lambda_{i+1}]$. This is equivalent to the condition that v_i must maximize the ratio $\frac{\widetilde{c}^t (v_i - v_{i-1})}{c^t (v_i - v_{i-1})}$.
Thus, in geometric terms, the shadow vertex rule finds a path that projects (under $x \longmapsto (c^t x, \widetilde{c}^t x)$) to a path on the boundary of a 2-dimensional shadow [16] [8].

Observation 2.6 (A relation between Dantzig's and Bland's rules)
If we multiply the i-th inequality $a_i^t x \leq b_i$ in a linear program by δ^{1-i} for some $\delta > 0$, then a_i is replaced by $a_i' = \delta^{1-i} a_i$, b_i is replaced by $b_i' = \delta^{1-i} b_i$, the polytope and the objective function do not change at all, and the reduced costs are scaled by $\gamma_i' := \delta^{i-1} \gamma_i$. Hence, if we choose $\delta > 0$ small enough, then at every vertex we obtain $\gamma_i' > \gamma_j'$ whenever $i, j \in I(v)$ and $i < j$.

In other words, Dantzig's rule follows the same sequence of pivots that is chosen by Bland's rule, provided that we suitably scale the inequalities.

General theory [36] [22] implies that all the constructions in the following can be realized with small coefficients. Thus every time we exhibit for some pivot rule paths whose length is exponential in the parameters d and n, this also provides examples for which the number of steps is exponential if measured in the (bit) encoding length of the linear programs in question.

2.3 Some Basic Observations

Definition 2.7 (Basic counting functions)
For d-dimensional polytopes with at most n facets, and linear functions φ on them, we study and relate the following quantities:

$M(d, n)$: the maximal number of vertices,

$H(d, n)$: the maximal number of vertices on an increasing path,

$H_{\text{gi}}(d, n)$: the maximal number of vertices on a *reversible* greatest increase path, that is, the maximal number l such that the greatest increase paths with respect to φ and to $-\varphi$, starting at the vertices that minimize φ respectively $-\varphi$, both contain at least l vertices,

$H_{\text{Da}}(d, n)$: the maximal number of vertices on a reversible path for Dantzig's largest coefficient rule,

$H_{\mathrm{Bl}}(d,n)$: the maximal number of vertices on a reversible path for Bland's least index rule,

$H_{\text{2-sh}}(d,n)$: the maximal number of vertices on a 2-dimensional projection (a "shadow").

(Counting the vertices rather than the edges on monotone paths amounts to a shift of 1: this simplifies some of the formulas in the following.)

Lemma 2.8 *For all $n > d \geq 0$,*

$$\begin{aligned} H_{\mathrm{gi}}(d,n) \ &\leq\ H(d,n) \ \leq\ M(d,n) && (3)\\ H_{\mathrm{gi}}(d,n) + (d-1) \ &\leq\ M(d,n) && (4)\\ H_{\mathrm{Bl}}(d,n) \ \leq\ H_{\mathrm{Da}}(d,n) \ &\leq\ H(d,n) \ \leq\ M(d,n) && (5)\\ H_{\text{2-sh}}(d,n) \ &\leq\ H(d,n) \ \leq\ M(d,n) && (6) \end{aligned}$$

Proof. Inequality (4) follows from the observation that among the (at least) d neighbors of the starting vertex, the greatest increase rule will visit only one: the one which provides the greatest increase. Inequality (5) is a consequence of Observation 2.6. The first inequality in (6), on the other side, follows from the following lemma. □

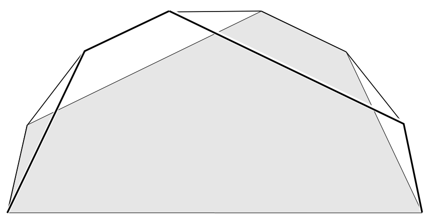
Lemma 2.9 *For all $n > d \geq 0$, $H_{\text{2-sh}}(d,n)$ is the maximal number of vertices on a simplex path according to the shadow vertex rule on a nondegenerate linear program of type (d,n).*

Proof. Every 2-dimensional k-gon Q is projectively equivalent to a k-gon Q' such that all k vertices lie on a path that is increasing with respect to a given linear function $\alpha : \mathbb{R}^2 \longrightarrow \mathbb{R}$. If Q is the shadow of a d-polytope P (under the projection to the last two coordinates, say), then the projective transformation extended to $\mathbb{R}^d$ provides us with P' such that Q' is the shadow of P'. Thus there are k vertices in an increasing edge path of Q', and hence also on a shadow vertex path of P'. □

A key question, which remains open in our study, is whether we really have $H(d,n) = M(d,n)$ for all $n > d \geq 0$, that is, whether the only upper bound on the length of monotone paths is restricted only by the Upper Bound Theorem, which bounds the total number of vertices. The answer is positive for $d \leq 3$. In fact, for $d = 3$ let $m := 2n-5$ and $k = 0, 1, \ldots, m$ and define

$$v_k \ := \ \begin{cases} (\cos(k\pi/m), \sin(k\pi/m), \sin(k\pi/m))^t & \text{if } k \equiv 0 \text{ or } k \equiv 1 \pmod 4, \\ (\cos(k\pi/m), \sin(k\pi/m), 0)^t & \text{if } k \equiv 2 \text{ or } k \equiv 3 \pmod 4. \end{cases}$$

This yields a simple polytope with n facets, whose projection to the first two coordinates is "half a regular $2m$-gon," with all its $m+1$ vertices on a semicircular arc. Our figure represents this for the case $m = 7$, $n = 6$.



The combinatorial type of the resulting 3-polytope is that of the polar of a cyclic polytope $C_3(n)$. Hence for $d = 3$, the maximum of $H_{\text{2-sh}}(3,n) = H(3,n)$ is achieved by the polar of a cyclic polytope. We will, however, see in Theorem 5.15 that the corresponding statement fails for $d = 4$. Thus it is not at all clear whether $H(d,n) = M(d,n)$ "deserves to be true" for $d > 3$. Nevertheless, we prove that the quantities $H_{\text{2-sh}}(d,n)$ and $H(d,n)$ have the same asymptotic growth for any fixed dimension d, in Theorem 5.3.

3 Deformed Products

Here is a simple basic version of the deformed products construction.

Definition 3.1 (Deformed products)
Let $P \subseteq \mathbb{R}^d$ be a convex polytope, and $\varphi\colon P \longrightarrow \mathbb{R}$ a linear functional with $\varphi(P) \subseteq [0,1]$. Let $V, W \subseteq \mathbb{R}^e$ be convex polytopes. Then the *deformed product* of (P,φ) and of (V,W) is

$$(P,\varphi) \bowtie (V,W) \quad := \quad \left\{ \begin{pmatrix} x \\ v + \varphi(x)(w-v) \end{pmatrix} : \begin{array}{c} x \in P \\ v \in V,\ w \in W \end{array} \right\} \quad \subseteq \quad \mathbb{R}^{d+e}.$$

Examples 3.2 (i) If $V = W$, then the deformed product is the "standard product":

$$(P,\varphi) \bowtie (V,V) \quad = \quad P \times V.$$

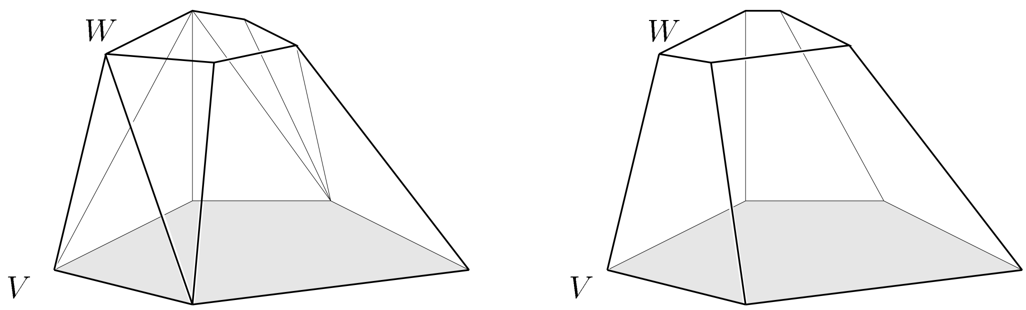
(ii) If φ is constant ($\varphi(x) = \lambda$ for all $x \in P$), then the deformed product is a standard product with an appropriate Minkowski sum:

$$(P,\varphi) \bowtie (V,W) \quad = \quad P \ \times \ (\lambda W + (1-\lambda)V).$$

(iii) If $P = Q \times [0,1]$ with $\varphi(q,t) = t$, then

$$(Q \times I, \varphi) \bowtie (V,W) \quad = \quad Q \times \operatorname{conv}((V \times \{0\}) \ \cup \ (W \times \{1\})).$$

This includes the case where $Q = \{0\}$ and $P = [0,1]$ with $\varphi(x) = x$, as illustrated in our figure. Here the left drawing depicts a "general" case, where for the right drawing V and W are normally equivalent, so the resulting polytope turns out to be (combinatorially equivalent to) a prism over V.



Lemma 3.3 *Deformed products are convex.*

Proof. Define $f(x,v,w) := \binom{x}{v+\varphi(x)(w-v)}$, and let $f(x',v',w')$ and $f(x'',v'',w'')$ be two points in the deformed product. We may restrict P to the line segment $[x',x'']$, and verify that the deformed product $([x',x''],\varphi) \bowtie (V,W)$ is convex.

Now if φ is constant on $[x',x'']$ (that is, if $\varphi(x') = \varphi(x'') = \lambda_0$), then we get that

$$([x',x''],\varphi) \bowtie (V,W) = [x',x''] \times (\lambda_0 W + (1-\lambda_0)V),$$

which is convex. (This is the situation of Example 3.2(ii).)

If φ is not constant, take $\lambda' := \varphi(x')$ and $\lambda'' := \varphi(x'')$ and define $V' := \lambda' W + (1-\lambda')V$ and $W'' := \lambda'' W + (1-\lambda'')V$. Both V' and W'' are convex. With this notation the map $\binom{x}{u} \longmapsto \binom{\varphi(x)}{u}$ provides us with an isomorphism

$$([x', x''], \varphi) \bowtie (V, W) \quad \cong \quad \mathrm{conv}((V' \times \{\lambda'\}) \ \cup \ (W'' \times \{\lambda''\})).$$

(This is the situation of Example 3.2(iii).) $\square$

We now show that a deformed product taken with a pair of *normally equivalent* polytopes is combinatorially equivalent to a regular (not deformed) product.

Theorem 3.4 (Vertices and facets of deformed products with normally equivalent polytopes)
Let $P \subseteq \mathbb{R}^d$ be a d-polytope, $\varphi\colon \mathbb{R}^d \longrightarrow \mathbb{R}$ a linear function with $\varphi(P) = [0,1]$, and let $V, W \subseteq \mathbb{R}^e$ be normally equivalent e-polytopes.

(i) *For $P = \mathrm{conv}\{p_1, \ldots, p_m\}$, $V = \mathrm{conv}\{v_1, \ldots, v_n\}$ and $W = \mathrm{conv}\{w_1, \ldots, w_n\}$, the deformed product is given by*

$$(P, \varphi) \bowtie (V, W) \quad = \quad \mathrm{conv}\left\{ \begin{pmatrix} p_i \\ v_j + \varphi(p_i)(w_j - v_j) \end{pmatrix} : \begin{array}{c} 1 \le i \le m \\ 1 \le j \le n \end{array} \right\}. \tag{7}$$

In particular, the deformed product is a polytope.

(ii) *If P, V and W are given by*

$$\begin{array}{rcll} P & = & \{x \in \mathbb{R}^d \colon a_k^t x \le \alpha_k \text{ for } 1 \le k \le s\}, & \\ V & = & \{u \in \mathbb{R}^e \colon b_l^t u \le \beta_l \text{ for } 1 \le l \le t\}, & \text{and} \\ W & = & \{u \in \mathbb{R}^e \colon b_l^t u \le \beta_l' \text{ for } 1 \le l \le t\}, & \end{array}$$

then the deformed product is given by

$$(P, \varphi) \bowtie (V, W) \ = \ \left\{ \begin{pmatrix} x \\ u \end{pmatrix} \in \mathbb{R}^{d+e} : \begin{array}{rll} (a_k^t, 0^t) \begin{pmatrix} x \\ u \end{pmatrix} \le \alpha_k & \text{for } 1 \le k \le s, \\ ((\beta_l - \beta_l')\varphi\,,\, b_l^t) \begin{pmatrix} x \\ u \end{pmatrix} \le \beta_l & \text{for } 1 \le l \le t. \end{array} \right\} \tag{8}$$

(iii) *The deformed product $(P, \varphi) \bowtie (V, W)$ is combinatorially equivalent to $P \times V$ and to $P \times W$. In particular, if P is a d-polytope with n vertices and s facets, and if V and W are e-polytopes with m vertices and t facets (that is, if the descriptions given in* (i) *and* (ii) *were irredundant), then $Q := (P, \varphi) \bowtie (V, W)$ is a $(d+e)$-polytope with nm vertices and $s+t$ facets, and the descriptions of Q are also irredundant: the points in the convex hull description* (7) *are the vertices, and the inequalities in* (8) *determine the facets of Q.*

Proof.

(i) Denote by $Q \subseteq \mathbb{R}^{d+e}$ the deformed product, by $\mathbb{R} \subseteq \mathbb{R}^{d+e}$ the right-hand-side of equation (7). Then clearly $Q \supseteq R$. Take an arbitrary linear functional $\ell = (\ell^1, \ell^2)\colon \mathbb{R}^{d+e} \longrightarrow \mathbb{R}$. The maximum of ℓ on Q is attained by some $\binom{x}{\varphi(x)w + (1-\varphi(x))v}$. Since $1 \ge \varphi(x) \ge 0$ the maximum is attained in vertices $w \in W$ and $v \in V$. Since W, V are normally equivalent the maximum of $\ell^2(\varphi(x)w + (1-\varphi(x))v)$, for fixed x, is attained by corresponding vertices v_j and w_j. Hence we may assume that $v = v_j$ and $w = w_j$ for some $1 \le j \le n$. Fixing those, we get that ℓ is linear in x, and hence the maximum of ℓ on Q is attained for some $x = p_i$, where $1 \le i \le m$. Thus the maximum of ℓ over Q is attained for some point of $\mathbb{R}$. Since $\mathbb{R}$ is convex, we conclude $Q \subseteq R$.

(ii) Let $U \subseteq \mathbb{R}^{d+e}$ denote the solution set of the inequality system in (8), that is, the right-hand side of the equation (8). First, a simple computation shows that the inequalities in (8) are valid for Q:

$$(a_k^t, 0^t)\begin{pmatrix} x \\ v + \varphi(x)(w-v) \end{pmatrix} \quad = \quad a_k^t x \quad \leq \quad \alpha_k$$

and

$$\begin{aligned} ((\beta_l - \beta_l')\varphi, b_l^t)\begin{pmatrix} x \\ v + \varphi(x)(w-v) \end{pmatrix} \quad &= \\ (\beta_l - \beta_l')\varphi(x) + b_l^t v + \varphi(x)(b_l^t w - b_l^t v) \quad &= \\ (1-\varphi(x))(b_l^t v - \beta_l) + \beta_l + \varphi(x)(b_l^t w - \beta_l) \quad &\leq \quad \beta_l \end{aligned}$$

follow easily from the corresponding inequalities for P, V and W. This implies $Q \subseteq U$. Furthermore, we have equality for a point of the form $\binom{p_i}{v_j + \varphi(p_i)(w_j - v_j)}$ in the first case if and only if $a_k^t p_i = \alpha_k$, and in the second case if and only if $b_l^t v_j = \beta_l$ and $b_l^t w_j = \beta_l'$: to see this, note that in case $\varphi(x) > 0$ we get $b_l v_j = \beta_l$, while in case $\varphi(x) < 1$ we get $b_l w_j = \beta_l'$. Finally, we know that the two conditions $b_l^t v_j = \beta_l$ and $b_l^t w_j = \beta_l'$ are equivalent.

At this point we invoke the Isomorphism Lemma 2.4, applied to $P \times V$ as the first polytope and the deformed product $(P, \varphi) \bowtie (V, W)$ as the second polytope. This shows us that the description of the deformed product in terms of inequalities (8) is complete, and it also establishes part (iii). $\square$

Remark 3.5 From now on we will maintain the assumptions of Theorem 3.4, namely that

- $P \subseteq \mathbb{R}^d$ is d-dimensional (although this is not essential),
- $\varphi(P) = [0, 1]$ (it is only important that $\varphi(P) \subseteq [0, 1]$: without this assumption the deformed product is not usually convex, as may be seen for $P = [0, 1] \subseteq \mathbb{R}^1$, $\varphi(x) = 2x$, $V = [0, 1]$, $W = \{0\}$),
- $V, W \subseteq \mathbb{R}^e$ are e-dimensional and normally equivalent.

Corollary 3.6 *Let $Q := (P, \varphi) \bowtie (V, W) \subseteq \mathbb{R}^{d+e}$ be a deformed product as above.*

(i) *The projection $\pi_1\colon \mathbb{R}^{d+e} \longrightarrow \mathbb{R}^d$ to the first d components maps Q to $\pi_1(Q) = P$.*

(ii) *The projection $\pi_2\colon \mathbb{R}^{d+e} \longrightarrow \mathbb{R}^e$ to the last e components maps Q to $\pi_2(Q) = \operatorname{conv}(V \cup W)$.*

Corollary 3.7 *The deformed product $(P, \varphi) \bowtie (V, W)$ is connected to the "standard" product $P \times V$ in the realization space of the product.*

Proof. Consider

$$(P, t\varphi) \bowtie (V, W) \quad := \quad \operatorname{conv}\left\{ \begin{pmatrix} p_i \\ u \end{pmatrix} : \begin{array}{l} p_i \in \operatorname{vert}(P), \\ u = v_j + t\varphi(p_i)(w_j - v_j), \ 1 \leq j \leq n \end{array} \right\}$$

for $0 \leq t \leq 1$. For $t = 0$ this realizes the standard product, for $t = 1$ one gets the deformed product. $\square$

Remark 3.8 R. Seidel has pointed out the following alternative way to view the construction of deformed products $(P,\varphi) \bowtie (V,W)$. For this again assume that $\varphi(P) = [0,1]$. If V and W are normally equivalent, then $\mathrm{conv}(V{\times}\{0\} \cup W{\times}\{1\}) \subseteq \mathbb{R}^{e+1}$ is combinatorially equivalent to $V \times I$ (as in Example 3.2(iii)). Hence

$$C \quad := \quad P \times \mathrm{conv}(V{\times}\{0\} \cup W{\times}\{1\}) \quad = \quad \mathrm{conv}(P \times V{\times}\{0\} \cup P \times W{\times}\{1\}) \quad \subseteq \quad \mathbb{R}^{d+e+1}$$

is combinatorially equivalent to $P \times V \times I$ (it is a "deformed prism" over $P \times V$), where

$$C \quad = \quad \left\{ \begin{pmatrix} x \\ u \\ t \end{pmatrix} : \quad \begin{array}{l} x \in P \\ u = v + t(w-v), \;\; v \in V, \;\; w \in W \\ t \in I \end{array} \right\}.$$

Now cut C with the $(d+e)$-dimensional hyperplane $\varphi(x) = t$, and project the intersection to $\mathbb{R}^{d+e}$ by deleting the last coordinate. Call the image S. We get

$$S \quad = \quad \left\{ \begin{pmatrix} x \\ u \end{pmatrix} : \quad \begin{array}{l} x \in P \\ u = v + \varphi(x)(w-v), \;\; v \in V, \;\; w \in W \end{array} \right\},$$

which is the deformed product. S is a convex polytope. It is combinatorially equivalent to $P \times V$ or $P \times W$, since it is a slice of a (combinatorial) prism that has the top on one side and the bottom on the other side.

Definition 3.9 (Increasing edges)
If p', p'' are adjacent vertices of a polytope $P \subseteq \mathbb{R}^d$, then we write $[p', p'']$ for the edge between p' and p''. We say that $[p', p'']$ is *α-increasing* if $\alpha(p'') > \alpha(p')$ for a linear function $\alpha\colon \mathbb{R}^d \longrightarrow \mathbb{R}$. (The ordering of the vertices p', p'' is relevant in this notation.)

Definition 3.10 (Deformed product programs)
With the notation and assumptions of Theorem 3.4, let

$$\max \varphi(x)\colon \quad x \in P$$

be a linear program in $\mathbb{R}^d$, and let

$$\max \alpha(u)\colon \;\; u \in V \qquad \text{resp.} \qquad \max \alpha(u) : \;\; u \in W$$

be two normally equivalent linear programs (that is, one arises from the other by changing the right hand side, without change of the combinatorial type).

The linear function $\alpha\colon \mathbb{R}^e \longrightarrow \mathbb{R}$ induces a linear function $\widehat{\alpha}\colon \mathbb{R}^{d+e} \longrightarrow \mathbb{R}$ via $\widehat{\alpha}\binom{x}{u} := \alpha(u)$. Using this function, the *deformed product program* is given by

$$\max \widehat{\alpha}\begin{pmatrix} x \\ u \end{pmatrix} = \alpha(u)\colon \quad \begin{pmatrix} x \\ u \end{pmatrix} \in Q = (P,\varphi) \bowtie (V,W).$$

For the deformed product $Q = (P,\varphi) \bowtie (V,W)$ of Theorem 3.4 use the notation

$$v(i,j) \quad := \quad \begin{pmatrix} p_i \\ v_j + \varphi(p_i)(w_j - v_j) \end{pmatrix}$$

for the vertices ($1 \le i \le m$, $1 \le j \le n$) of Q. By Theorem 3.4(iii) the edges of a deformed product just correspond to those of the regular product. Thus Q has two types of edges:

- edges of the form $[v(i',j),v(i'',j)]$ (called *P-edges*) for $1 \le j \le n$ and for any $1 \le i',i'' \le m$ such that $p_{i'}$ and $p_{i''}$ are adjacent vertices of P, and
- edges of the form $[v(i,j'),v(i,j'')]$ (named *(V,W)-edges*) for any $1 \le i \le m$ and for $1 \le j',j'' \le n$ such that $v_{j'}$ and $v_{j''}$ are adjacent vertices of V (equivalently, $w_{j'}$ and $w_{j''}$ are adjacent vertices of W).

Proposition 3.11 (Increasing edges of deformed products)
For any deformed product program $\max \widehat{\alpha}\binom{x}{u} = \alpha(u)\colon \binom{x}{u} \in Q = (P,\varphi) \bowtie (V,W)$,

- *a P-edge $[v(i',j),v(i'',j)]$ is $\widehat{\alpha}$-increasing if and only if*
  either $[p_{i'},p_{i''}]$ is φ-increasing and $\alpha(w_j) > \alpha(v_j)$,
  or $[p_{i'},p_{i''}]$ is φ-decreasing and $\alpha(w_j) < \alpha(v_j)$,
- *a (V,W)-edge $[v(i,j'),v(i,j'')]$ is $\widehat{\alpha}$-increasing if and only if $[v_{j'},v_{j''}]$ is α-increasing.*

Proof. We compute for a P-edge that

$$\begin{array}{rcl} & [v(i',j),v(i'',j)] \text{ is } \widehat{\alpha}\text{-increasing} & \iff \\ \alpha(v_j+\varphi(p_{i''})(w_j-v_j)) \;-\; \alpha(v_j+\varphi(p_{i'})(w_j-v_j)) & > 0 & \iff \\ (\varphi(p_{i''})-\varphi(p_{i'}))(\alpha(w_j)-\alpha(v_j)) & > 0 & \end{array}$$

which yields the claim. For a (V,W)-edge, we get

$$\begin{array}{rcl} & [v(i,j'),v(i,j'')] \text{ is } \widehat{\alpha}\text{-increasing} & \iff \\ \alpha(v_{j''}+\varphi(p_i)(w_{j''}-v_{j''})) \;-\; \alpha(v_{j'}+\varphi(p_i)(w_{j'}-v_{j'})) & > 0 & \iff \\ \varphi(p_i)(w_{j''}-w_{j'}) \;+\; (1-\varphi(p_i))(v_{j''}-v_{j'}) & > 0 & \end{array}$$

which implies the claim since $[v_{j'},v_{j''}]$ is α-increasing if and only if $[w_{j'},w_{j''}]$ is α-increasing. $\square$

Corollary 3.12 (Bland's rule on deformed product programs)
Let p_1 and p_m be the vertices of P that minimize resp. maximize φ, with $\varphi(p_1) = 0$ and $\varphi(p_m) = 1$. On the deformed product program $\max \widehat{\alpha}\binom{x}{u}\colon \binom{x}{u} \in (P,\varphi) \bowtie (V,W)$, *according to* Definition 3.10, *we have the following behavior.*

If the inequalities of Q are numbered such that all the inequalities of P get smaller indexes than all the inequalities corresponding to (V,W), then the simplex algorithm with Bland's rule prefers P-edges over (V,W)-edges.

Hence, if Bland's rule on P for the objective function φ takes a path with k vertices from p_1 to p_m, and for $-\varphi$ takes a path with k' vertices from p_m to p_1, then for $(Q,\widehat{\alpha})$ it will follow a path with k vertices from $v(1,j)$ to $v(m,j)$ if $\alpha(v_j) < \alpha(w_j)$, and a path with k' vertices from $v(m,j)$ to $v(1,j)$ if $\alpha(v_j) > \alpha(w_j)$.

Analogous results are easy to derive for many other pivot rules — see the examples in the next section. This is the basis for virtually *all* known constructions of classes linear programs on which variants of the simplex algorithm are exponential.

4 Some (Old) Examples

4.1 The Klee-Minty Cubes

The Klee-Minty cubes [27, Sect. 4] are classical: they can be found in any standard text on linear programming, in Schrijver [36], in Chvátal [12], and, reluctantly, in Padberg [33, Sect. 5.7.1].

$$\max x_d\colon\ x \in \mathrm{KM}_d$$

where KM_d is, for some ε with $0 < \varepsilon < \frac{1}{2}$, given by the inequality system

$$\begin{array}{rcccl} 0 & \leq & x_1 & \leq & 1 \\ \varepsilon x_{j-1} & \leq & x_j & \leq & 1 - \varepsilon x_{j-1} \text{ for } 2 \leq j \leq d \end{array}$$

As deformed products, we obtain the Klee-Minty cubes recursively as

$$\begin{array}{lcll} \mathrm{KM}_0 & = & \{0\} & \subseteq \mathbb{R}^0 \\ \mathrm{KM}_1 & = & [0,1] & \subseteq \mathbb{R}^1 \\ \mathrm{KM}_{d+1} & = & (\mathrm{KM}_d, x_d) \bowtie ([0,1], [\varepsilon, 1-\varepsilon]) & \subseteq \mathbb{R}^{d+1}. \end{array}$$

The boundary case $\varepsilon = \frac{1}{2}$ is analyzed in Clausen [14]. It is not difficult to explicitly determine both vertex coordinates and the combinatorial structure of the Klee-Minty cubes, see Avis & Chvátal [4], Clausen [13], and Gärtner & Ziegler [17]. However, this is not necessary for us, since the basic extremal properties follow already from the construction as deformed products.

Theorem 4.1 (The Klee-Minty cubes)
Number the inequalities of KM_d *in exactly the order in which they appear in the system, that is, the* i*-th inequality is* $x_j \geq \varepsilon x_{j-1}$ *for* $i = 2j-1$ *and* $x_j \leq 1 - \varepsilon x_{j-1}$ *for* $i = 2j$*. Then the simplex algorithm with Bland's least index rule follows an* x_d*-increasing path through all* 2^d *vertices of the* d*-dimensional Klee-Minty cube* KM_d*.*

If we multiply the i*-th inequality by* δ^{1-i} *for some small enough* $\delta > 0$*, then the same is true for Dantzig's largest coefficient rule.*

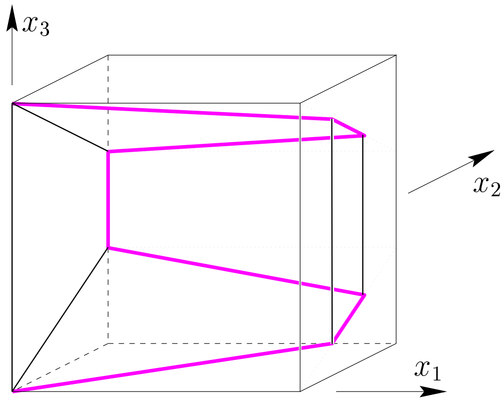
Proof. Take $V = \mathrm{conv}\{v_1, v_2\}$ with $v_1 = 0$, $v_2 = 1$ and $W = \mathrm{conv}\{w_1, w_2\}$ with $w_1 = \varepsilon$, $w_2 = 1 - \varepsilon$. The linear functional we use on KM_d is $\varphi(x) = x_d$, so we get $\widehat{\alpha}\binom{x}{u} = u = x_{d+1}$ for $\binom{x}{u} \in \mathbb{R}^{d+1}$. Among the 2^d vertices of KM_d we find $p_1 = 0$ and $p_{2^d} = e_d$.

By Corollary 3.12 and induction on d, we find that starting at $v(1,1) = 0$ Bland's rule on KM_{d+1} will take a path through all the 2^d vertices of the form $v(i,1)$, ending at $v(2^d,1) = \varepsilon e_{d+1}$. At this point all P-edges are x_{d+1}-decreasing, so the simplex algorithm takes the (unique) (V,W)-edge to $v(2^d,2) = (1-\varepsilon)e_{d+1}$. After that, Bland's rule takes a path through all the 2^d vertices $v(i,2)$, ending at $v(1,2) = e_{d+1}$. □

Avis & Chvátal [4] proved that if we take the i-th inequality to be $x_i \geq \varepsilon x_{i-1}$ for $i \leq d$ and to be $x_{i-d} \leq 1 - \varepsilon x_{i-d-1}$ for $i > d$, then the Bland's rule will follow a shorter path, whose number of vertices is, however, still exponential. By Observation 2.6, the same is true for the worst case of the Dantzig rule for this ordering of the inequalities.

Observation 4.2 The Klee-Minty cubes are *not* projectively equivalent to "standard" cubes, as was recently stated by Gritzmann & Klee [21, p. 646]. To see this, analyze the following picture, which represents a parallel projection of the 3-dimensional Klee-Minty cube (for $\varepsilon = \frac{1}{3}$).

If the Klee-Minty cube was a projective image of a regular cube, then the four "horizontal" edges would have to meet in one point (or be parallel), which is clearly not the case.



Theorem 4.3 *For $n > d \geq 0$:*

$$H_{\mathrm{Bl}}(d{+}1, n{+}2) \quad \geq \quad 2\, H_{\mathrm{Bl}}(d, n).$$

Analogous inequalities are true for the maximal numbers of vertices $H(d, n)$ and $H_{\mathrm{Da}}(d, n)$ on arbitrary increasing paths resp. for paths according to Dantzig's rule. Hence for $d \geq 0$ we have

$$H(d, 2d) \quad \geq \quad H_{\mathrm{Da}}(d, 2d) \quad \geq \quad H_{\mathrm{Bl}}(d, 2d) \quad \geq \quad 2^d.$$

Proof. Let $P \subseteq \mathbb{R}^d$ be a simple (d, n)-polytope which achieves the maximum length for an arbitrary increasing path (respectively of a path according to Bland's or Dantzig's rule), and let $\varphi : P \longrightarrow [0, 1]$ be a corresponding objective function. Then for some $0 < \varepsilon < \frac{1}{2}$ form the deformed product

$$(P, \varphi) \bowtie ([0, 1], [\varepsilon, 1{-}\varepsilon])$$

and use the same proof as for the Klee-Minty cubes. □

4.2 The Goldfarb-Sit Cubes

Goldfarb & Sit [20] constructed linear programs — rescaled Klee-Minty cubes tailored to fool the steepest increase rule, see also Clausen [13] — as follows. They analyzed the programs

$$\max \sum_{i=1}^{d} \beta^{i-1} x_i \colon \ x \in \mathrm{GS}_d$$

where $\mathrm{GS}_d \subseteq \mathbb{R}^d$ is the polytope given by

$$\begin{array}{rcccl} 0 & \leq & x_1 & \leq & 1 \\ \beta x_{j-1} & \leq & x_j & \leq & \delta_j - \beta x_{j-1} \quad \text{for } 2 \leq j \leq d, \end{array}$$

where $\beta \geq 2$, $\theta > 2$ and $\delta_i := (\theta\beta)^{i-1}$. It is immediate from Theorem 3.4 that we may construct the *Goldfarb-Sit cubes* as deformed products

$$\begin{array}{lcll} \mathrm{GS}_0 & = & \{0\} & \subseteq \mathbb{R}^0 \\ \mathrm{GS}_1 & = & [0, 1] & \subseteq \mathbb{R}^1 \\ \mathrm{GS}_{d+1} & = & (\mathrm{GS}_d, x_d) \bowtie ([0, \delta_{d+1}], [\beta, \delta_{d+1} - \beta]) & \subseteq \mathbb{R}^{d+1}. \end{array}$$

and thus that they are combinatorial d-cubes.

4.3 The Goldfarb Cubes

Murty [32] was the first to construct (in the framework of linear complementarity problems) linear programs of type $(d, 2d)$ that are exponential for the shadow vertex algorithm. Essentially equivalent, but much more explicit examples (at least from a linear programming point of view) were given by Goldfarb [18, 19]. Goldfarb's programs are

$$\max \xi_d\colon \quad \xi \in \mathrm{Gol}_d$$

where the polytopes Gol_d are given by

$$\begin{array}{rcccl} 0 & \le & \xi_1 & \le & 1 \\ \beta\xi_1 & \le & \xi_2 & \le & \delta - \beta\xi_1 \\ \beta\xi_j - \xi_{j-1} & \le & \xi_{j+1} & \le & \delta^j - \beta\xi_j + \xi_{j-1} \text{ for } 2 \le j < d \end{array}$$

with $\beta \ge 2$ and $\delta > 2\beta$. They are deformed products! The "additional rotation" that Goldfarb talks about [18, p. 2] is represented by a different functional φ:

$$\begin{array}{lcll} \mathrm{Gol}_0 & = & \{0\} & \subseteq \mathbb{R}^0 \\ \mathrm{Gol}_1 & = & [0,1] & \subseteq \mathbb{R}^1 \\ \mathrm{Gol}_{d+1} & = & (\mathrm{Gol}_d, \xi_d - \frac{1}{\beta}\xi_{d-1}) \bowtie ([0,\delta_d],[\beta,\delta_d - \beta]) & \subseteq \mathbb{R}^{d+1}. \end{array}$$

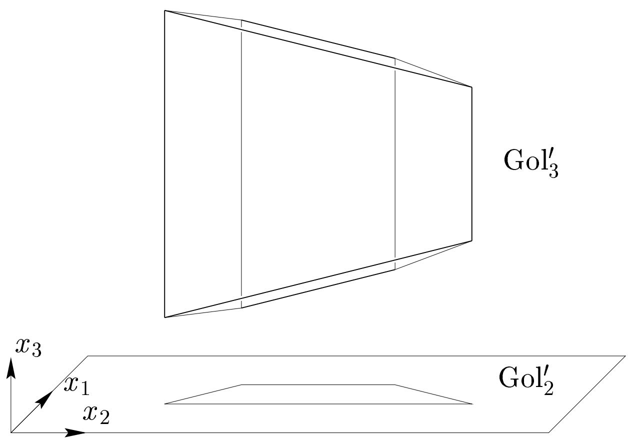
The β-δ-scaling in these examples seems, however, to be an artifact that remained from the previous work by Goldfarb & Sit. Setting $\varepsilon := \frac{\beta}{\delta} < \frac{1}{2}$, $\gamma := \frac{1}{\beta\delta} \le \frac{1}{4}\varepsilon$, and substituting $x_j := \frac{\xi_j}{\delta^{j-1}}$ we get

$$\max x_d\colon \quad x \in \mathrm{Gol}'_d$$

where the polytopes Gol'_d are given by:

$$\begin{array}{rcccl} 0 & \le & x_1 & \le & 1 \\ \varepsilon x_1 & \le & x_2 & \le & 1 - \varepsilon x_1 \\ \varepsilon(x_j - \gamma x_{j-1}) & \le & x_{j+1} & \le & 1 - \varepsilon(x_j - \gamma x_{j-1}) \text{ for } 2 \le j \le d \end{array}$$

which for $\gamma \longrightarrow 0$ yields the Klee-Minty cube (and for $\varepsilon, \gamma \longrightarrow 0$ yields the usual 0/1-cube).



Again, these polytopes are deformed products:

$$\begin{array}{lcll} \mathrm{Gol}'_0 & = & \{0\} & \subseteq \mathbb{R}^0 \\ \mathrm{Gol}'_1 & = & [0,1] & \subseteq \mathbb{R}^1 \\ \mathrm{Gol}'_{d+1} & = & (\mathrm{Gol}'_d, x_d - \gamma x_{d-1}) \bowtie ([0,1],[\varepsilon, 1-\varepsilon]) & \subseteq \mathbb{R}^{d+1} \end{array}$$

for $0 < 4\gamma < \varepsilon < \frac{1}{2}$.

This simpler formulation leads to a somewhat simpler proof.

Theorem 4.4 (Goldfarb [18, 19])
The projection π: $\mathrm{Gol}'_d \longrightarrow \mathbb{R}^2$ *given by* $\pi(x) = (x_{d-1}, x_d)$ *has* 2^d *vertices.*

Proof. Our proof follows [18]. For any vertex v of Gol'_d, let A_v be the $(d \times d)$-matrix whose rows are the facet normals at v:

$$A_v \;=\; \begin{pmatrix} \sigma_1 & & & & & & \\ \varepsilon & \sigma_2 & & & & & \\ -\varepsilon\gamma & \varepsilon & \sigma_3 & & & & \\ & -\varepsilon\gamma & \varepsilon & \sigma_4 & & & \\ & & -\varepsilon\gamma & \ddots & \ddots & & \\ & & & \ddots & \ddots & \sigma_{d-1} & \\ & & & & -\varepsilon\gamma & \varepsilon & \sigma_d \end{pmatrix} \;=:\; \begin{pmatrix} a_1^t \\ a_2^t \\ a_3^t \\ a_4^t \\ \vdots \\ \vdots \\ a_d^t \end{pmatrix}$$

where $\sigma_i = -1$ if the i-th inequality at v is "$\varepsilon(x_j - \gamma x_{j-1}) \le x_{j+1}$," and $\sigma_i = +1$ otherwise. A_v can be taken in this form since Gol'_d is combinatorially equivalent to a d-cube. We get that

$$A_v\, x \;\le\; A_v\, v$$

is a system of inequalities that is valid for all $x \in \mathrm{Gol}'_d$, with equality in all components if and only if $x = v$.

Now define a row vector $\alpha^t = (\alpha_1, \dots, \alpha_d)$ recursively by $\alpha_1 := 1$, $\alpha_2 := \frac{2}{\varepsilon}$ and $\alpha_{i+2} := \frac{1}{\gamma\varepsilon}(\varepsilon\alpha_{i+1} + \sigma_i\alpha_i)$. We verify $\alpha_{i+1} \ge \frac{2}{\varepsilon}\alpha_i > 0$ for all $i \ge 1$. This is true for $i = 1, 2$, and for $i \ge 3$ we get

$$\alpha_{i+2} \;\ge\; \frac{1}{\gamma\varepsilon}(\varepsilon\alpha_{i+1} - \alpha_i) \;\ge\; \frac{1}{\gamma\varepsilon}(\varepsilon\alpha_{i+1} - \frac{\varepsilon}{2}\alpha_{i+1}) \;=\; \frac{1}{2\gamma}\alpha_{i+1} \;\ge\; \frac{2}{\varepsilon}\alpha_{i+1} > 0,$$

using induction for the first and for the second inequality. Since α^t is a positive row vector, we deduce that

$$\alpha^t A_v\, x \;\le\; \alpha^t A_v\, v$$

holds for all $x \in \mathrm{Gol}'_d$, with equality if and only if $x = v$.

From $\alpha^t A_v = (0, 0, \dots, 0, \alpha_{d-1}, \alpha_d)$ we get

$$(\alpha_{d-1}, \alpha_d)\pi(x) \;=\; \alpha^t A_v\, x \;\le\; \alpha^t A_v\, v$$

with equality if and only if $x = v$: hence $\pi(v)$ is a vertex of $\pi(\mathrm{Gol}'_d)$, and v is the only point of Gol'_d that projects to $\pi(v)$, so all the 2^d vertices $\pi(v)$ of $\pi(\mathrm{Gol}'_d)$ are distinct. □

The condition $\gamma \le \frac{\varepsilon}{4}$ that we have used in the proof (equivalent to Goldfarb's $\beta \ge 2$) is not the best one could do, but it is the condition that works without problems for the proofs.

The sequence in which the shadow boundary algorithm visits the vertices of the Goldfarb cubes is the "same" as for the Klee-Minty cubes. Using that for $\gamma \to 0$ we get the Klee-Minty cubes from the Goldfarb cubes, this follows from the fact that the shadow vertex rule visits the vertices in increasing order with respect to the objective function $\varphi(x) = x_d$.

Corollary 4.5 *For* $d \ge 0$, $H_{2\text{-sh}}(d, 2d) \ge 2^d$.

4.4 The Klee-Minty Products

In view of the deformed product construction, it is not hard to construct polytopes with "many" vertices on increasing paths in fixed dimension [27, Sect. 5]: we iteratively construct deformed products with polygons (that is, we take $e = 2$). The single extra ingredient needed here — and in the following constructions — is a construction of suitable polygons $V, W \subseteq \mathbb{R}^2$. Again, this was first done by Klee and Minty; we replace their lemma [27, p. 165] by the following.

Lemma 4.6 *Let $\alpha(x)$ denote a linear function on the plane ($x \in \mathbb{R}^2$).*

For each $k \geq 4$, and for each $\lambda > 0$, there exist normally equivalent k-gons

$$V = \operatorname{conv}\{v_1, \ldots, v_k\} \qquad \text{and} \qquad W = \operatorname{conv}\{w_1, \ldots, w_k\}$$

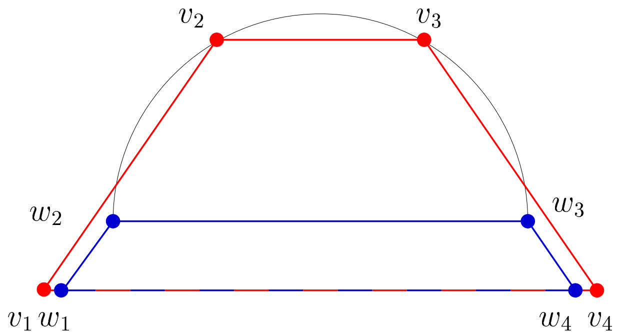
(both labeled in clockwise order), such that

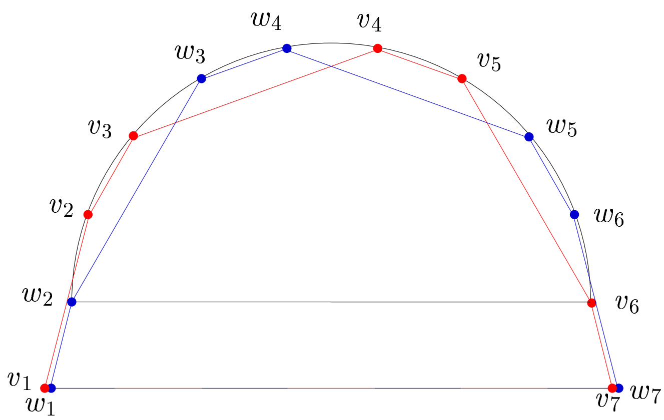
$$0 \;=\; \alpha(v_1) \;<\; \alpha(w_1) \;<\; \alpha(w_2) \;<\; \alpha(v_2) \;<\; \alpha(v_3) \;<\; \alpha(w_3) \;<\; \alpha(w_4) \;<\; \ldots$$

(The sequence ends with $\ldots < \alpha(v_k) < \alpha(w_k) = 1$ if k is odd, and with $\ldots < \alpha(w_k) < \alpha(v_k) = 1$ if k is even.)

Proof. Take $2k{-}4 \geq 4$ points equally spaced on a semicircular arc ("half a $2(2k{-}5)$-gon") and label them $w_2, v_2, v_3, w_3, w_4 \ldots$, ending with $\ldots, v_{k-1}, w_{k-1}$ if k is odd and with $\ldots, w_{k-1}, v_{k-1}$ if k is even. The additional points v_1, w_1, v_k and w_k are then chosen appropriately on a line that is parallel to the diameter on which the semicircular arc was based.

Our figures illustrate this in the cases $k = 4$ and $k = 7$.





Note: this lemma is false for $k = 3$. $\square$

Theorem 4.7 *For $k \geq 4$ and $n > d \geq 0$,*

$$H_{\mathrm{Bl}}(d{+}2, n{+}k) \quad \geq \quad k\, H_{\mathrm{Bl}}(d, n). \tag{9}$$

The same recursion is valid for $H(d, n)$ and for $H_{\mathrm{Da}}(d, n)$.

Proof. After rescaling the objective function and relabeling the vertices we may take a polytope P of type (d, n) for which Bland's rule, for a functional $c^t x = \varphi(x)$ with $\varphi(P) = [0, 1]$, follows an increasing path on $l := H_{\mathrm{Bl}}(d, n)$ vertices starting at the vertex p_1 with $\varphi(p_1) = 0$ and ending at the vertex p_m with $\varphi(p_m) = 1$. Now construct the deformed product

$$Q \quad := \quad (P, \varphi) \bowtie (V, W),$$

where $V, W \subseteq \mathbb{R}^2$ are convex k-gons according to Lemma 4.6.

By Corollary 3.12 we get that Bland's rule, applied to Q with objective function $\widehat{\alpha}\binom{x}{u} = \alpha(u)$, first follows a P-path with l vertices from $v(1,1)$ to $v(m,1)$, then after one (V,W)-pivot it follows a P-path with l vertices from $v(m,2)$ to $v(1,2)$, then after one (V,W)-pivot it first follows a P-path with l vertices from $v(1,3)$ to $v(m,3)$, etc.. The complete path will thus visit kl vertices, arriving at $v(1,k)$ or at $v(m,k)$, depending on whether k is even or odd. ∎

Corollary 4.8 *For $n \geq 2d \geq 2$:*

$$H(d,n) \;\geq\; H_{\mathrm{Da}}(d,n) \;\geq\; H_{\mathrm{Bl}}(d,n) \;\geq\; \left\lfloor \frac{2n}{d} \right\rfloor^{d/2} \qquad \text{if } d \text{ is even,}$$

and

$$H(d,n) \;\geq\; H_{\mathrm{Da}}(d,n) \;\geq\; H_{\mathrm{Bl}}(d,n) \;\geq\; 2\left\lfloor \frac{2n-4}{d-1} \right\rfloor^{\lfloor d/2 \rfloor} \qquad \text{if } d \text{ is odd.}$$

Proof. Starting at $H_{\mathrm{Bl}}(0,0) = 1$ and $H_{\mathrm{Bl}}(1,2) = 2$, we get

$$H_{\mathrm{Bl}}(2m, km) \;\geq\; k^m \tag{10}$$

and

$$H_{\mathrm{Bl}}(2m{+}1, km{+}2) \;\geq\; 2\,k^m \tag{11}$$

for all $m \geq 0$, from (9) together with induction on m.

For even d, we use (10), and substitute $m := \frac{d}{2}$ and $k := \lfloor \frac{n}{m} \rfloor = \lfloor \frac{2n}{d} \rfloor$, where $n \geq 2d$ guarantees $k \geq 4$. For odd d, we use (11), and substitute $m := \frac{d-1}{2}$ and $k := \lfloor \frac{n-2}{m} \rfloor = \lfloor \frac{2n-4}{d-1} \rfloor$, where $n \geq 2d$ guarantees $k \geq 4$. ∎

Corollary 4.9 *For constant dimension d, the function $H(d,n)$ grows like a polynomial of degree $\lfloor \frac{d}{2} \rfloor$ in n. Furthermore, for even d*

$$\frac{1}{(d/2)!} \;=\; \lim_{n \longrightarrow \infty} \frac{M(d,n)}{n^{d/2}} \;\geq\; \liminf_{n \longrightarrow \infty} \frac{H(d,n)}{n^{d/2}} \;\geq\; \frac{1}{(d/2)^{d/2}},$$

and for odd d

$$\frac{2}{\lfloor d/2 \rfloor!} \;=\; \lim_{n \longrightarrow \infty} \frac{M(d,n)}{n^{\lfloor d/2 \rfloor}} \;\geq\; \liminf_{n \longrightarrow \infty} \frac{H(d,n)}{n^{\lfloor d/2 \rfloor}} \;\geq\; \frac{2}{\lfloor d/2 \rfloor^{\lfloor d/2 \rfloor}}.$$

Proof. While the upper bounds for $H(d,n)$ are from the Upper Bound Theorem 1.1, the lower bounds are easily derived from Corollary 4.8: for example, in the case where d is even we have $H(d,n) \geq (\frac{2n}{d} - 1)^{d/2}$, and hence

$$\frac{H(d,n)}{n^{(d-1)/2}} \;\geq\; \left(\frac{1}{d/2} - \frac{2}{n}\right)^{d/2},$$

which converges nicely for $n \longrightarrow \infty$ if d is fixed. ∎

(This amounts to a certain improvement in comparison with the analysis in Klee & Minty [27, p. 170].)

4.5 The Jeroslow Construction

Jeroslow [24] gave a construction of polytopes for which the greatest increase rule makes many pivots, which we can restate as deformed products. (Later Blair [6] proposed different programs for a diagonal case, which are also essentially deformed products.)

In the following, we keep close to Jeroslow's construction and notation. However, we complement his remark "The reader may wish to follow our construction with paper and pencil, since we shall refer to geometrical aspects of it." [24, p. 372] by a suitable figure.

Lemma 4.10 *Let $\alpha(x)$ denote the first coordinate for points $x \in \mathbb{R}^2$ in the plane.*

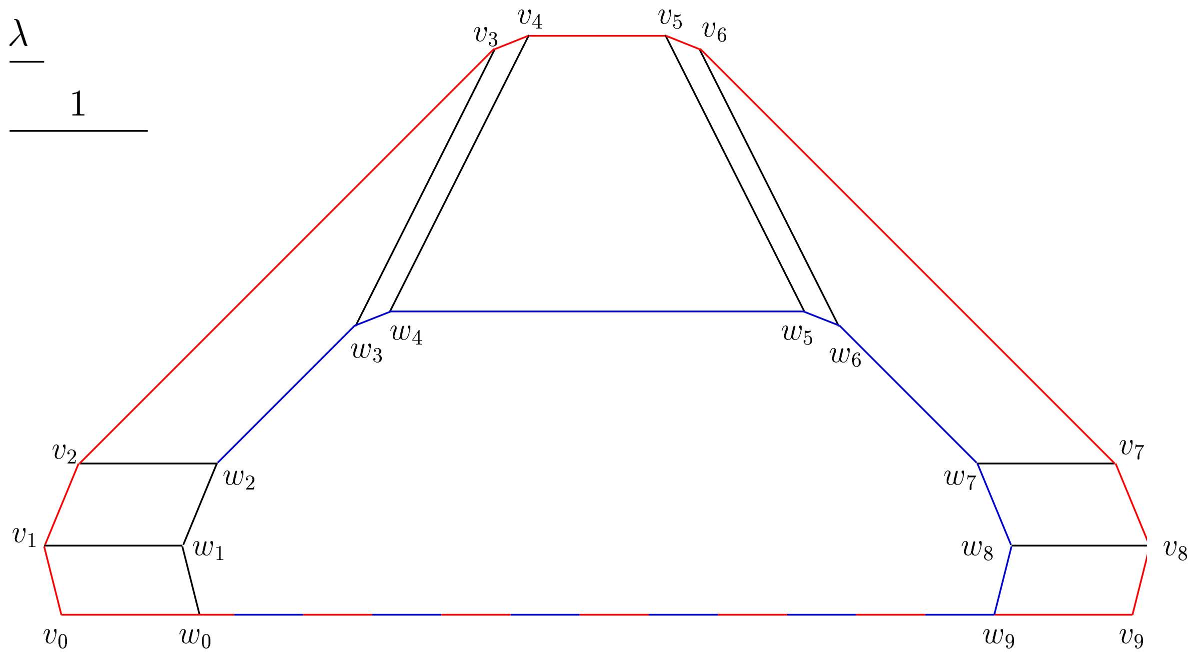
For each $k \geq 1$, and for each $\lambda > 0$, there exist normally equivalent $(4k{+}2)$-gons $V = \operatorname{conv}\{v_0, v_1, \ldots, v_{4k+1}\}$ and $W = \operatorname{conv}\{w_0, w_1, \ldots, w_{4k+1}\}$, such that, for $1 \leq l < 4k$,

$$\alpha(v_{l+1} - v_l) \;=\; \begin{cases} \lambda & \text{if } l \equiv 1 \bmod 4 \\ 3 & \text{if } l \equiv 2 \bmod 4 \\ \lambda & \text{if } l \equiv 3 \bmod 4 \\ 1 & \text{if } l \equiv 0 \bmod 4 \end{cases} \qquad \text{and} \qquad \alpha(w_{l+1} - w_l) \;=\; \begin{cases} \lambda & \text{if } l \equiv 1 \bmod 4 \\ 1 & \text{if } l \equiv 2 \bmod 4 \\ \lambda & \text{if } l \equiv 3 \bmod 4 \\ 3 & \text{if } l \equiv 0 \bmod 4 \end{cases}$$

and

$$\alpha(w_l - v_l) \;=\; \begin{cases} +1 & \text{if } l \equiv 1 \text{ or } 2 \bmod 4, \\ -1 & \text{if } l \equiv 3 \text{ or } 0 \bmod 4. \end{cases}$$

Furthermore, we require that $\alpha(v_1) < \alpha(v_0) < \alpha(v_2)$ and $\alpha(v_{4k-1}) < \alpha(v_{4k+1}) < \alpha(v_{4k})$.



Proof. If we place the vertex v_1 at the origin, then the polygons V and W are, in fact, determined by k and λ as soon as we prescribe the slopes m_i of the the edges $[v_l v_{l+1}]$ and $[w_l w_{l+1}]$ against the x-axis.

One possible choice is to take the slopes of a regular $8k$-gon, with $m_i = \cot(\pi/4k)$: this is what we (roughly) took for our figure, which represents the situation for $k = 2$ and $\lambda = 1/4$. (In the upper lefthand corner we represent the scale of the drawing.)

Alternatively one could for example take $m_l := 2k-l$: this yields "small" rational coordinates for the vertices, as is required in complexity arguments for the greatest increase pivot rule. □

With these polygons, a recursive deformed products construction can be used to establish the asymptotic lower bound on the length of greatest increase paths, as indicated in the next figure. To improve the bound, Jerowlow uses the fact that the resulting plytopes are "reversible":

Definition 4.11 (Reversible pairs, Jeroslow [24])
Let $P \subseteq \mathbb{R}^n$ be a polytope, and $\varphi\colon \mathbb{R}^d \longrightarrow \mathbb{R}$ a linear function with $\varphi(P) = [0,1]$ in general position on P. Let p_1 denote the unique minimal vertex of P with $\varphi(p_1) = 0$, and p_m the unique maximal vertex with $\varphi(p_m) = 1$.

Then (P, φ) is a *(reversible) greatest increase pair of length* l if the greatest increase path with respect to the objective function $c^t x := \varphi(x)$ (from p_1 to p_m) and the greatest increase path with respect to $c^t x := -\varphi(x)$ (from p_m to p_1) both have at least l vertices.

Thus $H_{\text{gi}}(d, n)$ is the greatest length of a reversible pair (P, φ) of type (d, n).

Theorem 4.12 *For all* $k \geq 1$,

$$H_{\text{gi}}(d{+}2, n{+}4k{+}2) \quad \geq 2k\, H_{\text{gi}}(d, n) \ + \ 2k.$$

Proof. For any reversible pair (P, φ) of length l of dimension d we can choose a number $\lambda > 0$ such that the progress along each edge in the two greatest increase paths for φ and for $-\varphi$ on P is larger than λ. (In particular, this condition implies that $\lambda < \frac{1}{l} \leq 1$.) Corresponding to this value of λ, construct the $(4k{+}2)$-gons $V, W \subseteq \mathbb{R}^2$ described in Lemma 4.10. With these one obtains a deformed product

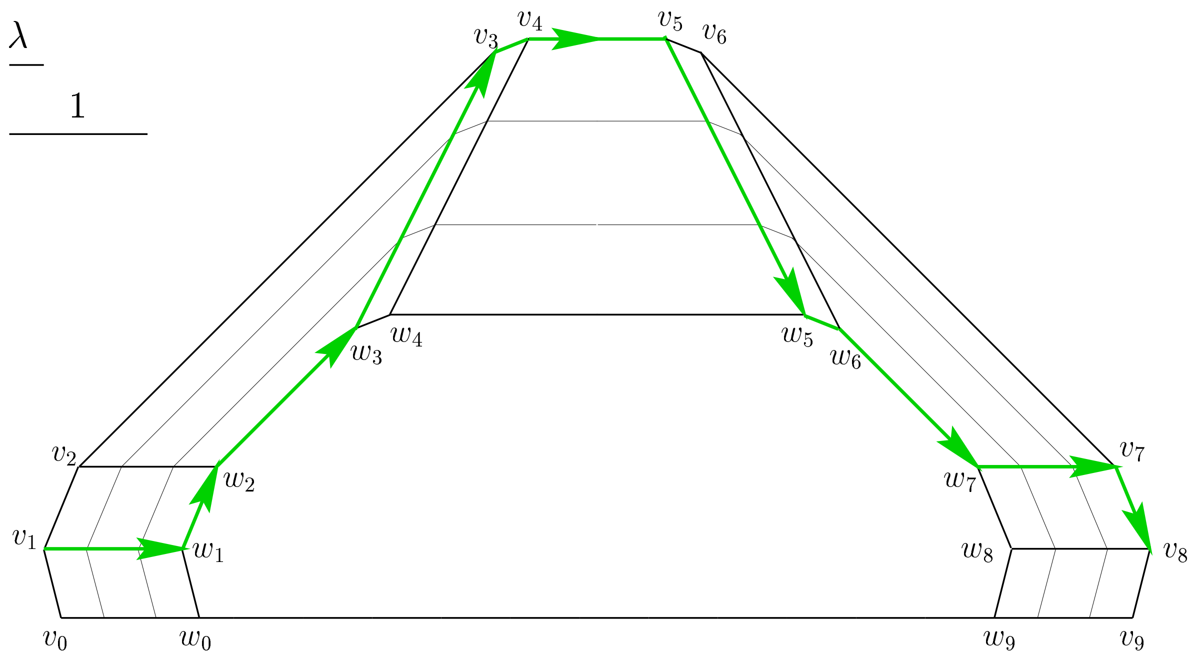
$$Q \quad := \quad (P, \varphi) \bowtie (V, W)$$

of dimension $d + 2$, with $n{+}4k{+}2$ facets. The linear function $\widehat{\alpha}\binom{x}{u} = \alpha(u)$ defines an objective function on Q that is in general position.

We have constructed V and W such that on every P-edge that the simplex algorithm may encounter the progress is more than λ, while the progress along a V-edge or W-edge of the deformed product is at most λ. Thus, once again, the simplex algorithm with the greatest increase rule prefers P-edges over (V, W)-edges. Thus the greatest increase rule applied to the objective function $\alpha \circ \pi_2$ on Q will trace a greatest increase path with l vertices from $v(1,1)$ to $v(m,1)$, then two (V, W)-steps from $v(m,1)$ via $v(m,2)$ to $v(m,3)$, then trace a greatest increase path of length l to $v(1,3)$, then do two steps from $v(1,3)$ via $v(1,4)$ to $v(1,5)$, etc. The vertices v_0, w_0, v_{4k+1} and w_{4k+1} are used to force the initial step of the path from v_1 resp. from v_{4k} to choose the P-edge. Our figure illustrates this for the case $l = 3$ and $\lambda = \frac{1}{4}$.

All in all, this determines a greatest increase path that has $2k$ sequences of P-edges that visit l vertices each, and touches $2k$ additional vertices, namely $w_2, v_4, w_2, \ldots, v_{4k}$. Thus the greatest increase path on (P, α) has $2kl + 2k$ vertices.

Analogously (or by symmetry), the same is true for the greatest increase path on $(Q, -\alpha)$. Thus (Q, α) is a reversible pair of length $2k(l{+}1)$. □



We use this lemma to derive both an exponential lower bound for $H_{\text{gi}}(d,n)$ in the diagonal case and to show that it grows like $n^{\lfloor d/2\rfloor}$ in the case of constant dimension.

Corollary 4.13 *The greatest increase rule traces, in the worst case, paths of exponential length:*

$$H_{\text{gi}}(d, 3d-1) \quad \geq \quad 3 \cdot 2^{\lfloor d/2\rfloor} - 2$$

holds for $d \geq 1$. *Furthermore,*

$$H_{\text{gi}}(d,n) \quad \geq \quad 2\left(2\left\lfloor\frac{n-d}{2d}\right\rfloor\right)^{d/2} - 2 \qquad\qquad \text{if } d > 0 \text{ is even,}$$

and

$$H_{\text{gi}}(d,n) \quad \geq \quad 3\left(2\left\lfloor\frac{n-d-1}{2(d-1)}\right\rfloor\right)^{\lfloor d/2\rfloor} - 2 \qquad \text{if } d \text{ is odd.}$$

Proof. The case $k = 1$ of Theorem 4.12, the initial conditions $H_{\text{gi}}(0,0) = 1$ and $H_{\text{gi}}(1,2) = 2$ and induction on m provide

$$H_{\text{gi}}(2m, 6m) \;\geq\; 3\cdot 2^m - 2 \qquad \text{and} \qquad H_{\text{gi}}(2m{+}1, 6m{+}2) \;\geq\; 4\cdot 2^m - 2$$

for all $m \geq 0$. Clearly $H_{\text{gi}}(d,n) \leq H_{\text{gi}}(d,n{+}1)$, from which we derive the first claim.

For the second claim, we start at $H_{\text{gi}}(0,0) = 1$ and $H_{\text{gi}}(1,2) = 2$, and get by induction that

$$H_{\text{gi}}(2m, (4k{+}2)m) \quad \geq \quad 2\,(2k)^m - 2$$

and

$$H_{\text{gi}}(2m{+}1, (4k{+}2)m{+}2) \quad \geq \quad 3\,(2k)^m - 2$$

for all $m \geq 0$, from which the result follows via suitable substitutions (as in the proof of Corollary 4.8). ∎

With the method of Corollary 4.9 one also gets a lower bound for $\liminf\limits_{n\longrightarrow\infty} H_{\text{gi}}(d,n)/n^{\lfloor d/2\rfloor}$ that improves somewhat on that by Jeroslow [24, p. 370].

5 Shadows

To the previous constructions that we have surveyed, we now add a new deformed product construction, which — via Lemma 2.9 — gives a lower bound for the number of simplex pivots required by the shadow vertex rule. We do this, of course, by constructing polytopes whose shadows have many vertices.

5.1 Shadows of d-Polytopes

Lemma 5.1 *Let $\alpha(x)$ denote a non-constant linear function on the plane ($x \in \mathbb{R}^2$). For each even $k \geq 4$, and for each $\lambda > 0$, there exist normally equivalent k-gons*

$$V = \mathrm{conv}\{v_1, \ldots, v_k\} \qquad \textit{and} \qquad W = \mathrm{conv}\{w_1, \ldots, w_k\}$$

(both labeled in clockwise order), such that

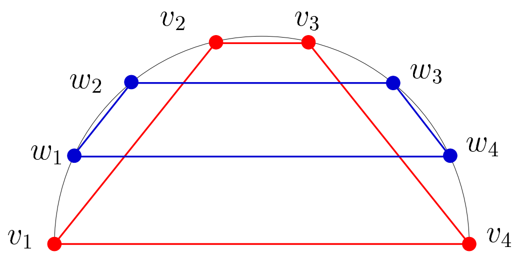
$$0 = \alpha(v_1) < \alpha(w_1) < \alpha(w_2) < \alpha(v_2) < \alpha(v_3) < \alpha(w_3) < \ldots < \alpha(w_k) < \alpha(v_k) = 1$$

and such that $\mathrm{conv}(V \cup W)$ is a convex $2k$-gon whose vertices appear on its boundary in the order

$$v_1, w_1, w_2, v_2, v_3, w_3, w_4, \ldots, w_k, v_k.$$

Proof. Take $2k$ points equally spaced on a semicircular arc of radius $\frac{1}{2}$ ("half a $2(2k-1)$-gon") and label them according to the order given by this lemma.

Our figure illustrates the case $k = 4$. □



Note: this lemma is false for all odd k.

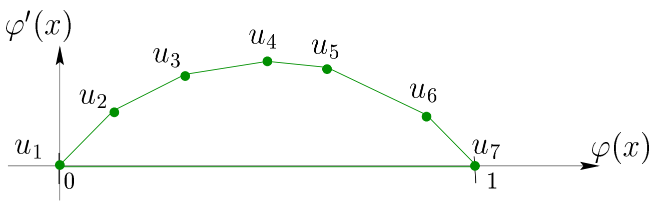
Theorem 5.2 *For all even $k \geq 4$,*

$$H_{2\text{-sh}}(d+2, n+k) \quad \geq \quad k\, H_{2\text{-sh}}(d, n).$$

Proof. Let P be a convex polytope with a 2-dimensional projection that has $m := H_{2\text{-sh}}(d, n)$ vertices. After a projective transformation, we may assume that the projection is given by

$$\pi\colon \mathbb{R}^d \longrightarrow \mathbb{R}^2, \qquad x \longmapsto (\varphi(x), \varphi'(x)),$$

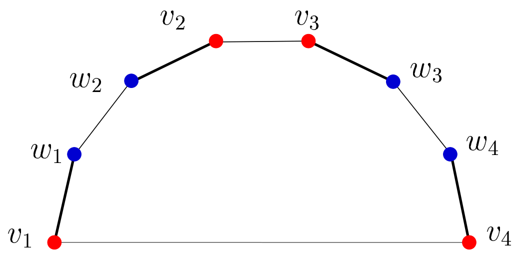
such that the image of this projection is a planar m-gon $\mathrm{conv}\{u_1, \ldots, u_m\}$ (labeled clockwise) with $u_1 = (0, 0)$, $u_m = (1, 0)$, and $0 = \varphi(u_1) < \varphi(u_2) < \ldots < \varphi(u_m) = 1$, while $\varphi'(i) \geq 0$ (with equality only for $i = 1$ and $i = m$), as in the figure:



With this preparation, we construct polygons $V, W \subseteq \mathbb{R}^2$ as in Lemma 5.1, and with these the deformed product

$$Q \quad := \quad (P, \varphi) \bowtie (V, W).$$

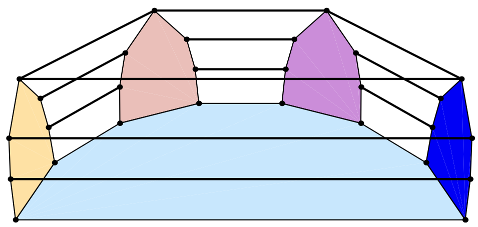
The projection of this polytope to the last two coordinates given by Corollary 3.6(ii) produces the following picture:



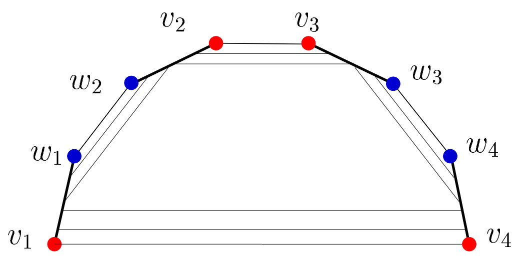
where the fat edges represent the images of the copies P_i of the polytope P. In fact, if instead we use the projection to $\mathbb{R}^3$ given by

$$\widehat{\pi}\colon \mathbb{R}^d \longrightarrow \mathbb{R}^3, \qquad x \longmapsto (\pi_2(x), \varphi'(x)),$$

then the image is a 3-dimensional polytope whose "base" is still given by this figure, while there is a little "arc" erected above each of the fat edges. The graph of the resulting polytope,



in the top view (for $m = 6$), may look somewhat like the following:



This top view corresponds to deleting the last coordinate from $\mathbb{R}^3$, and thus to taking the projection π_2 applied to Q. Now we take a slightly shifted projection, namely the projection

$$\begin{array}{rcccccc} \widehat{\pi}^\varepsilon\colon & \mathbb{R}^d & \xrightarrow{\widehat{\pi}} & \mathbb{R}^3 & \longrightarrow & \mathbb{R}^2 \\ & x & \longmapsto & (\pi_2(x), \varphi'(x)) & \longmapsto & \pi_2(x) + \varepsilon\varphi'(x)\, e_2 \end{array}$$

If $\varepsilon > 0$ is chosen small enough, then all the vertices of the "arcs above the fat edges" will appear in the shadow boundary; thus instead of the $2k$ vertices of the image of π_2 (two for each of the fat edges $[v_i, w_i]$), the image of the shifted projection will have mk vertices (very flat arcs with m vertices each for each of the k fat edges).

Hence the projection $\widehat{\pi}^\varepsilon$ of Q has $km = k\, H_{2\text{-sh}}(d, m)$ vertices. $\square$

Corollary 5.3 *For $n \geq 2d \geq 2$,*

$$H_{2\text{-sh}}(d,n) \quad \geq \quad \left(2\left\lfloor\frac{n}{d}\right\rfloor\right)^{d/2} \qquad \text{if } d \text{ is even,}$$

and

$$H_{2\text{-sh}}(d,n) \quad \geq \quad 2\left(2\left\lfloor\frac{n-2}{d-1}\right\rfloor\right)^{\lfloor d/2\rfloor} \qquad \text{if } d \text{ is odd.}$$

Proof. This is (nearly) the same analysis as for Corollary 4.8. ∎

For $\liminf_{n\longrightarrow\infty} H_{2\text{-sh}}(d,n)/n^{\lfloor d/2\rfloor}$ we also get the *same* bounds that were derived in Corollary 4.9 for $\liminf_{n\longrightarrow\infty} H(d,n)/n^{\lfloor d/2\rfloor}$, with the same proof.

5.2 *k*-Shadows and Applications

The deformed product of polytopes with "large shadows" described in the last section has a number of applications outside of linear programming. The lower bound applies to more general geometric constructions, thanks to the following observations.

Observation 5.4 The complexity of the shadow of a polytope on the plane (the 2-shadow) is a lower bound on the complexity of a k-shadow (the projection of a d-dimensional polytope to a k-dimensional subspace), since projections do not increase the number of vertices of a polytope.

A *k-section* of a polytope P in $\mathbb{R}^d$ is the intersection of P with a k-dimensional hyperplane.

Observation 5.5 The polar of a k-shadow of P is a k-section of the polar of P. (For instance the polar of a projection of P from a point x is the intersection of the polar hyperplane of x with the polar of P.)

Analogous to $H_{k\text{-sh}}(d,n)$, which denotes the maximal number of vertices of any k-shadow of a (d,n)-polytope, let $H_{k\text{-sec}}(d,n)$ be the maximum, over all d polytopes with n vertices, of the number of facets in a k-section.

Corollary 5.6 $H_{k\text{-sec}}(d,n) = H_{k\text{-sh}}(d,n) \ \in \ O(n^{\lfloor d/2\rfloor})$.

The computation of shadows and sections are natural problems in Computational Geometry; we can think of the computation of k-shadows as an intermediate problem between Linear Programming (the 1-shadow) and Halfspace Intersection (the d-shadow). One natural application is the maximization of a convex function of k variables over a polytope with n facets; the maximum is achieved at a vertex of the k-shadow.

Several researchers have studied the construction of k-shadows with respect to particular geometric search and optimization problems. Ponce & Faverjon citePoFa compute the set of stable three-finger grasps, with friction, of a polygon in the plane, as the 3-shadow of a 5-dimensional polytope. This result is extended in [35], where a subset of the stable four-finger grasps, with friction, of a polyhedron in $\mathbb{R}^3$ is computed the as 8-shadow of an 11-dimensional polytope. Agarwal, Amenta & Sharir [1] reduce the problem of finding the largest similar copy of one convex polygon contained in another to the maximization of a convex function in two dimensions over a polytope in $\mathbb{R}^4$; they use the particular structure of this polytope to get an efficient algorithm for the computation of the shadow. Algorithms for the computation of general shadows were also developed in the context of Logic Programming by Huynh, Lassez & Lassez [23].

5.3 k-shadow Algorithms

In this section we consider some old and new algorithms for the problem:

Problem 5.7 (k-Shadow)
Input: A set $\mathcal{H}$ of n linear halfspaces in $\mathbb{R}^d$
Output: The projection of the polytope $P = \bigcap \mathcal{H}$ to the subspace spanned by $e^1, \ldots, e^k$.

Corollary 5.6 settles the question of the worst-case complexity of the problem when P is a simple polytope. But the problem remains interesting because one expects that, in most situations, the k-shadow, $k < d$, will be significantly less complex than P itself. (For $k = 2$ this intuition is proved, in a probabilistic model, by Borgwardt's analysis of the Shadow Vertex Algorithm [8].) The goal therefore is an optimal output-sensitive algorithm. An algorithm is *output-sensitive* when the running time depends on both the size of the input and the size of the output.

A hyperplane of the form $\{x \in \mathbb{R}^d : a_1x_1 + a_2x_2 + \ldots + a_kx_k + b = 0\}$ in $\mathbb{R}^d$ is called *k-vertical* (a k-vertical hyperplane is parallel to the $x_{k+1}, \ldots, x_d$ axes). We say that a face f of P is *supported by* a k-vertical hyperplane H if f is contained in H and P is entirely contained in one of the closed halfspaces bounded by H. The faces of P supported by k-vertical hyperplanes are the faces of the k-shadow.

We begin with the obvious worst-case optimal algorithm for computing the k-shadow. This algorithm returns a list F of all faces of the k-shadow.

Algorithm 5.8 (Worst-case Algorithm)

1 Compute P
2 For every face f of P
3   if f is supported by a k-vertical hyperplane
4   add f to F

For Step **1**, there is an asymptotically worst-case optimal algorithm, due to Chazelle [10], for computing a simple polytope as an intersection of halfspaces, which runs in time $O(n^{\lfloor d/2 \rfloor})$. The test in Step **3** can be performed in constant time (when d is taken to be constant). Together with Corollary 5.6 we get the following.

Theorem 5.9 *The k-shadow of a simple polytope P can be computed in $O(n^{\lfloor d/2 \rfloor})$ time, and this is optimal in the worst case.*

We now turn to algorithms that achieve some degree of output-sensitivity by computing the k-shadow of P without computing P itself. We compare the running times in terms of both the number s of shadow facets and the number v of shadow vertices. We shall consider only the case which P is simple. Since any vertex of a simple polytope is adjacent to exactly $\binom{d}{k-1}$ faces of dimension $k-1$, taking d and k to be constants implies that, in the k-shadow, $s = O(v)$. But the only bound we have on v with respect to s is that $v = O(s^{\lfloor k/2 \rfloor})$, from the Upper Bound Theorem. So it is possible, for large k, that $v \gg s$.

Ponce et al. [35] described and tested the following pivoting algorithm, analogous, for instance, to the convex hull algorithm of Avis & Fukuda [5]. It uses two data structures, a stack S and a dictionary D, and returns a list V of the vertices of the k-shadow.

Algorithm 5.10 (Shadow Vertex Pivoting Algorithm)

1 Find the minimum vertex p_0 of P w.r.t. the x_1 coordinate
2 Insert p_0 in S
3 Insert p_0 in D
4 Add p_0 to V
5 While $S \neq \emptyset$
6 remove a vertex p from S
7 for each edge e adjacent to p
8 if e is supported by a k-vertical hyperplane
9 find the other endpoint p' of e
10 if $p' \notin D$
11 insert p' in D
12 insert p' in S
13 add p' to V
14 Return V

Step **1** can be done by linear programming, which, when d is taken to be a constant, requires time $O(n)$ (although exponential in d). Step **9** is done by intersecting the ray supporting e anchored at p with each of the input halfspaces in turn. The intersection nearest p is p'. Since the degree of each vertex in the k-shadow is constant, the entire face structure can be derived from the list of vertices in time $O(v)$.

Theorem 5.11 (Ponce et. al., [35])
The k-shadow of a simple polytope given as an intersection of n halfspaces in fixed dimension can be computed in time $O(vn)$.

We give a different algorithm for computing the k-shadow, which is more efficient when $v \gg s$. The algorithm operates in two phases. In the first phase, we find all the shadow facets, rather than the shadow vertices, by pivoting. In the second phase we construct the k-shadow as the intersection of the shadow facets using a halfspace intersection algorithm.

In a pivoting step in the first phase, we find all shadow facets adjacent to the current one. Let f be the current shadow facet, a $(k-1)$-face of P that projects to a facet of the shadow. The face f lies in the intersection G of some set $\mathcal{G}$ of $d-k+1$ input hyperplanes. For each input hyperplane F not in $\mathcal{G}$, we replace each of the $d-k+1$ hyperplanes of $\mathcal{G}$ in turn with F, to get a set $\mathcal{G}'$ which intersects in a different $(k-1)$-flat G'. G' contains a shadow facet if, first, G' contains a $(k-1)$-face of P, and second, if that $(k-1)$-face supported by a vertical hyperplane.

The input halfspaces corresponding to $\mathcal{G}'$ define an unbounded polytope P' which contains P. If G', as a face of P', is not supported by a k-vertical hyperplane, then any face of P contained in G' will not be supported by a k-vertical hyperplane either. In that case we know that G' cannot contain an adjacent shadow facet.

Otherwise we run a $(k-1)$-dimensional linear program in the affine subspace G', minimizing x_1, to find out if $G' \bigcap P = \emptyset$. If the linear program is infeasible, G' fails to intersect P; otherwise it returns a point p which minimizes x_1 on an adjacent shadow facet f' induced by G'.

For each f, we run at most $(d-k)n$ such linear programs, each requiring $O(n)$ time. So finding all shadow facets adjacent to f requires $O(n^2)$ time.

The algorithm for the pivoting phase again requires a stack S and a dictionary D, and returns a list F of shadow facets.

Algorithm 5.12 (Shadow Facet Pivoting Algorithm)

1 Find the minimum vertex p_0 of P w.r.t. the x_1 coordinate

```
2     For each (k−1)-face f adjacent to p_0
3         if f is supported by a k-vertical hyperplane
4             insert f in S
5             insert f in D
6             add f to F
7     While S ≠ ∅
8             remove a shadow facet f from S
9             find all shadow facets adjacent to f
10            for every shadow facet f' adjacent to f
11                if f' ∉ D
12                    insert f' in D
13                    insert f' in S
14                    add f' to F
15    Return F
```

Step **9** is executed once for every shadow facet and requires (by the previous discussion) $O(n^2)$ time, so the running time of the pivoting phase is $O(sn^2)$.

From the list F of shadow facets, the entire k-shadow can be computed by a convex hull algorithm in dimension k. Using Chazelle's algorithm this takes time $O(s^{\lfloor k/2 \rfloor})$, for an overall running time of $O(sn^2 + s^{\lfloor k/2 \rfloor})$. When $s^{\lfloor k/2 \rfloor} \gg v \gg s$, we can do better by using Seidel's output-sensitive convex hull algorithm, which takes time $O(s^2 + v \log s)$. The s^2 term in the running time for Seidel's algorithm comes from a first phase which finds the minimal vertex, with respect to x_1, in each facet of the final halfspace intersection. Since the Shadow Facet Pivoting Algorithm produces these minimal vertices as a by-product of finding the shadow facets, we can skip the first phase, for an overall running time of $O(sn^2 + v \log s)$.

Theorem 5.13 *A k-shadow of a polytope P, given as the intersection of an input set $\mathcal{H}$ of halfspaces, can be computed in time $O(sn^2 + \min(s^{\lfloor k/2 \rfloor}, v \log s))$, where $n = |\mathcal{H}|$, s is the number shadow facets, and v is the number shadow vertices.*

Both pivoting algorithms run a lot of linear programs using almost identical sets of constraints. The running times can therefore be improved slightly using the ray-shooting data structure of Matoušek & Schwarzkopf [28, 29] (a similar observation is made in [35], and by Chan [9]).

5.4 Shadows of Cyclic Polytopes

The polar of a cyclic polytope has the maximum number of faces of all dimensions among polytopes with n facets. We show, however, that in even dimension the polars of cyclic polytopes do not maximize the complexity of the shadow among all polytopes with n facets. This is equivalent to the conclusion that for large n a monotone path through all vertices cannot be found by the "shadow vertex algorithm," for any realization of $C_d(n)^{\Delta}$. Equivalently, a Bruggesser-Mani shelling of $C_d(n)$ [40, Sect. 8.2] cannot in general be generated by a 2-dimensional section of $C_d(n)$ that would cut all the facets. In fact, we show that in even dimensions the shadow of a polar of a cyclic polytope must have asymptotically fewer vertices than the polytope itself.

Let us review some properties of cyclic polytopes; for more details, see [40, Example 0.6]. A curve C in $\mathbb{R}^d$ is *of order d* if every hyperplane intersects C in at most d points. The convex hull of any set of $n > d$ points that lie on a curve of order d is a cyclic polytope $C_d(n)$. Sturmfels [38] has shown that if the dimension d is even, then any polytope P that is combinatorially isomorphic to a polytope $C_d(n)$ has its vertices on some curve of order d.

A facet of P is supported by a $(d-1)$-plane H which contains d vertices. Let H^+ denote the positive halfspace of H, which is determined by $P \subseteq H^+$. Index the vertices $v_1, \ldots, v_n$ along a suitable directed d-order curve C. If C leaves H^+ at a vertex v_i, then it must reenter H^+ at v_{i+1}, since otherwise v_{i+1} would be outside H^+ and hence outside P. Thus the vertex set of any facet of P is a union of adjacent pairs v_i, v_{i+1} ("Gale's evenness criterion" [40, Thm. 0.7]).

Every face of smaller dimension is determined by a subset of the vertex set of a facet. This means that the $(d-2)$-faces of P are of the form

$$T(i_1, \ldots, i_{d/2}) := \mathrm{conv}\{v_{i_1}, v_{i_1+1}, v_{i_2}, v_{i_2+1}, \ldots, v_{i_{d/2}}\},$$

with $i_k \notin \{i_l, i_l+1\}$ (and indices taken mod n). That is, there are $d/2-1$ adjacent pairs of vertices, and then one vertex, $v_{i_{d/2}}$, which needs not be adjacent to any other. We choose our numbering in such a way that $v_{i_{d/2}+1}$ is not a vertex of $T(i_1, \ldots, i_{d/2})$.

Theorem 5.14 *Let $P \subseteq \mathbb{R}^d$ be a cyclic d-polytope with n vertices, where $d \geq 4$ is even, and let P^Δ be a polar polytope of P. Then any 2-dimensional shadow of P^Δ has at most*

$$\frac{6n}{d-2}\binom{n-d/2}{d/2-2} \quad = \quad O(n^{d/2-1})$$

vertices.

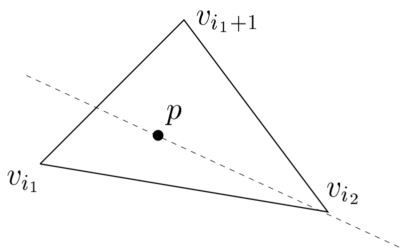
Proof. Using the duality between projections and sections, we consider a 2-dimensional section $F \cap P$ of the cyclic polytope P, and verify that it cannot have more than $\frac{6n}{d-2}\binom{n-d/2}{d/2-2}$ vertices.

The maximum is clearly achieved for a 2-plane that meets the interior of P, and that is in general position with respect to P. In fact, given any 2-plane F that meets the interior of P, the number of vertices of $F \cap P$ cannot decrease by a sufficiently small perturbation of F.

Under these conditions, every vertex p of $F \cap P$ is determined by a unique $(d-2)$-dimensional face $T(i_1, \ldots, i_{d/2})$. Consider the $(d-1)$-dimensional hyperplane

$$H(i_2, \ldots, i_{d/2}) \quad := \quad \mathrm{aff}(F \cup \{v_{i_2}, \ldots, v_{i_{d/2}}\}).$$

This hyperplane intersects the $(d-2)$-flat supporting $T(i_1, \ldots, i_{d/2})$ in a $(d-3)$-flat which contains the vertices $v_{i_2}, \ldots, v_{i_{d/2}}$. There is a one-dimensional family of such $(d-3)$-flats, one of which supports the $(d-3)$-face $\mathrm{conv}\{v_{i_1}, v_{i_2}, \ldots, v_{i_{d/2}}\}$ and another supporting the $(d-3)$-face $\mathrm{conv}\{v_{i_1+1}, \ldots, v_{i_{d/2}}\}$. The $(d-3)$-flats that intersect the interior of $T(i_1, \ldots, i_{d/2})$ separate the points v_{i_1} and v_{i_1+1}. If F hits $T(i_1, \ldots, i_{d/2})$ in the interior point p, then $H(i_2, \ldots, i_{d/2})$ separates the points v_{i_1} and v_{i_1+1} on the moment curve. The figure represents the four-dimensional case.



But every hyperplane intersects C at most d times; and for $H(i_2, \ldots, i_{d/2})$, $d-3$ of those intersections are the vertices $i_2, \ldots, i_{d/2}$. Hence there are at most three possible choices for the pair v_{i_1}, v_{i_1+1} separated by $H(i_2, \ldots, i_{d/2})$.

Thus we can bound the number of vertices of $F \cap P$. There are n choices for the index $i_{d/2}$, after that there are $\binom{n-2-(d/2-2)}{d/2-2}$ choices for the set set of indices $\{i_2, \ldots, i_{d/2-1}\}$, and after that there are at most three choices for i_1. In this count, each face appears $(d/2-1)$-times, since the choice of i_1 among $\{i_1, \ldots, i_{d/2-1}\}$ was arbitrary. Thus there are at most

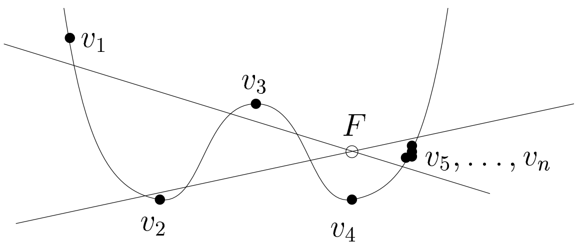
$$\frac{1}{d/2-1} 3n \binom{n-d/2}{d/2-2} \quad = \quad \frac{6n}{d-2}\binom{n-d/2}{d/2-2} \quad = \quad O(n^{d/2-1})$$

vertices p of $F \cap P$. ☐

In dimension $d = 4$ this means that the 2-shadow cannot have more than $3n$ vertices. For this we have an (almost) matching lower bound.

Theorem 5.15 *Let P be a cyclic polytope in $\mathbb{R}^4$ with n vertices, with 0 in its interior, and let P^Δ be the polar polytope of P. The shadow of P^Δ in $\mathbb{R}^2$ can have at many as $3n-10$ vertices.*

Proof. For the lower bound, construct a curve of degree 4, a flat F and points $v_1, \ldots, v_n$ on the curve so that if you project the curve to an orthogonal complement to F, which maps F itself to a point, you obtain the situation in our figure.



This can, for example, be achieved very explicitly by taking the curve C given by $\gamma(t) := (t, \binom{t}{2}, \binom{t}{3}, \binom{t}{4})$, taking π_{14} to be the projection to the first and fourth coordinate, and taking $F := \{(\frac{5}{2}, x, y, 0)\}$, and choosing the points on C as $v_1 := \gamma(-1)$, $v_2 := \gamma(\frac{1}{2})$, $v_3 := \gamma(\frac{3}{2})$, $v_4 := \gamma(\frac{5}{2})$, and $v_i := \gamma(t_i)$ with t_i very close to 4, for all i $(5 \le i \le n)$.

$F \cap T(i,j) \neq \emptyset$ holds if and only if in the 2-dimensional projection the point $\pi_{14}(F)$ is in the interior of the triangle $\pi_{14}(T(i,j))$, which is spanned by $\pi_{14}(v_i)$, $\pi_{14}(v_j)$ and $\pi_{14}(v_{j+1})$.

Every 2-face of P formed by $\{v_1, v_2\}$, $\{v_2, v_3\}$ or $\{v_3, v_4\}$, together with any one of the vertices $v_5, \ldots, v_n$ (that is, the triangles $T(i,1)$, $T(i,2)$ and $T(i,3)$ for $5 \le i \le n$) intersect F. This gives $3(n-4)$ vertices in $P \cap F$. In addition, the 2-faces $\mathrm{conv}\{v_1, v_4, v_5\}$ and $\mathrm{conv}\{v_1, v_4, v_n\}$ (i.e., the triangles $T(1,4)$ and $T(4,n)$) also intersect F, for a total of $3n-10$. ☐

Acknowledgments

We are grateful to Bernd Gärtner and to Raimund Seidel for helpful discussions.

References

[1] Pankaj Agarwal, Nina Amenta, & Micha Sharir: *Largest placement of one convex polygon inside another, Discrete Comput. Geometry* **19** (1998), 95-104.

[2] Nina Amenta & Günter M. Ziegler: *Shadows and slices of polytopes,* in: Proceedings of the *12th Annual ACM Symposium on Computational Geometry*, May 1996, pp. 10-19.

[3] DAVID AVIS, DAVID BREMNER & RAIMUND SEIDEL: *How good are convex hull algorithms? Computational Geometry: Theory and Applications* **7** (1997), 265-301.

[4] DAVID AVIS & VAŠEK CHVÁTAL: *Notes on Bland's pivoting rule,* in: "Polyhedral Combinatorics" (M.L. Balinski, A.J. Hoffmann, eds.), *Math. Programming Study* **8** (1978), 24-34.

[5] DAVID AVIS & KOMEI FUKUDA: *A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra, Discrete Comput. Geometry* **8** (1992) 295-314.

[6] CHARLES E. BLAIR: *Some linear programs requiring many pivots,* Working Paper No. 867, College of Commerce and Business Administration, University of Illinois, May 1982.

[7] ROBERT G. BLAND: *New finite pivoting rules for the simplex method, Math. Operations Research* **2** (1977), 103-107.

[8] KARL HEINZ BORGWARDT: *The Simplex Method. A Probabilistic Analysis,* Algorithms and Combinatorics **1**, Springer-Verlag, Heidelberg 1987.

[9] TIMOTHY CHAN: *Output-sensitive results on convex hulls, extreme points and related problems,* in: Proc. "11th Annual Symposium on Computational Geometry" 1995, 10-19.

[10] BERNARD CHAZELLE: *An optimal convex hull algorithm for point sets in any fixed dimension Discrete & Computational Geometry* **9** (1993), 145-158.

[11] BERNARD CHAZELLE, HERBERT EDELSBRUNNER & LEONIDAS J. GUIBAS: *The complexity of cutting complexes, Discrete Comput. Geom.* **4** (1989), 139-182.

[12] VAŠEK CHVÁTAL: *Linear Programming,* W. H. Freeman, New York 1983.

[13] JENS CLAUSEN: *Worst-case behaviour of simplex-algorithms: From theory to numerical examples,* Report 82/1, DIKU – Institute of Datology, University of Copenhagen, October 1981, revised February 1985, 35 pages.

[14] JENS CLAUSEN: *A new family of exponential LP problems, European J. Operational Research* **32** (1987), 130-139.

[15] YAHYA FATHI & CRAIG TOVEY: *Affirmative action algorithms, Math. Programming* **34** (1986), 292-301.

[16] SAUL I. GASS AND THOMAS SAATY: *The computational algorithm for the parametric objective function, Naval Research Logistics Quarterly* **2** (1955), 39-45.

[17] BERND GÄRTNER & GÜNTER M. ZIEGLER: *Randomized simplex algorithms on Klee-Minty cubes,* in: Proc. 35th Annual "Symposium on Foundations of Computer Science" (FOCS), November 20-22, 1994, Santa Fe NM, IEEE Computer Society Press, Los Alamitos CA, 1994, pp. 502-510.

[18] DONALD GOLDFARB: *Worst case complexity of the shadow vertex simplex algorithm,* Preprint, Columbia University 1983, 11 pages.

[19] DONALD GOLDFARB: *On the complexity of the simplex algorithm,* in: *Advances in optimization and numerical analysis*, Proc. 6th Workshop on Optimization and Numerical Analysis, Oaxaca, Mexico, January 1992; Kluwer, Dordrecht 1994, 25-38.

[20] DONALD GOLDFARB & WILLIAM T. SIT: *Worst case behaviour of the steepest edge simplex method, Discrete Applied Math.* **1** (1979), 277-285.

[21] PETER GRITZMANN & VICTOR KLEE: *Mathematical programming and convex geometry,* in: Handbook of Convex Geometry (P. Gruber, J. Wills, eds.), Vol. B, North Holland, Amsterdam 1993, pp. 627-674.

[22] Martin Grötschel, László Lovász & Alexander Schrijver: *Geometric Algorithms and Combinatorial Optimization,* Algorithms and Combinatorics **2**, Springer-Verlag, Berlin Heidelberg 1988; second edition 1994.

[23] Tien Huynh, Catherine Lassez and Jean-Louis Lassez: *Practical issues on the projection of polyhedral sets,* Annals of Mathematics and Artificial Intelligence **6** (1992), 295-316.

[24] Robert G. Jeroslow: *The simplex algorithm with the pivot rule of maximizing improvement criterion,* Discrete Math. **4** (1973), 367-377.

[25] Gil Kalai: *A subexponential randomized simplex algorithm,* in: "Proc. 24th ACM Symposium on the Theory of Computing (STOC)", ACM Press 1992, pp. 475-482.

[26] Victor Klee: *A class of linear programs requiring a large number of iterations,* Numerical Math. **7** (1965), 313-321.

[27] Victor Klee & George J. Minty: *How good is the simplex algorithm?,* in: "Inequalities III," (O. Shisha, ed.), Academic Press, New York, (1972), 159-175.

[28] Jiří Matoušek: *Linear optimization queries,* Journal of Algorithms **14** (1993), 432-448.

[29] Jiří Matoušek & Otfried Schwarzkopf: *Linear optimization queries,* in: "Proc. 8th Annual Symposium on Computational Geometry" (Berlin 1992), ACM Press 1992, 16-25.

[30] Jiří Matoušek, Micha Sharir & Emo Welzl: *A subexponential bound for linear programming,* in: "Proc. 8th Annual ACM Symp. Computational Geometry" (Berlin 1992), ACM Press 1992, 1-8.

[31] Peter McMullen: *The maximum numbers of faces of a convex polytope,* Mathematika **17** (1970), 179–184.

[32] Katta G. Murty: *Computational complexity of parametric linear programming,* Math. Programming **19** (1980), 213-219.

[33] Manfred Padberg: *Linear Programming and Extensions,* Algorithms and Combinatorics **12**, Springer-Verlag, Heidelberg 1995.

[34] Jean Ponce & Bernard Faverjon: *On computing three-finger force-closure grasps of polygonal objects,* IEEE Transactions on Robotics and Automation **11** (1995), 868-881.

[35] Jean Ponce, Steve Sullivan, Attawith Sudsang, Jean-Daniel Boissonnat & Jean-Pierre Merlet: *On computing four-finger equilibrium and force-closure grasps of polyhedral objects,* International Journal of Robotics Research **16** (1996), 11-35.

[36] Alexander Schrijver: *Theory of Linear and Integer Programming,* Wiley-Interscience Series in Discrete Mathematics and Optimization, John Wiley & Sons, Chichester New York 1986.

[37] Raimund Seidel: *Constructing higher dimensional convex hulls at logarithmic cost per face,* Proc. 18th Annual Symposium on the Theory of Computing, (1986), 404-413.

[38] Bernd Sturmfels: *Cyclic polytopes and d-order curves,* Geometriae Dedicata **24** (1987), 103-107.

[39] Norman Zadeh: *What is the worst case behavior of the simplex algorithm?,* Technical Report No. 27, Dept. Operations Research, Stanford 1980, 26 pages.

[40] Günter M. Ziegler: *Lectures on Polytopes,* Graduate Texts in Mathematics **152**, Springer-Verlag, New York 1995; Revised edition 1998.
(*Updates, corrections, and more:* `http://www.math.tu-berlin.de/~ziegler`)