Randomized Simplex Algorithms on Klee-Minty Cubes[‡]

Bernd Gärtner[*] Martin Henk[**] Günter M. Ziegler[**]

Abstract

We investigate the behavior of randomized simplex algorithms on special linear programs. For this, we use combinatorial models for the Klee-Minty cubes [22] and similar linear programs with exponential decreasing paths.

The analysis of two most natural randomized pivot rules on the Klee-Minty cubes leads to (nearly) quadratic lower bounds for the complexity of linear programming with random pivots. Thus we disprove two bounds (for the expected running time of the RANDOM-EDGE simplex algorithm on Klee-Minty cubes) conjectured in the literature.

At the same time, we establish quadratic upper bounds for the expected length of a path for a simplex algorithm with random pivots on the classes of linear programs under investigation. In contrast to this, we find that the average length of an increasing path in a Klee-Minty cube is exponential when all paths are taken with equal probability.

1 Introduction

Linear programming is the problem of minimizing a linear objective function over a polyhedron $P \subseteq \mathbb{R}^n$ given by a system of m linear inequalities.

Without loss of generality [30] we may assume that the problem is primally and dually nondegenerate, that the feasible region is full-dimensional and bounded, and that the objective function is given by the last coordinate. In other words, we consider the problem of finding the "lowest vertex" (minimizing x_n) of a simple n-dimensional polytope $P \subseteq \mathbb{R}^n$ with at most m facets, where the last coordinate x_n is not constant on any edge, and thus the lowest vertex is unique.

In this setting, the (geometric interpretation of the) simplex algorithm proceeds from some starting vertex of P along edges in such a way that the objective function decreases, until the unique lowest vertex of P is found. The (theoretical and practical) efficiency of the simplex algorithm [31] depends on a suitable choice of decreasing edges that "quickly leads to the lowest vertex." Connected to this are two major problems of linear programming: the diameter problem "Is there a short path to the lowest vertex?", and the

[‡]A shorter "extended abstract" version of this paper has appeared in [11].

[*]Supported by the ESPRIT Basic Research Action No. 7141 of the EU (project ALCOM II).

[**]Supported by a grant in the "Gerhard-Hess-Program" of the German Science Foundation (DFG).

algorithm problem "Is there an algorithm which quickly finds a (short) path to the lowest vertex?"

The **diameter problem** is closely related to the "Hirsch conjecture" (from 1957) and its variants [8] [20] [34]. Currently there is no counterexample to the "Strong monotone Hirsch conjecture" [34] that there always has to be a decreasing path, from the vertex which maximizes x_n to the lowest vertex, of length at most $m-n$. On the other hand, the best arguments known for upper bounds establish paths whose length is roughly bounded by $m^{\log_2 2n}$ [17], see also [34].

The **algorithm problem** includes the quest for a strongly polynomial algorithm for linear programming. Klee & Minty [22] showed in 1972 that linear programs with exponentially long decreasing paths exist, and that Dantzig's "largest coefficient" pivot rule [8] can be tricked into selecting such a path. Using variations of the Klee-Minty constructions, it has been shown that the simplex algorithm may take an exponential number of steps for virtually every deterministic pivot rule [20] [1]. (A notable exception is Zadeh's rule [33] [20], locally minimizing revisits, for which Zadeh's $1,000.– prize [20, p. 730] has not been collected, yet.)

No such evidence exists for some natural *randomized* pivot rules, among them the following three rules.

RANDOM-EDGE: At any nonoptimal vertex x of P, follow one of the decreasing edges leaving x with equal probability.

RANDOM-FACET: If x admits only one decreasing edge, then take it. Otherwise restrict the program to a randomly chosen facet containing x. This yields a linear program of smaller dimension in which x is nonoptimal, and which can be solved by recursive call to RANDOM-FACET. Then repeat with the vertex obtained from the recursive call.

RANDOM-SHADOW: Start at the unique vertex $y \in P$ which maximizes x_n. Choose a random unit vector $\mathbf{c}$ orthogonal to $\mathbf{e}_n$. Now take the path from y to the lowest vertex given by $\{x \in P : \mathbf{c}x \leq \mathbf{c}z \text{ for all } z \in P \text{ with } z_n = x_n\}$.

RANDOM-FACET is a randomized version, due to Kalai [16], of Bland's PROCEDURE A [3], which assumes that the facets are numbered, and always restricts to the facet with the smallest index. Interestingly enough, on an n-dimensional linear program with $m = n+k$ inequalities, the maximum expected number of steps performed by RANDOM-FACET is bounded by $O(ne^{O(\sqrt{k \log n})})$, which leads to a remarkable subexponential bound if k is small, see Kalai [16]. (Matoušek, Sharir & Welzl [25] prove a good bound if k is large, in a very similar dual setting [13].)

The RANDOM-SHADOW rule is a randomized version of Borgwardt's SHADOW VERTEX ALGORITHM [4] (also known as the GASS-SAATY RULE [21]), for which the auxiliary function $\mathbf{c}$ is deterministically obtained, depending on the starting vertex. Borgwardt [4] has successfully analyzed this algorithm under the assumption that P is random in a suitable model (where the secondary objective function $\mathbf{c}$ can be fixed arbitrarily), and obtained polynomial upper bounds for the expected number of simplex steps.

None of the available evidence contradicts the possibility that the expected running time of all three randomized algorithms we consider is bounded from above by a polynomial, even a quadratic function, in n and m. (But see [6].) In this connection, we report investigations on the performance of such algorithms on infinite families of "test problems": specific linear programs which have decreasing paths of exponential length.

It is not generally believed that polynomial upper bounds can be achieved; it is equally conceivable that subexponential bounds such as those by Kalai [16] are essentially best possible. An interesting open problem in this context is to find linear programs on which the algorithms in [16] and [25] actually behave superpolynomially; Matoušek [24] has constructed a class of abstract optimization problems — more general than linear programs — for which the subexponential analysis is tight. For all actual linear programs in Matoušek's class, however, a polynomial (in fact quadratic) upper bound holds [9].

In this paper we concentrate on the analysis of the "Klee-Minty cubes," see Section 2. These are very interesting linear programs whose polytope is a deformed n-cube, but for which some pivot rules follow a path through all the vertices and thus need an exponential number of steps.

Our main results are quadratic, respectively nearly quadratic, lower bounds for the expected number of steps taken by the RANDOM-FACET and the RANDOM-EDGE simplex algorithms. For the RANDOM-EDGE rule this seems to be the first superlinear bound. Specifically, our analysis of random pivots on the Klee-Minty cubes yields the following two theorems.

Theorem 1. *The* RANDOM-FACET *simplex algorithm on the n-dimensional Klee-Minty cube, started at the vertex $\overline{x}$ "opposite" (on the n-cube) to the optimal vertex, takes a quadratic expected number of steps $F_n(\overline{x})$:*

$$F_n(\overline{x}) = n + 2\sum_{k=1}^{n} \frac{(-1)^{k+1}}{k+2}\binom{n-k}{2} \approx \left(\frac{\pi}{4} - \frac{1}{2}\right) n^2.$$

Moreover, for a random starting vertex the expected number of steps is

$$F_n := \frac{1}{2^n}\sum_{x} F_n(x) = \frac{n^2+3n}{8}.$$

We will see that one gets a linear lower bound and a quadratic upper bound $F_n(x) \leq \frac{n^2+3n}{4}$ for the expected number of steps from an arbitrary starting vertex x. Furthermore, there are starting points in the upper facet for which the RANDOM-FACET rule will take only linearly many steps. The fact that for some starting vertices the expected number of steps is quadratic follows from an explicit formula for the expectation value, given in Section 2, or from the bound for a random starting vertex.

A result very similar to Theorem 1, in the setting of dual simplex algorithms, was earlier obtained by Matoušek [24], who analyzed the behavior of the Matoušek-Sharir-Welzl dual simplex algorithm on a special class of linear programs.

Similarly, for RANDOM-EDGE one gets an upper bound $E_n(x) \leq \binom{n+1}{2}$ for the expected number of steps starting at any vertex x of the n-dimensional Klee-Minty cube, see Section 2. This was first observed by Kelly [18], see also [32].

Theorem 2. *The expected number E_n of steps that the* RANDOM-EDGE *rule will take, starting at a random vertex on the n-dimensional Klee-Minty cube, is bounded by*

$$\frac{n^2}{4(H_{n+1}-1)} \leq E_n \leq \binom{n+1}{2},$$

where $H_n = 1 + 1/2 + \ldots + 1/n$ is the n-th harmonic number.

The superlinear lower bound requires substantially harder work, see Section 3. It implies that there is a vertex x with $E_n(x) = \Omega(n^2/\log n)$, but compared to the case of RANDOM-FACET we are not able to show this bound for a specific starting vertex, e.g. the top vertex.

Our proof is based on a combinatorial model for the Klee-Minty cubes, which describes the RANDOM-EDGE algorithm as a random walk on an acyclic directed graph, see Section 2. (This model was first, it seems, derived by Avis & Chvátal [2].)

The combinatorial model also makes it possible to do simulation experiments. Our tests in the range $n \leq 1,000$ suggested that the $O(n^2)$ upper bound is close to the truth, although the constant $1/2$ derived from Theorem 2 is an overestimate. Motivated by this observation, we show in Section 3 that one can improve on the constant.

Also, it seems that a (nearly) quadratic lower bound is valid also if the starting vertex is chosen to be the top vertex of the program, but as mentioned above, our method does not prove this.

Still, our result contradicts Exercise 8.10* in [29, p. 188], where it is claimed that $E_n(x) = O(n)$. It also disproves a conjecture of Kelly [18] that $E_n(x) = O(n(\log n)^2)$ for all starting vertices x.

In contrast to these results, we show in Section 4 that the average length Φ_n of a decreasing path from the highest to the lowest vertex in the n-dimensional Klee-Minty cube — taking all paths with equal probability — satisfies $\Phi_n > (1 + 1/\sqrt{5})^{n-1}$: it is exponential. Thus, the "average" path is exponentially long, but the RANDOM-EDGE and RANDOM-FACET pivot rules take the long paths with low probability.

Another conjecture of Kelly [18], about the "worst starting vertex" for the RANDOM EDGE algorithm, also turned out to be false. Kelly had conjectured that the expected number of RANDOM-EDGE pivots is maximal if the starting vertex is the vertex $\overline{x} = (1, 1, \ldots, 1, 1)^t$ — with 0/1-coordinates associated to the vertices of the Klee-Minty in the standard way reviewed below — that is diametrically opposite to the lowest vertex. We found by explicit computation of expectation values (in rational arithmetic, using REDUCE) that the smallest dimension in which this fails is $n = 18$. Here one has

$$E(\overline{x}') \approx 54.547655 \quad > \quad E(\overline{x}) \approx 54.547444$$

for the vertex $\overline{x}' = (0, 1, \ldots, 1, 1)^t$, which is adjacent to the vertex $\overline{x}$. Floating-point computations (with rounding errors) up to dimension $n = 24$ indicate that one has this effect in even dimensions $n \geq 18$, while $\overline{x}$ seems to be the worst starting vertex in all odd dimensions.

The RANDOM-SHADOW algorithm has not yet been analyzed on special programs. Murty [28] and Goldfarb [12] have constructed variants of the Klee-Minty cubes for which the deterministic SHADOW VERTEX ALGORITHM takes an exponential number of steps.

There is hope for a successful analysis since Borgwardt's work [4] shows that methods of integral geometry can be very powerful when applied in this context.

Besides the Klee-Minty cubes and their variants, there are other natural classes of "test problems" for (randomized) linear programming algorithms. They include the deformed products of Klee & Minty [22], for which a combinatorial model is produced in Section 5. Also there is a natural model on polars of cyclic polytopes, for which the actual program has not been constructed, yet. This relates to the unsolved "upper bound problem for linear programs."

2 Combinatorial Models

The *Klee-Minty cubes* [22] [29] are the polytopes of the linear programs in $\mathbb{R}^n$ with $m = 2n$ facets given by

$$\begin{aligned} \min\ & x_n : \\ & 0 \leq x_1 \leq 1, \\ & \varepsilon x_{i-1} \leq x_i \leq 1 - \varepsilon x_{i-1} \end{aligned}$$

for $2 \leq i \leq n$ and $0 < \varepsilon < 1/2$. Our illustration shows the 3-dimensional Klee-Minty cube for $\varepsilon = 1/3$.

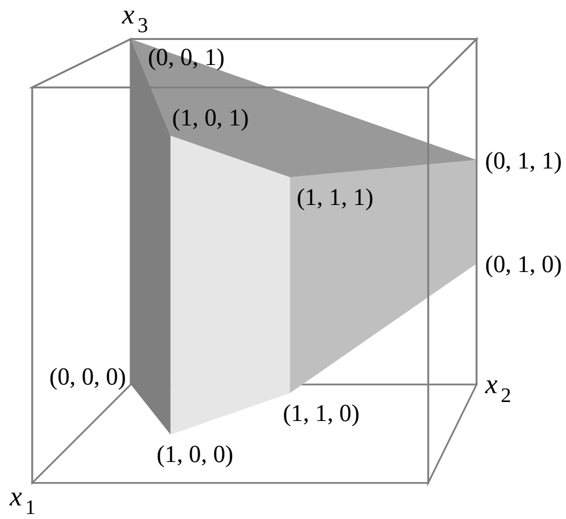

Figure 1: Klee-Minty cube for $n = 3, \varepsilon = 1/3$

Considering the geometry in the limit $\varepsilon \to 0$, one sees that the feasible region is a (slightly) deformed unit cube. Thus the feasible vertices of the program are in bijection with the set $\{0,1\}^n$ of all 0/1-vectors of length n, where we obtain the 0/1-vector for any vertex by rounding the coordinates. Two vertices are adjacent if the corresponding 0/1-vectors differ in exactly one coordinate. (The identification of $\{0,1\}$ with $GF(2)$ will turn out useful in the next section, where linear algebra over $GF(2)$ is a key tool in our approach to lower bounds.)

In the following, we identify the vertices of the Klee-Minty cubes with the corresponding 0/1-vectors. Since the simplex algorithm proceeds along decreasing edges, we have to describe the edge orientations. We claim that if $x, x' \in \{0,1\}^n$ differ in their i-th component, then the corresponding edge is directed from x to x' if and only if the sum $x_i + x_{i+1} + \ldots + x_n$ is odd. We write $x \to x'$ in this situation. To prove the claim, we first note that at a vertex, every coordinate of the program either matches its lower or its upper bound, therefore the value of the n-th coordinate at the vertex (its height) can be written as a linear expression in the i-th coordinate, the latter appearing with sign

$$(-1)^{x_{i+1}+\ldots+x_n}$$

in this expression. Consequently, the height gets smaller by going from x to x' if and only if either the i-th coordinate gets smaller (i.e. $x_i = 1$ in the corresponding vector) and $x_{i+1} + \ldots + x_n$ is even, or the i-th coordinate gets larger (i.e. $x_i = 0$) and $x_{i+1} + \ldots + x_n$ is odd. This in turn is equivalent to $x_i + x_{i+1} + \ldots + x_n$ being odd.

This completes the description of the combinatorial model: a directed, acyclic graph with 2^n vertices, $n2^{n-1}$ directed arcs, and a unique source and sink. It can be used as a combinatorial model for the linear program.

The RANDOM-EDGE algorithm moves on the digraph of the Klee-Minty cube by leaving the current vertex, using one of the outgoing edges with equal probability, until it reaches the unique sink in the digraph. For example, a legal sequence of steps for $n = 3$, starting at the highest vertex and ending at the lowest, is given by

$$\begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{1} \end{pmatrix} \longrightarrow \begin{pmatrix} 1 \\ \mathbf{0} \\ \mathbf{1} \end{pmatrix} \longrightarrow \begin{pmatrix} \mathbf{1} \\ 0 \\ 0 \end{pmatrix} \longrightarrow \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}.$$

Here any coordinate that can be flipped is typeset bold: from this one can read off that the first step is taken with probability $p = 1/3$, the second one with $p = 1/2$, and the third with probability 1. Thus this path is taken with probability $1/6$.

The expected number of steps $E_n(x)$ from a vertex x to the lowest vertex satisfies the recursion

$$E_n(x) = 1 + \frac{1}{\#\{x' : x \to x'\}} \sum_{x' : x \to x'} E_n(x').$$

If $i(x)$ denotes the highest index i for which $x_i = 1$, then we can easily show

$$i(x) \leq E_n(x) \leq \binom{i(x)+1}{2} \leq \binom{n+1}{2}.$$

This implies the upper bound of Theorem 2, but only a linear lower bound. A complete analysis seems to be surprisingly difficult. In Section 3 we develop a method, based on linear algebra over $GF(2)$, that yields the nearly quadratic lower bounds "on average" of Theorem 2.

The RANDOM-FACET pivot rule can, however, be completely analyzed on the Klee-Minty cubes. For this, one first derives that

$$F_n(\mathbf{e}_i) = F_n(\mathbf{e}_i + \mathbf{e}_{i-1}) = i.$$

In particular, started at the highest vertex $\mathbf{e}_n$, the RANDOM-FACET rule only needs an expected number of $F_n(\mathbf{e}_n) = n$ steps. For an arbitrary starting vertex $x \in \{0,1\}^n$, the solution of the program restricted to a facet $x_i = 0$ delivers the lowest vertex; restricted to a facet $x_i = 1$ the algorithm yields the vector $\mathbf{e}_i + \mathbf{e}_{i-1}$, where we set $\mathbf{e}_0 = \mathbf{0}$. From this we get a recursion

$$F_n(x) = \frac{1}{n}\left(\sum_{i=1}^{n} i x_i + \sum_{i=1}^{n} F_{n-1}(x^{(i)})\right),$$

with $x^{(i)} := (x_1, \ldots, x_{i-2}, x_{i-1} + x_i, x_{i+1}, \ldots, x_n)^t \in \{0,1\}^{n-1}$ for $1 \le i \le n$. Using this recursion, it is easy to derive a linear lower bound and a quadratic upper bound for $F_n(x)$, namely

$$i(x) \le F_n(x) \le \frac{i(x)^2 + 3i(x)}{4} \le \frac{n^2 + 3n}{4}.$$

Equality in the linear lower bound holds for the infinite series of vectors $\mathbf{e}_i$ and $\mathbf{e}_i + \mathbf{e}_{i-1}$. Surprisingly, one can explicitly solve the above recursion. In particular, quadratic lower bounds as in Theorem 1 can be derived from the following result.

Proposition 3. *Started at a vertex $x \in \{0,1\}^n$ of the n-dimensional Klee-Minty cube, with*

$$\{t : x_t = 1\} = \{s_1, s_2, \ldots, s_k\}, \qquad s_1 < s_2 < \ldots < s_k,$$

the expected number of steps of the RANDOM-FACET *simplex algorithm is*

$$F_n(x) = \sum_{i=1}^{k} s_i + 2 \sum_{1 \le i < j \le k} \frac{(-1)^{j-i} s_i}{s_j - s_i + 1}.$$

To prove this, one can verify that the formula for $F_n(x)$ does, indeed, satisfy the recursion. This, however, leads to very tedious computations and does not give extra insight. The proof is therefore omitted here.

For a random starting vertex, the situation is substantially simpler: Let

$$G_n = \sum_{x \in \{0,1\}^n} F_n(x).$$

From the recursion we get $G_n = 2^{n-2}(n+1) + 2G_{n-1}$ with $G_1 = 1$. This yields $G_n = 2^{n-2}(\binom{n+2}{2} - 1)$, and the second part of Theorem 1 follows.

3 Bounds for RANDOM-EDGE

Our analysis of the RANDOM-EDGE rule on the Klee-Minty cubes starts with a coordinate transformation in $\mathcal{V} := GF(2)^n$. Namely, we associate with every vertex $x \in \mathcal{V}$ the label

$$Tx := (x_n, x_n + x_{n-1}, \ldots, x_n + x_{n-1} + \ldots + x_1)^t \in \mathcal{V}.$$

With these new labels, the vertex set of the digraph is again given by $\mathcal{V}$. An arc of the digraph now corresponds to vertices $x, x' \in \mathcal{V}$ such that $x_i = 1$ and x' arises from x by replacing x_j by $x_j + 1 (\bmod 2)$ for all $j \ge i$. (In particular, this yields $x'_i = 0$.)

Thus, for any vector $x \in \mathcal{V}$, we consider the *game* KM(x):

> Choose a random coordinate r for which $x_r = 1$, and flip this coordinate together with all coordinates of higher index. This operation is repeated until the zero vector is reached.

For example, the flipping sequence considered in Section 2 corresponds, after this coordinate transformation, to the sequence

$$\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \longrightarrow \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \longrightarrow \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \longrightarrow \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}.$$

The version in which we prove the lower bound of Theorem 2 in this section is the following: starting with a random vector $x \in \mathcal{V}$, the expected number $L(x)$ of *rounds played* is at least $cn^2/\log n$ for some constant $c > 0$.

The flipping operation. The flip at index r (in the new coordinate system) can conveniently be expressed as a linear transformation over $\mathcal{V}$, i.e., there is a matrix $A^{(r)}$ such that

$$x^r := (x_1, \ldots, x_{r-1}, 0, x_{r+1} + x_r, \ldots, x_n + x_r)^t = A^{(r)}x$$

for all vectors $x = (x_1, \ldots, x_n)$.

The columns of $A^{(r)}$ are the images of the unit vectors under the flip at r, i.e.

$$A^{(r)} = \begin{array}{cl} \begin{array}{ccccccc} & & & \downarrow \text{column } r & & & \end{array} & \\ \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & 0 & & & \\ & & & 1 & 1 & & \\ & & & \vdots & & \ddots & \\ & & & 1 & & & 1 \end{pmatrix} & \begin{array}{l} \\ \\ \\ \leftarrow \text{row } r \\ \\ \\ \\ \end{array} \end{array},$$

and all other entries are zero. Note that for $j \neq r$, ${\mathbf{e}_j}^r = \mathbf{e}_j$; in general, a flip with $x^r = x$ is called *void*, and although $\mathrm{KM}(x)$ does not perform void flips, this more general flipping concept is a key ingredient in our approach.

Flip sequences. Let $\mathcal{S}$ be the set of all (formally infinite) sequences $(s_1, s_2, \ldots)$ with elements in $\{1, \ldots, n\}$, where $\mathrm{prob}_{\mathcal{S}}(s_k = r) = 1/n$ independently for all k. We refer to the members of $\mathcal{S}$ as *flip sequences*. For a flip sequence s and an integer k we let $x^{(s,k)}$ be the result of 'applying' the first k flips of s to x, i.e.,

$$x^{(s,k)} := A^{(s,k)}x, \text{ with } A^{(s,k)} := A^{(s_k)} \cdots A^{(s_2)}A^{(s_1)}.$$

The analysis of game KM. It is clear that one can simulate game KM by flipping with a random $r \in \{1, \ldots, n\}$ in each step and ignoring the void flips. This means that the expected length $L(x)$ of game $\mathrm{KM}(x)$ is just the expected number of nonvoid flips encountered during the simulation. Using the linearity of expectation, this boils down to the following formula:

$$L(x) = \sum_{k\geq 1} \operatorname*{prob}_{\mathcal{S}}(x^{(s,k)} \neq x^{(s,k-1)}).$$

Recalling that the expectation of a nonnegative, integer-valued random variable X can be written as $\sum_{k\geq 0} \operatorname{prob}(X > k)$, we see that

$$L^*(x) := \sum_{k\geq 0} \operatorname*{prob}_{\mathcal{S}}(x^{(s,k)} \neq \mathbf{0})$$

is just the expected length of the simulation, including the void flips — this will be important later. Let us refer to the simulation as game KM*.

Recall that $x^{(s,k)} \neq x^{(s,k-1)}$ if and only if the k-th flip hits a 1-entry of the current vector $x^{(s,k-1)}$, which implies that the probability for a nonvoid k-th flip is just the expected number of 1-entries in $x^{(s,k-1)}$, divided by n. Thus

$$\operatorname*{prob}_{\mathcal{S}}(x^{(s,k)} \neq x^{(s,k-1)}) = \frac{1}{n}\sum_{r=1}^{n} \operatorname*{prob}_{\mathcal{S}}(x^{(s,k-1)}{}_r = 1).$$

Let

$$L(n) := \frac{1}{2^n}\sum_{x\in\mathcal{V}} L(x)$$

be the *average* expected length (over all vectors x) of game KM. We obtain

$$\begin{aligned}
L(n) &= \sum_{k\geq 1}\frac{1}{n}\sum_{r=1}^{n} \operatorname*{prob}_{\mathcal{S},\mathcal{V}}(x^{(s,k-1)}{}_r = 1) \\
&= \frac{1}{n}\sum_{r=1}^{n}\sum_{k\geq 1} \operatorname*{prob}_{\mathcal{S},\mathcal{V}}((A^{(s,k-1)}x)_r = 1) \\
&= \frac{1}{n}\sum_{r=1}^{n}\sum_{k\geq 1} \operatorname*{prob}_{\mathcal{S},\mathcal{V}}((\mathbf{e}_r{}^t A^{(s,k-1)})x = 1) \\
&= \frac{1}{2n}\sum_{r=1}^{n}\sum_{k\geq 1} \operatorname*{prob}_{\mathcal{S}}(\mathbf{e}_r{}^t A^{(s,k-1)} \neq \mathbf{0}),
\end{aligned}$$

since $\operatorname{prob}_{\mathcal{V}}((\mathbf{e}_r{}^t A^{(s,k-1)})x = 1)$ is equal to 1/2 if $\mathbf{e}_r{}^t A^{(s,k-1)} \neq \mathbf{0}$ (and 0 otherwise).

In general, $x^t A^{(s,k-1)}$ arises from x by playing $k-1$ rounds of another flipping game: choose in each round a random index r and replace x_r by $\sum_{r'>r} x_{r'} = x_r + \sum_{r'\geq r} x_{r'}$. But wait! This flipping game is nothing else than the simulation KM*, played in the original combinatorial model of the previous section: if $\sum_{r'\geq r} x_{r'}$ is odd, x_r gets flipped, inducing a proper round of KM* — otherwise nothing happens, and a void flip occurs in the simulation. (This correspondence is not a magic coincidence. See [10, Chapter 6] for a formal derivation in a more general setting.) Putting together this observation and the previous derivation gives a simple relation between the average expected length $L(n)$ of game KM and the expected length L^* of game KM* for specific starting vectors.

Lemma 4.

$$L(n) = \frac{1}{2n} \sum_{r=1}^{n} L^*(T\mathbf{e}_r),$$

where $T : x \mapsto (x_n, x_n + x_{n-1}, \ldots, x_n + x_{n-1} + \ldots + x_1)^t$ is the coordinate transformation relating the two combinatorial models of the Klee-Minty cubes.

Bounding L^*. Lemma 4 leaves us with the problem of determining how many flips (void or nonvoid) are necessary on the average to reduce the vector

$$T\mathbf{e}_r = (0, \ldots, 0, \underbrace{1, \ldots, 1}_{r})$$

to $\mathbf{0}$. To this end we will analyze how the vector evolves when applying a random flip sequence to it. Actually, the analysis will only trace the *dimension* which records the 1-entry with smallest index. Therefore, the considerations for $T\mathbf{e}_r$ are valid as well for any other vector x with the same leftmost 1-entry.

Definition 5. For a nonzero vector $x \in \mathcal{V}$, the dimension of x is the number

$$d(x) := n + 1 - \min\{r \mid x_r = 1\}.$$

Furthermore, $d(\mathbf{0}) := 0$.

For example, $d(T\mathbf{e}_r) = r$. Now define for $0 \leq d \leq n$ the numbers

$$l(d) := \min_{x \in \mathcal{V}} \{L^*(x) \mid d(x) = d\}.$$

We get $l(0) = 0$ and $L^*(T\mathbf{e}_r) \geq l(r)$, and our objective will be to bound $l(d)$ from below. To this end fix any vector x with dimension $d > 0$ and $L^*(x) = l(d)$, and apply a random flip sequence to it. Eventually the sequence will hit the leading 1-entry, thereby decreasing the dimension of the vector currently under consideration. The expected number of flips performed until this happens is exactly n (we have a sequence of Bernoulli-trials with probability of success equal to $1/n$ independently in every trial). The expected number of flips performed after the dimension has decreased depends on the actual dimension obtained. For $i < d$ let p_i denote the probability that the dimension goes down from d to i. Then

$$l(d) \geq n + \sum_{i=0}^{d-1} p_i l(i). \tag{1}$$

Lemma 6. *Let $i < d$. Then*

$$p_0 + \ldots + p_i \leq \frac{1}{d - i}.$$

Proof. The sum $p_0 + \ldots + p_i$ is the probability that the dimension goes down by at least $d - i$, and if this event is possible at all, the flip sequence must necessarily hit the leading 1-entry before it hits any of the $d - i - 1$ next higher indices — otherwise there is a 0-entry at the smallest such index which was hit, and this entry turns into one by the time the leading position is flipped, preventing the dimension from advancing by more than $d - i - 1$. However, the probability of hitting the leading 1-entry first is exactly $1/(d - i)$. □

To proceed further, we need a simple fact.

Lemma 7. *$l(d)$ is monotone increasing with d.*

Proof. As above, fix some vector

$$x = (\underbrace{0, \ldots, 0,}_{n-d} 1, x_{n-d+2}, \ldots, x_n)$$

of dimension $d > 0$ with $L^*(x) = l(d)$ and consider the right-shifted vector

$$x' = (\underbrace{0, \ldots, 0,}_{n-d+1} 1, x_{n-d+2}, \ldots, x_{n-1})$$

of dimension $d - 1$. We claim that $L^*(x) \geq L^*(x')$ holds which proves the lemma. To see this, consider the following bijection $S \leftrightarrow S'$ between flip sequences. $S = s_1, s_2, \ldots$ is mapped to $S' = s'_1, s'_2, \ldots$ defined by

$$s'_k := \begin{cases} s_k, & \text{if } s_k = 1, \ldots, n - d \\ s_k + 1, & \text{if } s_k = n - d + 1, \ldots, n - 1 \\ n - d + 1, & \text{if } s_k = n \end{cases}$$

Now, if playing the game with S reduces x to zero, then playing the game with S' reduces x' to zero. Consequently, on the average over all S, the game on x does not end before the game on x'. □

From the monotonicity it follows that the right hand side of (1) is minimized if the tuple $(p_{d-1}, \ldots, p_0)$ is lexicographically smallest subject to $\sum_{i=0}^{d-1} p_i = 1$ and the inequalities established by Lemma 6. This is the case if $p_i = 1/(d - i) - 1/(d - i + 1)$ for $i > 0$, $p_0 = 1/d$. Recalling that $l(0) = 0$, we get

Lemma 8.

$$l(d) \geq n + \sum_{i=1}^{d-1} \left(\frac{1}{d - i} - \frac{1}{d - i + 1}\right) l(i).$$

Theorem 9.

$$\sum_{d=1}^{n} l(d) \geq \frac{n^3}{2(H_{n+1} - 1)}.$$

Proof. The inequality of Lemma 8 can be rewritten as

$$\sum_{i=1}^{d} \frac{l(i)}{d-i+1} \geq n + \sum_{i=1}^{d-1} \frac{l(i)}{d-i},$$

and after setting $f(d) := \sum_{i=1}^{d} l(i)/(d-i+1)$ reads as $f(d) \geq n + f(d-1)$ with $f(0) = 0$. This implies $f(d) \geq dn$ for all $d \leq n$, so

$$\sum_{i=1}^{d} \frac{l(i)}{d-i+1} \geq dn.$$

Summing up the inequalities for all values of d up to n gives

$$\begin{aligned} \binom{n+1}{2} n &\leq \sum_{d=1}^{n} \sum_{i=1}^{d} \frac{l(i)}{d-i+1} \\ &= \sum_{i=1}^{n} l(i) \sum_{d=i}^{n} \frac{1}{d-i+1} \\ &= \sum_{i=1}^{n} l(i) H_{n-i+1}. \end{aligned} \tag{2}$$

While $l(i)$ increases with i, H_{n-i+1} decreases, and Chebyshev's "up-down" summation inequality [14, (2.34)] can be applied to yield

$$\begin{aligned} \sum_{i=1}^{n} l(i) H_{n-i+1} &\leq \frac{1}{n} \left(\sum_{i=1}^{n} l(i) \right) \left(\sum_{i=1}^{n} H_{n-i+1} \right) \\ &= \frac{1}{n} \left(\sum_{i=1}^{n} l(i) \right) (n+1)(H_{n+1} - 1). \end{aligned} \tag{3}$$

Putting together (2) and (3) then gives

$$\frac{n^3}{2} \leq (H_{n+1} - 1) \sum_{i=1}^{n} l(i),$$

as claimed. ∎

Putting everything together. From Lemma 4 we know that

$$L(n) = \frac{1}{2n} \sum_{r=1}^{n} L^*(T\mathbf{e}_r) \geq \frac{1}{2n} \sum_{r=1}^{n} l(r),$$

and by Theorem 9 we can argue that

$$L(n) \geq \frac{1}{2n} \frac{n^3}{2(H_{n+1} - 1)} = \frac{n^2}{4(H_{n+1} - 1)},$$

which finally implies Theorem 2.

We remark that within a factor of 2, the bound of Theorem 9 is the best one can deduce from the recurrence of Lemma 8. Namely, starting with equality in Lemma 8, we arrive at

$$\sum_{d=1}^{n} \frac{l(d)}{n-d+1} = n^2.$$

This time both $l(d)$ and $1/(n-d+1)$ increase with d, and Chebyshev's "up-up" summation inequality [14, (2.34)] gives

$$n^2 = \sum_{d=1}^{n} \frac{l(d)}{n-d+1} \geq \frac{1}{n}\left(\sum_{d=1}^{n} l(d)\right)\left(\sum_{d=1}^{n} \frac{1}{n-d+1}\right) = \frac{1}{n}\left(\sum_{d=1}^{n} l(d)\right) H_n.$$

This implies

$$\sum_{d=1}^{n} l(d) \leq \frac{n^3}{H_n}.$$

Thus, in order to beat the bound of Theorem 9 (and prove e.g. that $\sum_{d=1}^{n} l(d) = \Omega(n^3)$), one will have to keep track of more information than the dimension of vectors during the flipping process.

An improved upper bound. We conclude this section by exhibiting an upper bound on $E_n(x)$ that beats the bound of $\binom{n+1}{2} \approx 1/2\, n^2$ established in Theorem 2 by a constant factor. The aim is to show that the easy bound is not the truth, but we do not put emphasis on extracting the best possible factor.

Let $p_2(x)$ be the probability that the leftmost one-entry of x gets flipped (equivalently, the dimension decreases) within two rounds of game KM(x), and define

$$p_2(d) := \min_{x \in \mathcal{V}} \{p_2(x) \mid d(x) = d\}.$$

Then the expected number of rounds played before the dimension decreases is bounded by $2/p_2(d)$, and we get

$$E_n(x) \leq \sum_{d=1}^{n} \frac{2}{p_2(d)},$$

for any start vector x. Let s be the number of one-entries in a fixed vector x with $p_2(x) = p_2(d)$. Then we have the following lower bound on $p_2(d)$.

$$p_2(d) \geq \frac{1}{s}\left(1 + \sum_{i=1}^{s-1} \frac{1}{d-i}\right).$$

To see this note that the first flip decreases the dimension with probability exactly $1/s$; with the same probability, it hits the i-th one-entry from the right, for $i = 1, \ldots, s-1$, and in this case, at least i zero-entries are generated by the flip. This means, for the subsequent second round, that the probability of decreasing the dimension is at least $1/(d-i)$. We further compute

$$p_2(d) \geq \frac{1}{s}\left(1+\sum_{j=d-s+1}^{d-1}\frac{1}{j}\right) \geq \frac{1}{s}\left(1+\int_{j=d-s+1}^{d}\frac{1}{x}dx\right) = \frac{1}{s}\left(1+\ln\frac{d}{d-s+1}\right).$$

After taking derivatives, we see that this expression is minimized if

$$\frac{d}{d-s+1} = a_d,$$

where a_d is the solution of the equation

$$2+\ln a_d = \frac{d+1}{d}a_d.$$

Using the value $s^* \in \mathbb{R}$ defined by this gives

$$p_2(d) \geq \frac{1}{s^*}\left(1+\ln\frac{d}{d-s^*+1}\right) = \frac{a_d}{d}.$$

For $d \to \infty$, a_d converges to the value $a = 3.146\ldots$ satisfying $2+\ln a = a$, so that we obtain the asymptotic result

$$E_n(x) \leq \sum_{d=1}^{n}\frac{2}{p_2(d)} \leq \sum_{d=1}^{n}\frac{2d}{a_d} \approx \frac{2}{a}\sum_{d=1}^{n} d \approx .32\ n^2.$$

Further massage along these lines improves the constant to .27, but since we do not expect to get the optimal bound that way, we won't elaborate on this.

4 The average length of decreasing paths

The RANDOM-EDGE and the RANDOM-FACET algorithms both have (easy) quadratic upper bounds on a Klee-Minty cube, that is, the *expected length* of the path chosen by such an algorithm is polynomially bounded. In sharp contrast to this the *average length* of a simplex path (from the highest to the lowest vertex) is exponentially large in this situation, as we will establish in this section. This points to an important and strong imbalance in the way how randomized pivot rules select the paths they follow: most paths are exponentially long, but with high probability the randomized pivot rules choose short paths. At least this is true on the Klee-Minty cubes…

The length of a (decreasing) path from the highest vertex $(0,\ldots,0,1)^t$ to the lowest vertex $(0,\ldots,0)^t$ in the n-dimensional Klee-Minty cube is always an odd integer. Let $\phi(k,n)$ denote the number of all such paths of length $2k-1$. We have $\phi(1,n) = \phi(2^{n-1},n) = 1$ and for $k<1$ or $k>2^{n-1}$ we set $\phi(k,n)=0$. In order to estimate the average length of a decreasing path

$$\Phi_n = \sum_{k=1}^{2^{n-1}}(2k-1)\cdot\phi(k,n)\Big/\sum_{k=1}^{2^{n-1}}\phi(k,n)$$

(where all paths are taken with equal probability) we first prove the following recursion, for which we give a combinatorial (bijective) proof.

Lemma 10. *For $n \geq 2$ and all $k \in \mathbb{Z}$ one has*

$$\phi(k,n) \quad = \quad \sum_j \binom{k}{2j} \phi(k-j, n-1).$$

Proof. In the 0/1-model that we derived in Section 2, consider any path from the maximal vertex $(0,0,\ldots,0,1)^t$ to the minimal vertex $(0,0,\ldots,0)^t$. This path contains exactly one flip in the last coordinate (this coordinate is 1 at the start, and 0 at the end), but an even number of flips in every other coordinate (where we start and end with 0). Thus we have an odd number $2k-1$ of flips in total, and an even number $2j$ of flips in the first coordinate.

From the rules of the game we see that if we just *delete* the first coordinates from all vectors, then — if we ignore the void flips — we get a legal flip sequence, of length $(2k-1)-2j$, in the $(n-1)$-dimensional binary Klee-Minty cube.

When we try to insert the $2j$ 1-flips (i.e., flips of the first coordinate) into a given sequence of $(2k-1)-2j$ flips for the $(n-1)$-dimensional cube, then one always has to keep an *odd* number of low-dimensional flips between two successive 1-flips.

Our aim is to show that there are exactly $\binom{k}{2j}$ ways to do that, i.e., to derive a legal sequence of $2k-1$ flips in the n-dimensional cube from a legal sequence of $2k-1-2j$ flips for the $(n-1)$-cube. For this write any sequence of flips for the n-dimensional cube as a string of the form

$$1 * * * 1 * 1 * * * 1 * 1 * * * 1 * *$$

where $*$ represents an arbitrary flip not in the first coordinate. (In our example we have $k=9$ and $j=3$.) Every *legal* sequence of this form can be grouped in the form

$$\overline{1*}\,\overline{**}\,\overline{1*}\,\overline{1*}\,\overline{**}\,\overline{1*}\,\overline{1*}\,\overline{**}\,1\,\overline{**}$$

with $2j-1$ pairs of the form $\overline{1*}$ (i.e., a 1-flip followed by a different flip), one singleton 1-flip, and $\frac{1}{2}((2k-1)-2j-(2j-1)) = k-2j$ pairs $\overline{**}$ corresponding to two "other flips." Thus we have to determine the positions of the $2j$ different blocks involving 1 among k blocks altogether: so there are $\binom{k}{2j}$ possibilities. ∎

Here is an alternative interpretation of the family of $\binom{k}{2j}$ paths counted in the previous proof. The question is to count the number of directed paths of length $2k-1$ from the maximal vertex $s = (0,0,\ldots,0,1)^t$ to the minimal vertex $t = (0,0,\ldots,0,0)^t$ in a "ladder digraph" such as the one depicted in Figure 2, where the path along the top row would have length $2(k-j)-1$, but we are actually counting paths of length $2k-1$, which therefore must use $2j$ of the vertical edges.

One can prove an "explicit" formula for $\phi(k,n)$,

$$\phi(k,n) \quad = \sum_{\substack{d_1,\ldots,d_{n-1}\in\mathbb{N} \\ d_1+\cdots+d_{n-1}=k-1}} \prod_{i=1}^{n-1} \binom{1+\sum_{j=i}^{n-1} d_j}{2d_i}.$$

by induction on n. However, that will not be used in the following.

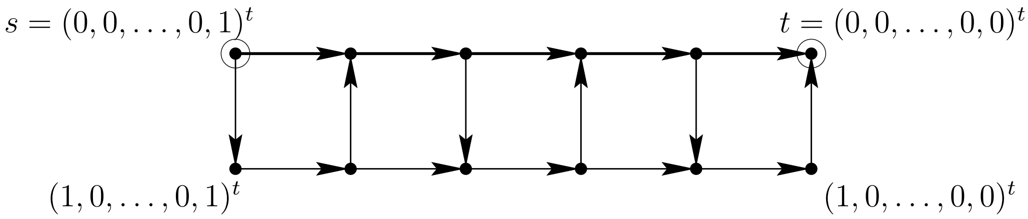

Figure 2: The ladder digraph for $k - 2j = 3$.

Theorem 11. *For $n \geq 2$ one has*

$$(1 + 1/\sqrt{5}) \cdot 2^{n-1} > \Phi_n > \left(1 + 1/\sqrt{5}\right)^{n-1}.$$

Proof. By Lemma (10) we get for the number of all paths

$$\sum_{k=1}^{2^{n-1}} \phi(k,n) = \sum_{k=1}^{2^{n-1}} \sum_j \binom{k}{2j} \phi(k-j, n-1) = \sum_{k=1}^{2^{n-2}} \phi(k, n-1) \sum_j \binom{k+j}{2j}.$$

In the same way we find

$$\sum_{k=1}^{2^{n-1}} (2k-1)\phi(k,n) = \sum_{k=1}^{2^{n-2}} \phi(k,n-1) \sum_j (2(k+j)-1) \binom{k+j}{2j}.$$

Let $F(m)$ denote the m-th Fibonacci number, i.e., $F(m)$ is defined by $F(0) = F(1) = 1$ and $F(m) = F(m-1) + F(m-2)$. One has $F(m) = \sum_{j \leq m} \binom{m-j}{j}$ [14, (6.130)] and hence $\sum_j \binom{k+j}{2j} = F(2k)$. With $G(k) := \sum_j (2(k+j)-1)\binom{k+j}{2j}$ we write

$$\Phi_n = \sum_{k=1}^{2^{n-2}} \phi(k, n-1) \cdot G(k) \Big/ \sum_{k=1}^{2^{n-2}} \phi(k, n-1) \cdot F(2k). \tag{4}$$

So we obtain the upper bound

$$\Phi_n \leq u(n) := \max\{G(k)/F(2k) : 1 \leq k \leq 2^{n-2}\}. \tag{5}$$

In order to find a lower bound we observe that $F(m)$ is a strictly increasing function of m and from this a simple calculation yields

$$\Phi_{n-1} = \frac{\sum_{k=1}^{2^{n-2}} (2k-1) \cdot \phi(k, n-1)}{\sum_{k=1}^{2^{n-2}} \phi(k, n-1)} < \frac{\sum_{k=1}^{2^{n-2}} (2k-1) \cdot \phi(k, n-1) \cdot F(2k)}{\sum_{k=1}^{2^{n-2}} \phi(k, n-1) \cdot F(2k)}$$

This implies by (4)

$$\Phi_n / \Phi_{n-1} > l(n) := \min\{G(k)/((2k-1) \cdot F(2k)) : 1 \leq k \leq 2^{n-2}\}. \tag{6}$$

It remains to estimate the functions $l(n)$ and $u(n)$. For this we prove that

$$G(k) \quad = \quad \frac{12k-5}{5}\,F(2k) \;+\; \frac{4k+2}{5}\,F(2k-1);$$

this is equivalent to

$$\sum_{j=0}^{k} 5j\binom{k+j}{2j} \quad = \quad k\,F(2k) \;+\; (2k+1)\,F(2k-1);$$

which can be verified by induction on k. Using the identity $F(k+1)F(k-1) - F(k)^2 = (-1)^{k+1}$ we see that $F(2k-1)/F(2k)$ is a strictly increasing function of k, with limit $(-1+\sqrt{5})/2$. Thus $G(k)/F(2k)$ is increasing and with (5) we get

$$\Phi_n \le u(n) = \frac{G(2^{n-2})}{F(2^{n-1})} < (1+1/\sqrt{5}) \cdot 2^{n-1}.$$

Now, it is not hard to verify that $G(k)/((2k-1)\cdot F(2k))$ is a strictly decreasing function and thus

$$l(n) > \lim_{k\to\infty} \frac{G(k)}{(2k-1)F(2k)} = \frac{6}{5} + \frac{2}{5}\lim_{k\to\infty}\frac{F(2k-1)}{F(2k)} = 1 + 1/\sqrt{5},$$

which with (6) shows the lower bound $\square$

In view of the following table it seems likely that $\Phi_n > 2^{n-1}$. Using generating function techniques, Bousquet-Mélou has now been able to prove the existence of a constant $C > 0$ such that $\Phi_n \ge C2^n$ [5].

n	2	3	4	5	6	7	8	9	10	11	12	13
$\frac{\Phi_n}{2^{n-1}}$	1	1.036	1.075	1.085	1.0875	1.0887	1.0893	1.0896	1.08981	1.08988	1.08992	1.08994

5 Related Models

In this final section, we provide two more combinatorial models for classes of linear programs with exponentially long decreasing paths. A main feature of these two classes — as compared to the Klee-Minty cubes — is that they include polytopes with arbitrarily large number of facets in any fixed dimension. In both classes, we can prove quadratic upper bounds for the running time of RANDOM-EDGE with arbitrary starting vertex.

Deformed products. This class of linear programs was also constructed by Klee & Minty [22]. Its polytopes are combinatorially equivalent to products of 1- and 2-dimensional polytopes. For the following, we restrict to the special case where the dimension n is even, and $P := (C_k)^{n/2}$ is a product of k-gons: an n-dimensional polytope with $m = \frac{kn}{2}$ facets. Such polytopes are now realized in $\mathbb{R}^n$ ("deformed") in such a way that they have an x_n-decreasing path through all the vertices. The tricky geometric construction of these programs [22] was simplified in [1]; the combinatorial model is very simple, as follows.

The vertex set of P can naturally be identified with the set of vectors $\{1,\ldots,k\}^{n/2}$. Two vertices are adjacent if their vectors $x, x' \in \{1,\ldots,k\}^{n/2}$ differ in a single coordinate, and in this coordinate the difference is either 1 or $k-1$. The directions of these edges are given as follows: if x and x' differ in their i-th coordinate, then we get a directed edge $x' \to x$ if either

- $x'_i > x_i$ and $(x_{i+1},\ldots,x_{n/2})$ contains an even number of even entries, or
- $x'_i < x_i$ and $(x_{i+1},\ldots,x_{n/2})$ contains an odd number of even entries.

This explicitly describes an acyclic digraph. The underlying graph is a product of cycles. It is oriented in a special way such that we get an acyclic digraph that has a (unique) Hamiltonian path that leads from the source ("maximal vertex") which is $s = (k,k,\ldots,k,k)$ if k is odd and $s = (1,1,\ldots,1,k)$ if k is even, to the sink ("minimal vertex") that is $t = (1,1,\ldots,1,1)$ in all cases. Our drawing shows this graph for $n = 4$, $k = 3$, where we have

$$33 \longrightarrow 23 \longrightarrow 13 \longrightarrow 12 \longrightarrow 22 \longrightarrow 32 \longrightarrow 31 \longrightarrow 21 \longrightarrow 11$$

as the directed path through all the vertices.

On this digraph pivoting algorithms such as RANDOM-EDGE take a random walk that will neccessarily end at the sink t.

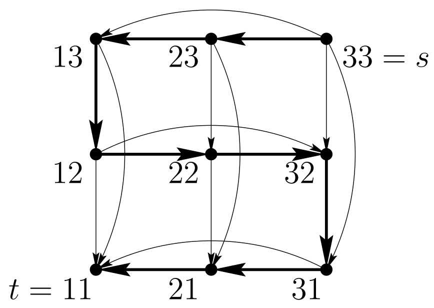

Figure 3: The direct product network for $n = 4$, $k = 3$

Proposition 12. *On a deformed product program with $k^{n/2}$ vertices and $m = kn/2$ facets, where $k \geq 3$, the expected number of steps taken by the* RANDOM-EDGE *algorithm, when starting at $x \in \{1,2,\ldots,k\}^{n/2}$, is bounded by a quadratic function,*

$$E(x) \quad \leq \quad E_{n,m} \;:=\; (k-1)^2\frac{n^2}{4} < \frac{nm}{2}.$$

Proof. Using induction (on the distance from the sink along the directed Hamilton path) we prove that

$$E(x) \quad \leq \quad E_{n-2,m} + (l-1)(n-1) \qquad \text{for } x_{n/2} = l. \tag{7}$$

This is true if $n = 2$ (with $E_{0,m} := 0$). Thus we may apply induction on n for the case $l = 1$. We use the notation $\nu(x) := \#\{x' : x \to x'\}$. For $1 < l < k$, every vertex x has exactly one "lower" neighbor x' with $x'_{n/2} < x_{n/2}$, and $\nu(x) - 1 \leq n - 2$ neighbors $x' \leftarrow x$ "at the same level" (i.e., $x'_{n/2} = l$). Thus we get

$$\begin{aligned} E(x) \quad &\leq \quad 1 + \frac{1}{\nu(x)}\Big(\nu(x)E_{n-2,m} + (\nu(x)-1)(l-1)(n-1) + (l-2)(n-1)\Big) \\ &= \quad E_{n-2,m} + (l-1)(n-1) + 1 - \frac{n-1}{\nu(x)} \\ &\leq \quad E_{n-2,m} + (l-1)(n-1). \end{aligned}$$

Similarly, for $l = k$ we see that x has one lower neighbor x' with $x'_{n/2} = 1$, one such neighbor x'' with $x''_{n/2} = k - 1$, and $\nu(x) - 2 \leq n - 2$ neighbors with last coordinate k, and thus we get

$$\begin{aligned} E(x) \quad &\leq \quad 1 + \frac{1}{\nu(x)}\Big(\nu(x)E_{n-2,m} + (\nu(x)-2)(k-1)(n-1) + (k-2)(n-1)\Big) \\ &= \quad E_{n-2,m} + (k-1)(n-1) + 1 - \frac{k(n-1)}{\nu(x)} \\ &\leq \quad E_{n-2,m} + (k-1)(n-1). \end{aligned}$$

Iterating the inequality (7) we obtain

$$E(x) \quad \leq \quad (k-1)(1+3+\ldots+(n-1)) \;=\; (k-1)\Big(\frac{n}{2}\Big)^2. \qquad \square$$

The function $E_{n,m}(x)$ is, however, not even completely analyzed for the case $n = 4$.

For the deformed products, there is always a (unique, decreasing) shortest path from the highest to the lowest vertex: it visits only these two vertices if k is odd, while it uses $\frac{n}{2} + 1$ vertices if k is even. In contrast to this very short path, the longest decreasing path visits all the $k^{n/2} = (\frac{2m}{n})^{n/2}$ vertices. In constant dimension this yields a longest decreasing path of length $O(m^{n/2})$, which is asymptotically sharp. However, for other interesting parameter settings, like $m = 2n$, there might be substantially longer paths — see the following construction.

Cyclic programs. Here the construction starts with the polars $C_n(m)^\Delta$ of cyclic polytopes [15] [34]. These simple polytopes have the maximal number of vertices for given m and n, namely

$$V(n,m) = \binom{m - \lceil \frac{n}{2} \rceil}{\lfloor \frac{n}{2} \rfloor} + \binom{m - 1 - \lceil \frac{n-1}{2} \rceil}{\lfloor \frac{n-1}{2} \rfloor},$$

according to McMullen's upper bound theorem [26] [34]. The facets of $C_n(m)^\Delta$ are identified with $[m] := \{1, 2, \ldots, m\}$; the vertices correspond to those n-subsets $F \subseteq [m]$ which satisfy "Gale's evenness condition": if $i, k \in [m]\backslash F$, then the set $\{j \in F : i < j < k\}$ has even cardinality.

Now any two ordered sets $F = \{i_1, i_2, \ldots, i_n\}_<$ and $G = \{j_1, j_2, \ldots, j_n\}_<$ satisfying Gale's evenness condition are compared by the following *twisted lexicographic order*: $F < G$ if and only if $i_1 < j_1$,
or $i_1 = j_1, \ldots, i_k = j_k, i_{k+1} < j_{k+1}$, and i_k is even,
or $i_1 = j_1, \ldots, i_k = j_k, i_{k+1} > j_{k+1}$, and i_k is odd.

Thus one compares the first element in which the (sorted) sets F and G differ, and takes the natural order if the element before is even (or doesn't exist), and the reversed order if the element before is odd. For example, for $C_4(8)^\Delta$ we get the ordering
$1678 < 1568 < 1458 < 1348 < 1238 < 1234 < 1245 < 1256 < 1267 < 1278 < 2378 < 2367 < 2356 < 2345 < 3456 < 3467 < 3478 < 4578 < 4567 < 5678$.

Now we use this ordering to construct the digraph model. Its vertices are the sets satisfying Gale's evenness condition. There is a directed edge $F \rightarrow F'$ if and only if $F' < F$ and F, F' differ in exactly one element, that is, the corresponding vertices of $C_n(m)^\Delta$ are adjacent.

The special property of the ordering is that every vertex is adjacent to the previous one. Thus the digraph is acyclic with unique source and sink, and with a directed path through all the vertices. (The construction is derived from Klee [19], where the order is constructed and described recursively.)

In general one *cannot* realize the polytope $C_n(m)^\Delta$ such that the x_n-coordinate orders the vertices according to twisted lexicographic order. (Equivalently, in general this order does not correspond to a Bruggesser-Mani shelling [7] [34] of some realization of the cyclic polytope. In fact, Carl Lee has observed that for $n = 7$ and $m = 10$ the twisted lexicographic order is not a shelling order.) Thus the following "upper bound problem for linear programs" is open:

> "What is the largest possible number $P(n, m)$ of vertices on a decreasing path in a linear program of dimension n with m facets?"

In other words, it is not clear whether the bound $P(n, m) \leq V(m, n)$, from the upper bound theorem for polytopes, holds with equality.

Even without such a realization, the twisted lexicographic ordering yields an interesting acyclic orientation of the graph of the polar cyclic polytope $C_n(m)^\Delta$. This digraph model may be a very reasonable "worst case" (?) scenario for the performance of randomized simplex algorithms. Both the RANDOM-EDGE and the RANDOM-FACET variants can, indeed, be analyzed in terms of this digraph model, without use of a metric realization.

Proposition 13. *For the* RANDOM-EDGE *rule, started at an arbitrary vertex F of the cyclic program, there is a linear lower bound and a quadratic upper bound for the expected number of steps. For this, we set $\ell(F) := m + 1 - \min(F)$, with $n \leq \ell(F) \leq m$, and obtain*

$$\ell(F) - n \quad \leq \quad E_{n,m}(x) \quad \leq \quad \binom{\ell(F)+1}{2} - \binom{n+1}{2}.$$

(The proof for this result is similar to that of Proposition 12, and thus omitted.)

Since both the diameter problem [23] [20] and the algorithm problem [27] [25] have upper bounds that are linear in m, it would be interesting to know that $E_{n,m}(x)$ indeed grows at most linearly in m for such problems. On the other hand, it is certainly challenging to strive for a nonlinear lower bound for these models.

Thanks. We wish to thank Carl Lee, Markus Reiss and Emo Welzl for helpful discussions, comments and contributions, and Claudia Betz-Haubold, Dietmar Kühl, Winfried Neun, Wolfram Schlickenrieder and Tuomo Takkula for extensive computations. We are indebted to Jiří Matoušek for a substantial simplification of the lower bound proof in Section 3. Finally, Noga Alon contributed to the improved upper bound of Section 3.

References

[1] N. Amenta & G. M. Ziegler: *Deformed products and maximal shadows,* in: "Advances in Discrete and Computational Geometry" (B. Chazelle, J.E. Goodman, R. Pollack, eds.), *Contemporary Mathematics* (1998), Amer. Math. Soc., Providence, in print.

[2] D. Avis & V. Chvátal: *Notes on Bland's pivoting rule,* in: "Polyhedral Combinatorics" (M. L. Balinski, A. J. Hoffmann, eds.), *Math. Programming Study* **8** (1978), 24-34.

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

[4] K.H. Borgwardt: *The Simplex Method. A Probabilistic Analysis,* Algorithms and Combinatorics **1**, Springer 1987.

[5] M. Bousquet-Mélou: *The number and average length of decreasing paths in the Klee-Minty cube,* Manuscript, 1996.

[6] A. Z. Broder, M. E. Dyer, A. M. Frieze, P. Raghavan & E. Upfal: *The worst-case running time of the random simplex algorithm is exponential in the height, Information Processing Letters* **56** (1995), 79-81.

[7] H. Bruggesser & P. Mani: *Shellable decompositions of cells and spheres, Math. Scand.* **29** (1971), 197-205.

[8] G.B. Dantzig: *Linear Programming and Extensions,* Princeton University Press, Princeton 1963.

[9] B. Gärtner: *Combinatorial Linear Programming: Geometry Can Help,* in: Proc. "2nd. International Workshop on Randomization and Approximation Techniques in Computer Science "(RANDOM), Barcelona, 1998, Lecture Notes in Computer Science, to appear

[10] B. Gärtner: *Randomized Optimization by Simplex Type Methods,* Ph.D. Thesis, Freie Universität Berlin 1995.

[11] B. Gärtner & G. 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 Press, Los Alamitos 1994, pp. 502-510.

[12] D. Goldfarb: *Worst case complexity of the shadow vertex simplex algorithm,* preprint, Columbia University 1983, 11 pages; published in: D. Goldfarb: *On the complexity of the simplex algorithm,* in: *Advances in optimization and numerical analysis*, (S. Gomez, ed.), Kluwer, Dordrecht 1994, 25-38.

[13] M. Goldwasser: *A survey of linear programming in randomized subexponential time, ACM-SIGACT News* **26** (1995), 96-104.

[14] R.L. Graham, D.E. Knuth & O. Patashnik: *Concrete Mathematics,* Addison-Wesley, Reading, MA 1989.

[15] B. Grünbaum: *Convex Polytopes,* Interscience, London 1967.

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

[17] G. Kalai & D. J. Kleitman: *A quasi-polynomial bound for the diameter of graphs of polyhedra, Bulletin Amer. Math. Soc.* **26** (1992), 315–316.

[18] D.G. Kelly: *Some results on random linear programs,* Methods of Operations Research **40** (1981), 351-355.

[19] V. Klee: *Paths on polyhedra II,* Pacific Journal of Mathematics **17** (1966), 249-262.

[20] V. Klee & P. Kleinschmidt: *The d-step conjecture and its relatives, Math. Operations Research* **12** (1987), 718-755.

[21] V. Klee & P. Kleinschmidt: *Geometry of the Gass-Saaty parametric cost LP algorithm, Discrete Comput. Geometry* **5** (1990), 13-26.

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

[23] D. G. Larman: *Paths on polytopes, Proc. London Math. Soc.* **20** (1970), 161-178.

[24] J. Matoušek: *Lower bounds for a subexponential optimization algorithm,* Random Structures & Algorithms **5** (1994), 591-607.

[25] J. Matoušek, M. Sharir & E. Welzl: *A subexponential bound for linear programming, Algorithmica* **16** (1996), 498-516.

[26] P. McMullen: *The maximum numbers of faces of a convex polytope,* Mathematika **17** (1970), 179-184.

[27] N. Megiddo: *Linear programming in linear time when the dimension is fixed, J. ACM* **31** (1984), 114-127.

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

[29] C.H. Papadimitriou & K. Steiglitz: *Combinatorial Optimization: Algorithms and Complexity,* Prentice-Hall, Inc., Englewood Cliffs, NJ 1982.

[30] A. Schrijver: *Theory of Linear and Integer Programming,* Wiley-Interscience, Chichester 1986.

[31] R. Shamir: *The efficiency of the simplex method: a survey, Management Sci.* **33** (1987), 301-334.

[32] K. Williamson Hoke: *Completely unimodal numberings of a simple polytope, Discrete Applied Math.* **20** (1988), 69-81.

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

[34] G.M. Ziegler: *Lectures on Polytopes,* Graduate Texts in Mathematics **152**, Springer-Verlag, New York 1995.

Bernd Gärtner
Institut für Informatik
ETH Zürich
Haldeneggsteig 4
CH-8092 Zürich, Switzerland
gaertner@inf.ethz.ch

Martin Henk and Günter M. Ziegler
Fachbereich Mathematik, Sekr. 7-1
Technische Universität Berlin
Str. des 17. Juni 136
D-10623 Berlin, Germany
[henk,ziegler]@math.tu-berlin.de