Linear Programming and Unique Sink Orientations

Bernd Gärtner*
Institute of Theoretical Computer Science
ETH Zürich
8092 Zürich
Switzerland
gaertner@inf.ethz.ch

Ingo Schurr†
Institute of Theoretical Computer Science
ETH Zürich
8092 Zürich
Switzerland
schurr@inf.ethz.ch

Abstract

We show that any linear program (LP) in n nonnegative variables and m equality constraints defines in a natural way a *unique sink orientation* of the n-dimensional cube. From the sink of the cube, we can either read off an optimal solution to the LP, or we obtain certificates for infeasibility or unboundedness.

This reduction complements the implicit local neighborhoods induced by the vertex-edge structure of the feasible region with an explicit neighborhood structure that allows random access to all 2^n candidate solutions. Using the currently best sink-finding algorithm for general unique sink orientations, we obtain the fastest deterministic LP algorithm in the RAM model, for the central case $n = 2m$.

1 Introduction

Linear Programming is the problem of maximizing a linear function subject to linear (in)equality constraints. Here, we consider linear programs (LP) of the form

$$\text{(LP)} \quad \begin{array}{ll} \max & c^T x \\ \text{s.t.} & Ax = b \\ & x \geq 0, \end{array} \tag{1.1}$$

with $A \in \mathbb{R}^{m \times n}, c \in \mathbb{R}^n, b \in \mathbb{R}^m$. Any LP can be converted into this form. If the polyhedron $\mathrm{P}(A, b) = \{x \in \mathbb{R}^n \mid Ax = b, x \geq 0\}$ is nonempty, (1.1) is called *feasible*, otherwise *infeasible*. If in the feasible case, the objective function value $c^T x$ is bounded above over $\mathrm{P}(A, b)$, (1.1) is called *bounded*, otherwise *unbounded*. Any feasible and bounded linear program has a maximum value $z^* \in \mathbb{R}$, and any $x \in \mathrm{P}(A, b)$ with $c^T x = z^*$ is called an *optimal solution* [1].

To solve the linear program (1.1) means to compute an optimal solution x^* if the problem is feasible and bounded, and to report infeasibility or unboundedness otherwise.

In this paper, we introduce a new method for solving linear programs, based on the concept of *unique sink orientations of cubes* (USO) [17]. An orientation of the n-cube graph is said to be USO if all subgraphs induced by nonempty faces of the cube hae unique sinks. Like in the *simplex method* [3, 1], some rule guides us through a finite sequence of candidate solutions (associated with cube vertices), where a single iteration boils down to solving some systems of linear equations. In almost all other aspects, our method is quite different from the simplex method. Here are its distinguishing features.

- There is only one phase; the output is a pair of primal and dual optimal solutions to a modified linear program, and this pair is a certificate for either infeasibility, unboundedness, or optimality of the original problem (in the latter case, the modified linear program is the original one).
- The output does not depend on the internal rule being used and therefore defines a *canonical solution* to any LP, even in the infeasible or unbounded case.
- All inputs (A, b, c) are dealt with in the same manner. The method implicitly resolves rank deficiencies and other degeneracies.
- The internal rule does not necessarily induce a path in the cube graph; it can take full advantage of random vertex access.
- Under a suitable rule that actually "jumps around" in the cube, the new method is the fastest known deterministic algorithm for LP with $n = 2m$ variables, in the RAM model.

*supported by the Swiss Science Foundation (SNF), project No. 200021-100316/1
†supported by the joint Berlin/Zürich graduate program Combinatorics, Geometry and Computation (CGC), financed by the German Science Foundation (DFG) and ETH Zürich

This research is motivated by an open problem in complexity theory: although LP is known to be in **P** since 1979 [7], no *strongly* polynomial-time algorithm (a polynomial-time algorithm in the RAM model) is known. The quest for such an algorithm is a research challenge since more than thirty years [18]. The most prominent problem here is to find a pivot rule under which the simplex algorithm performs a polynomial number of steps in the worst case.

Despite substantial research, there is no pivot rule known that leads to substantially less iterations than the total number of vertices of $P(A, b)$ in the worst case, and this number may be exponential in m.[1] This is where unique sink orientations come in. USO were first introduced by Stickney and Watson [16] as digraph models for P-*matrix linear complementarity problems* (PLCP); recent years have seen a number of results concerned with the combinatorial and complexity theoretic aspects of USO, see the references in the next section.

The *Fibonacci Seesaw* of Szabó and Welzl [17] for finding the sink of a USO, implicitly given by a *vertex evaluation* oracle, features an interesting new twist: the possibility of random access to all candidate solutions may indeed help. For PLCP—not known to be in **P**—this "jumping" approach yields the fastest algorithm so far.

In this paper, we show that any linear program defines a USO in a natural way, with the unique global sink of the USO corresponding to the solution of the LP. Moreover, the vertex evaluation oracle required by the Fibonacci Seesaw (and other algorithms for general USO) boils down to Gaussian elimination. Plugging in the bounds of Szabó and Welzl for finding the sink in general USO [17], we obtain the following result, showing that jumping also helps for LP.

Theorem 1.1. *Any LP with $n = 2m$ variables can be solved in time*

$$O\left(1.606^{2m}\right) = O\left(2.58^{m}\right).$$

In contrast, there are linear programs with $n = 2m$ but a higher number of basic feasible solutions, almost all of which might be visited by a deterministic simplex algorithm. For example, any dual-to-cyclic polytope with n facets in $\mathbb{R}^{n-m}$ induces such an LP, after an affine transformation that writes it as the intersection of the positive orthant in $\mathbb{R}^n$ with an $(n-m)$-dimensional affine subspace. The number of vertices, equivalently the number of basic feasible solutions in the LP, is equal to the largest number allowed by the Upper Bound Theorem, and it is in particular larger than

$$\binom{m-1+\lfloor \frac{m}{2} \rfloor}{\lfloor \frac{m}{2} \rfloor} = 2^{\frac{3}{2}H\left(\frac{1}{3}\right)m+o(m)} = \Omega(2.598^m),$$

where $H(x)$ is the binary entropy function [19].

In fact, no deterministic algorithm for solving LP in the RAM model is known to require asymptotically less time than the number of basic feasible solutions, in the worst case. For $n = 2m$, the algorithm behind Theorem 1.1 beats this bound: it is decoupled from the concept of basic feasible solutions, and its runtime bound is indeed (exponentially) smaller than the worst-case number of basic feasible solutions.

The outline of the paper is as follows. In Section 2, we briefly review the concept of unique sink orientations. Section 3 presents our main technical tool for the LP-to-USO reduction, the Karush–Kuhn–Tucker conditions for convex programming. Based on this, section 4 shows that a certain class of very simple strictly convex programs give rise to USO.

Using this (essentially known) machinery, Section 5 starts our actual contribution. We symbolically perturb any LP to a family of strictly convex *quadratic* programs parameterized with $\varepsilon > 0$. As the key steps, we analyze the limiting USO obtained from letting ε tend to zero, and we show how we can perform vertex evaluations in this *LP-induced* USO without computing with ε's.

In Section 6, we examine the sink of the LP-induced USO, and we show how a certified solution to the LP can be extracted from it, even in the infeasible and unbounded case.

Section 7 addresses open questions; in particular, we point out that LP-induced USO possess the *Holt-Klee* property that set them apart from general USO, and we discuss possible algorithmic consequences of this result.

2 Unique sink orientations

Definition 2.1. *An orientation of the vertex-edge graph of the n-dimensional cube is called* unique sink orientation *(USO) if every subgraph induced by a nonempty cube face has a unique sink.*

In particular, the cube has a unique global sink. Starting from $n = 3$, there are USO containing directed cycles, like the one in Figure 1.

After their invention by Stickney and Watson in 1978 [16], USO have suffered a long period of non-observance. They were independently revived by two papers. On the negative side, Morris showed that for odd n, there are highly cyclic n-cube USO (actually coming from PLCP) on which the natural directed random walk takes more than $((n-1)/2)!$ steps to

[1] There are randomized pivot rules that lead to an expected *subexponential* number of steps[6, 9].

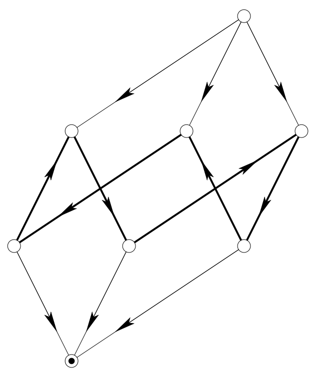

Figure 1: A cyclic USO of the 3-cube

reach the global sink, much more than the number of vertices [11]. On the positive side, Szabó and Welzl found a deterministic algorithm—the Fibonacci Seesaw—that finds the global sink of any n-cube USO by looking at less than 1.61^n vertices (more precisely, at the orientations of the incident edges) [17]. This is the *vertex evaluation* model.

Encouraged by these findings that established USO as fruitful objects with interesting combinatorial properties, a number of other results followed [8, 10, 4, 15, 13, 5, 14].

3 The Karush–Kuhn–Tucker conditions

Let us fix some notation first. For $x \in \mathbb{R}^n$ and $J \subseteq [n] := \{1, \ldots, n\}$, x_J is the $|J|$-dimensional vector obtained from x by collecting all coordinates with subscript in J. For $A \in \mathbb{R}^{m\times n}$, $A_J \in \mathbb{R}^{m\times|J|}$ collects the *columns* of A with subscript in J.

With $\mathbf{0}$ being the zero vector of the appropriate dimension, we also use

$$\mathbb{R}^J := \{x \in \mathbb{R}^n \mid x_{[n]\setminus J} = \mathbf{0}\}.$$

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a differentiable convex function with continuous partial derivatives. For $I \subseteq J \subseteq [n]$ we consider the *convex programming* problem

$$\text{(3.2)} \qquad \begin{array}{lll} \mathrm{CP}(I, J) & \min & f(x) \\ & \text{s.t.} & Ax = b \\ & & x \in \mathbb{R}^J \\ & & x_{J\setminus I} \geq 0. \end{array}$$

where $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$. The following is a specialization of a very general theorem to our scenario. For completeness, we include a simple proof using LP duality.

THEOREM 3.1. *$x^* \in \mathbb{R}^n$ is an optimal solution to problem* $\mathrm{CP}(I, J)$ *if and only if*

(i) $Ax^* = b,\ x^* \in \mathbb{R}^J,\ x^*_{J\setminus I} \geq 0$*, and*

(ii) there exists $\lambda \in \mathbb{R}^m$ *such that for all* $j \in J$*,*

$$\nabla f(x^*)_j - \lambda^T A_j \geq 0,$$

with equality if $j \in I$ *or* $x^*_j > 0$.

Here, $\nabla f(x^*)$ *is the gradient of* f *at* x^* *which by convention is an* n*-dimensional row vector.*

The entries of λ are called *Karush–Kuhn–Tucker* (KKT) *multipliers.*

Proof. Assume (i) and (ii) hold. Then we get

$$\begin{array}{rcl} (\nabla f(x^*) - \lambda^T A)x^* & = & 0, \text{ and} \\ (\nabla f(x^*) - \lambda^T A)x & \geq & 0 \end{array}$$

for all feasible solutions x to $\mathrm{CP}(I, J)$. Subtracting the first equation from the second, the contributions of $\lambda^T A$ cancel, and we get

$$\text{(3.3)} \qquad 0 \leq \nabla f(x^*)(x - x^*).$$

The fact that this equation holds for all feasible x is well-known to characterize optimality of x^*, see for example [12].

For the other direction, assume that x^* is optimal, in particular feasible, which gives (i). Since in this case, (3.3) holds for all feasible x, the vector x^* is an optimal solution to the linear program

$$\begin{array}{ll} \min & \nabla f(x^*)x \\ \text{s.t.} & Ax = b \\ & x \in \mathbb{R}^J \\ & x_{J\setminus I} \geq 0. \end{array}$$

Let y^* be an optimal solution to the *dual* linear program

$$\begin{array}{lll} \max & b^T y & \\ \text{s.t.} & y^T A_j = \nabla f(x^*)_j, & j \in I \\ & y^T A_j \leq \nabla f(x^*)_j, & j \in J \setminus I. \end{array}$$

By the complementary slackness condition, $(y^*)^T A_j = \nabla f(x^*)_j$ for all $j \in J \setminus I$ with $x^*_j > 0$, implying that $\lambda := y^*$ fulfills (ii).

4 Simple convex programming

Let us fix a *strictly* convex function f and consider for $I \subseteq J \subseteq [n]$ the problem

$$\text{(4.4)} \qquad \begin{array}{lll} \mathrm{SCP}(I, J) & \min & f(x) \\ & s.t. & x \in \mathbb{R}^J, \\ & & x_{J\setminus I} \geq 0. \end{array}$$

This is just the program CP(I,J) from the previous section, restricted to the strictly convex case, and with $A \in \mathbb{R}^{0\times n}$. We refer to this setup as *simple convex programming* (SCP).

Since f is strictly convex and SCP(I,J) is feasible, the program SCP(I,J) has a unique solution $x^*(I,J)$, for all pairs $I \subseteq J$. Applying Theorem 3.1,[2] we see that $x^* = x^*(I,J)$ if and only if we have *primal feasibility*

$$x^*_{[n]\setminus J} = \mathbf{0}, \tag{4.5}$$
$$x^*_{J\setminus I} \geq \mathbf{0}, \tag{4.6}$$

along with *dual feasibility*

$$\nabla f(x^*)_I = \mathbf{0}, \tag{4.7}$$
$$\nabla f(x^*)_{J\setminus I} \geq \mathbf{0}, \tag{4.8}$$

and *complementarity*

$$\nabla f(x^*)_j x^*_j = 0, \quad j \in J \setminus I. \tag{4.9}$$

Let us focus on the case $I = J$ for a moment. Conditions (4.6), (4.8) and (4.9) are vacuous, so we get that $x^* = x^*(J,J)$ if and only if

$$\begin{array}{rcl} x^*_{[n]\setminus J} &=& \mathbf{0}, \\ \nabla f(x^*)_J &=& \mathbf{0}. \end{array} \tag{4.10}$$

Towards the USO. Identifying the n-cube vertices with the sets $J \subseteq [n]$, we will derive the edge orientations from the vectors $x^*(J,J)$. We still need one preparatory

LEMMA 4.1. *For $J \subseteq [n]$, $j \in J$ and $I := J \setminus \{j\}$, the following two statements are equivalent.*

(i) $x^*(J,J)_j > 0$.

(ii) $\nabla f(x^*(I,I))_j < 0$.

Proof. If $x^*(J,J)_j > 0$, then $x^*(J,J)$ is feasible and therefore optimal for the more restricted problem SCP(I,J). On the other hand, $x^*(J,J)_j > 0$ shows that $x^*(J,J) \neq x^*(I,I)$. This means, we have $x^*(I,J) \neq x^*(I,I)$, the only possible reason being that (4.8) fails for $x^* = x^*(I,I)$. This shows $\nabla f(x^*(I,I))_j < 0$.

Conversely, $\nabla f(x^*(I,I))_j < 0$ implies $x^*(I,J) \neq x^*(I,I)$, so $x^*(I,J)_j > 0$. Complementarity yields $\nabla f(x^*(I,J))_j = 0$, so $x^*(I,J)$ is also optimal for the less restricted problem SCP(J,J) by (4.10). This yields $x^*(J,J)_j = x^*(I,J)_j > 0$.

Here is the main result of this section.

[2]you might want to recheck it for the case of a matrix A with no rows

THEOREM 4.1. *For $J \subseteq [n], j \in J$ and $I := J \setminus \{j\}$, the edge orientations*

$$I \to J :\Leftrightarrow x^*(J,J)_j > 0 \quad (\Leftrightarrow \nabla f(x^*(I,I))_j < 0)$$

define a USO of the n-cube.

Proof. We have to show that every nonempty cube face has a unique sink. In our interpretation of cube vertices as subsets $J \subseteq [n]$, the faces can be identified with *set intervals* of the form

$$[I,J] := \{F \subseteq [n] \mid I \subseteq F \subseteq J\}.$$

We claim that

$$S := I \cup \{j \in J \mid x^*(I,J)_j > 0\} \tag{4.11}$$

is the desired sink of the face $[I,J], I \subseteq J$. First observe that by this definition of S, $x^* = x^*(I,J)$ satisfies

$$x^*_{[n]\setminus S} = \mathbf{0}, \tag{4.12}$$
$$\nabla f(x^*)_S = \mathbf{0}, \tag{4.13}$$

by (4.7) and complementarity (4.9). It follows that $x^*(I,J) = x^*(S,S)$, by (4.10). Therefore,

$$x^*(S,S)_j > 0, \quad j \in S \setminus I, \tag{4.14}$$
$$\nabla f(x^*(S,S))_j \geq 0, \quad j \in J \setminus S, \tag{4.15}$$

by (4.8). According to the definition of the orientation, S is a sink in $[I,J]$.

Conversely, if S is any sink in $[I,J]$, then the two previous inequalities hold, so $x^*(S,S)$ is feasible for SCP(I,J) since (4.12) and (4.14) imply (4.5) and (4.6), and it is dual feasible since (4.13) and (4.15) imply (4.7) and (4.8). Complementarity (4.9) follows from (4.12) and (4.13). Thus, $x^*(S,S) = x^*(I,J)$, where (4.14) forces S to coincide with the set defined in (4.11).

We remark that this reduction of SCP to USO is known for the case where f is a *quadratic* function, and this is the situation in which we will apply it.

In the quadratic case, f is of the form

$$f(x) = x^T Q x + u^T x + w$$

for some symmetric positive definite matrix $Q \in \mathbb{R}^{n\times n}$, $u \in \mathbb{R}^n$, and $w \in \mathbb{R}$. For $I = \emptyset, J = [n]$, the optimality conditions (4.5) through (4.9) define a linear complementarity problem whose matrix is in particular a P-matrix. The USO derived by Stickney and Watson [16] then coincides with the orientation we have defined in Theorem 4.1 [2].

The above generalization to strictly convex functions is straightforward, and we have included the proof mainly for completeness. Our actual contribution is the nontrivial reduction of LP to strictly convex quadratic programming in the next two sections.

5 LP-induced USO

Given the linear program (1.1) with n variables, we define for any $\varepsilon > 0$ a quadratic function f_ε by

$$\begin{aligned} f_\varepsilon(x) &:= x^T(A^TA + \varepsilon^2 I)x - 2b^TAx - 2\varepsilon c^T x \\ &= \|Ax - b\|^2 - 2\varepsilon c^T x + \varepsilon^2\|x\|^2 - b^T b. \end{aligned}$$

Here, I is the identity matrix of the appropriate dimension (n in this case). Since $A^TA + \varepsilon^2 I$ is positive definite for all $\varepsilon > 0$, $f_\varepsilon(x)$ is a strictly convex function.

Let us denote by $\mathrm{SCP}_\varepsilon(I, J)$ the program (4.4) with function $f = f_\varepsilon$. We are interested in the behavior for $\varepsilon \to 0$. We expect that in the limit, the program lexicographically minimizes the triple

$$(\|Ax - b\|^2, -2c^T x, \|x\|^2).$$

In the feasible and bounded case, the solution $x^*_\varepsilon(\emptyset, [n])$ of $\mathrm{SCP}_\varepsilon(\emptyset, [n])$ should therefore converge to the optimal LP solution of minimum norm.

In order to understand the USO induced by f_ε, we have to know the values $x^*_\varepsilon(J, J)$. From (4.10) it follows that $x^*_\varepsilon(J, J) \in \mathbb{R}^J$ is obtained from the unique solution of the linear equation system

$$(5.16)\quad \frac{\nabla f(x)_J^T}{2} = (A_J^T A_J + \varepsilon^2 I)x_J - A_J^T b - \varepsilon c_J = \mathbf{0}$$

with $x \in \mathbb{R}^J$.

Lemma 5.1. *Let $\stackrel{\varepsilon}{\to}$ be the USO of the n-cube induced by f_ε according to Theorem 4.1. Then there exists a USO $\to$ such that $\stackrel{\varepsilon}{\to} = \to$ for sufficiently small ε.*

We call this "limiting" USO the USO *induced* by the LP.

Proof. Using Cramer's rule to compute the solution $x^*_\varepsilon(J, J)_J$ of the system (5.16), we see that the entries of all $x^*_\varepsilon(J, J)_j$ are rational functions in ε. By Theorem 4.1, $\stackrel{\varepsilon}{\to}$ is determined by the signs of finitely many of these rational functions.

Now, for any nonzero rational function $r(\varepsilon)$, there is an open interval of the form $(0, \delta)$ in which neither its numerator nor its denominator has any zeros. In this interval, the sign of $r(\varepsilon)$ is fixed. The lemma follows.

This also provides a way of computing edge orientations in the limiting USO $\to$: simply compute the rational function "responsible" for the orientation in question and find the terms with the smallest ε-power in both the numerator and the denominator. These determine the sign of the rational function for $\varepsilon \to 0$.

Nevertheless, since the limiting USO does not depend on ε, there must be a way of avoiding computations involving ε. Our approach is to develop $x^*_\varepsilon(J, J)$ into a power series, and this will also be crucial for understanding the global sink of $\to$ in the next section.

A zoo of easy programs. It will turn out that the coefficients of the power series expansion are solutions to unconstrained strictly convex quadratic programs (where unconstrained means that there are no inequalities). All "animals" in the following list are unique optimal solutions of their defining programs.

Definition 5.1. *For $J \subseteq [n]$, set*

$$\begin{aligned} \bar{b}(J) &= \operatorname{argmin} && (b-y)^T(b-y) \\ & \text{s.t.} && Ax = y \\ &&& x \in \mathbb{R}^J, \end{aligned}$$

$$\begin{aligned} x(J) &= \operatorname{argmin} && x^T x \\ & \text{s.t.} && Ax = \bar{b}(J) \\ &&& x \in \mathbb{R}^J, \end{aligned}$$

$$\begin{aligned} \bar{c}(J) &= \operatorname{argmin} && (c-x)^T(c-x) \\ & \text{s.t.} && A_J^T y = x_J, \end{aligned}$$

$$\begin{aligned} y(J) &= \operatorname{argmin} && y^T y \\ & \text{s.t.} && A_J^T y = \bar{c}(J)_J, \end{aligned}$$

$$\begin{aligned} c(J) &= \operatorname{argmin} && (c-x)^T(c-x) \\ & \text{s.t.} && Ax = \mathbf{0}, \\ &&& x \in \mathbb{R}^J, \end{aligned}$$

$$\begin{aligned} b(J) &= \operatorname{argmin} && (b-y)^T(b-y) \\ & \text{s.t.} && A_J^T y = \mathbf{0}, \end{aligned}$$

$$\begin{aligned} t(J) &= \operatorname{argmin} && x^T x \\ & \text{s.t.} && Ax = y(J) \\ &&& x \in \mathbb{R}^J. \end{aligned}$$

If not indicated otherwise, all x range over $\mathbb{R}^n$ and all y over $\mathbb{R}^m$.

Note that all programs except the last one are easily seen to have feasible and therefore unique optimal solutions. For the last one, this will be shown in Lemma 5.2 below.

To get an intuition what these values are, let us consider for $\gamma \in \mathbb{R}^n$ and $\beta \in \mathbb{R}^m$ the (unconstrained, and therefore quite boring) linear program

$$(5.17)\quad \begin{aligned} \mathrm{LP}(J) \quad & \max && \gamma^T x \\ & \text{s.t.} && Ax = \beta, \\ &&& x \in \mathbb{R}^J, \end{aligned}$$

along with its dual

$$(5.18)\quad \begin{aligned} \mathrm{LP}^\triangle(J) \quad & \min && \beta^T y \\ & \text{s.t.} && A_J^T y = \gamma_J. \end{aligned}$$

The vector $\beta = \bar{b}(J)$ is the vector closest to b such that $\mathrm{LP}(J)$ is feasible, and $x(J)$ is the feasible solution with minimum norm. Dually, $\gamma = \bar{c}(J)$ is the vector closest to c such that $\mathrm{LP}^\triangle(J)$ is feasible, and $y(J)$ is the feasible solution with minimum norm. $c(J)$ is the projection of c onto the kernel of A_J, while $b(J)$ is the projection of b onto the kernel of A_J^T.

The following technical lemma also explains how $t(J)$ enters the picture. (Upon first reading, this lemma can be skipped.)

LEMMA 5.2. *For all $J \subseteq [n]$, the following holds.*

(i) $2t(J)$ *is the shortest vector of KKT multipliers for the program defining* $y(J)$.

(ii) *If* $b(J) \neq \mathbf{0}$, *then* $b^T b(J) > 0$, *and if* $c(J) \neq \mathbf{0}$, *then* $c^T c(J) > 0$.

(iii) $b = b(J) + \bar{b}(J)$ *and* $c = c(J) + \bar{c}(J)$.

(iv) $x(J)$ *has the following alternative definition.*

$$\begin{array}{rclrcl} x(J) & = & \text{argmin} & & x^T x & \\ & & \text{s.t.} & A_J^T A x & = & A_J^T b \\ & & & x & \in & \mathbb{R}^J. \end{array}$$

(v) $t(J)$ *has the following alternative definition.*

$$\begin{array}{rclrcl} t(J) & = & \text{argmin} & & x^T x & \\ & & \text{s.t.} & A_J^T A x & = & \bar{c}(J)_J \\ & & & x & \in & \mathbb{R}^J. \end{array}$$

Proof. (i) Theorem 3.1 states that $2y(J)^T = \lambda^T A_J^T$, for some vector $\lambda \in \mathbb{R}^{|J|}$, and $\lambda = 2t(J)_J$ is the shortest such vector by definition of $t(J)$.

(ii) If $b(J) \neq \mathbf{0}$, optimality of $b(J)$ under a strictly convex function yields

$$(b - b(J))^T (b - b(J)) < (b - \mathbf{0})^T (b - \mathbf{0}) = b^T b.$$

The inequality $2b^T b(J) > b(J)^T b(J) \geq 0$ follows. The argument for $c(J)$ is the same.

(iii) We only give the argument for b here, the one for c is similar. The Karush–Kuhn–Tucker conditions for the program defining $\bar{b}(J)$ restricted to (x_J, y) (Theorem 3.1) show that

$$\begin{array}{rcl} A_J x^* & = & \bar{b}(J), \\ (\mathbf{0}^T, 2(\bar{b}(J) - b)) & = & \lambda^T (A_J, -I), \end{array}$$

for some $x^* \in \mathbb{R}^J, \lambda \in \mathbb{R}^m$. Set $y^* = \lambda/2$. It follows that $y^* = b - \bar{b}(J)$ and $A_J^T y^* = \mathbf{0}$. Moreover, with $\mu := -2x^*$, we have $2(y^* - b)^T = -2\bar{b}(J) = \mu^T A_J^T$. This means that y^* and μ satisfy the Karush–Kuhn–Tucker conditions for the program defining $b(J)$. Then $b(J) = y^* = b - \bar{b}(J)$ follows.

Geometrically, $\bar{b}(J)$ is the projection of b onto the column space of A_J, while $b(J)$ is the projection of b onto the orthogonal complement of the column space.

(iv) In proving $A_J^T y^* = \mathbf{0}$ in (iii), we have shown that $A_J^T \bar{b}(J) = A_J^T b$. Since any feasible linear system $Mx = q$ is equivalent to $M^T M x = M^T q$,[3] we know that the system $A_J x_J = \bar{b}(J)$ that yields $x(J)_J$ can be replaced by $A_J^T A_J x_J = A_J^T \bar{b}(J) = A_J^T b$.

(v) Knowing from (i) that $A_J x_J = y(J)$ is feasible, we may replace it by $A_J^T A_J x_J = A_J^T y(J) = \bar{c}(J)_J$.

Here is the promised power series expansion.

THEOREM 5.1. *For $J \subseteq [n]$, define*

$$\begin{array}{rcl} g_{-1}(J) & = & c(J), \\ g_0(J) & = & x(J), \\ g_1(J) & = & t(J), \end{array}$$

and for $i \geq 2$,

$$\begin{array}{rclrcl} g_i(J) & = & \text{argmin} & & x^T x & \\ & & \text{s.t.} & A_J^T A x & = & -g_{i-2}(J)_J \\ & & & x & \in & \mathbb{R}^J. \end{array}$$

Then for any $k \geq 1$,

$$x_\varepsilon^*(J, J) = \sum_{i=-1}^{k} \varepsilon^i g_i(J) + O(\varepsilon^{k+1}),$$

where the big-O notation refers to the asymptotic behavior for $\varepsilon \to 0$.

With Lemma 5.2 (iv) and (v), any $g_i(J), i \geq 0$, is by Theorem 3.1 of the form $2g_i(J)_J^T = \lambda^T A_J^T A_J$ for some λ. This guarantees that the programs defining the $g_i(J), i \geq 2$ are feasible, so all the $g_i(J)$ are indeed well-defined.

Proof. Let us write $x_\varepsilon^*(J, J)$ in the form

$$x_\varepsilon^*(J, J) = \sum_{i=-1}^{k} \varepsilon^i g_i(J) - r_\varepsilon(J), \tag{5.19}$$

with $r_\varepsilon(J) \in \mathbb{R}^J$ the remainder term. The fact that $x_\varepsilon^*(J, J)_J$ solves (5.16) implies that $r_\varepsilon(J)_J$ must be the unique solution to the system

$$(A_J^T A_J + \varepsilon^2 I) x_J = \varepsilon^{k+1} g_{k-1}(J)_J + \varepsilon^{k+2} g_k(J)_J.$$

To see this, plug (5.19) into (5.16) and use $A_J c(J)_J = \mathbf{0}$, $A_J^T A_J x(J)_J = A_J^T b$ (Lemma 5.2(iv)), $A_J^T A_J t(J) = \bar{c}(J)_J$ (Lemma 5.2(v)), as well as $c = c(J) + \bar{c}(J)$

[3] We need to show that $M^T M x = M^T q$ implies $Mx = q$. Take any x' such that $Mx' = q$. Then we get $M^T M x' = M^T q$ and $M^T M(x' - x) = \mathbf{0}$. Also, $(x' - x)^T M^T M (x' - x) = \|M(x' - x)\|^2 = 0$ and $M(x' - x) = \mathbf{0}$ follows; hence $Mx = Mx' = q$.

(Lemma 5.2(iii)). Then watch the terms cancel. In other words, $r_\varepsilon(J)_J$ is of the form

$$\varepsilon^{k+1}\left((A_J^TA_J+\varepsilon^2 I)^{-1}\left(g_{k-1}(J)_J+\varepsilon g_k(J)_J\right)\right).$$

If we can show that for all $i \geq 0$,

$$s_\varepsilon := (A_J^TA_J+\varepsilon^2 I)^{-1} g_i(J)_J$$

converges as $\varepsilon \to 0$, we have shown $r_\varepsilon(J) = O(\varepsilon^{k+1})$. . By the remark preceding this proof, we can write this system as

$$(Q+\varepsilon^2 I)s_\varepsilon = Q\lambda$$

for some vector λ and symmetric matrix Q. Choose a diagonalizing transformation P such that $Q = P^{-1}DP$, where D is a diagonal matrix with diagonal entries $a_1,\ldots,a_\ell,0,\ldots,0$, the first ℓ of them nonzero. Then the matrix equation can be rewritten as

$$P^{-1}(D+\varepsilon^2 I)Ps_\varepsilon = P^{-1}DP\lambda,$$

which in turn is equivalent to $(D+\varepsilon^2 I)s'_\varepsilon = D\lambda'$, with $s'_\varepsilon = Ps_\varepsilon, \lambda' = P\lambda$. We then get

$$s'_\varepsilon = \left(\frac{a_1\lambda'_1}{a_1+\varepsilon^2},\ldots,\frac{a_\ell\lambda'_\ell}{a_\ell+\varepsilon^2},0,\ldots,0\right),$$

meaning that s'_ε and therefore $s_\varepsilon = P^{-1}s'_\varepsilon$ converges. (It can be shown that the limiting value is the minimum-norm solution of $Qx = Q\lambda$.)

This theorem shows that we can read off the edge orientation $J\backslash\{j\} \to J$ in the LP-induced USO from the first nonzero coefficient in the power series expansion of $x^*_\varepsilon(J,J)$. Here are the details.

Corollary 5.1. *Let $J \subseteq [n], j \in J$, and set $I := J \setminus \{j\}$. Furthermore, define*

$$i(J,j) := \min\{i \geq -1 \mid g_i(J)_j \neq 0\}.$$

Then $i(J,j) = \infty$ or $i(J,j) \leq 2|J|-1$, and the LP-induced USO $\to$ derived in Lemma 5.1 induces the edge orientation

$$I \to J \quad \Leftrightarrow \quad g_{i(J,j)}(J)_j > 0, \tag{5.20}$$

where we set $g_\infty(J)_j := 0$.

Since our power series expansion also induces an expansion of $\nabla f(x^*_\varepsilon(J,J))$ (responsible for the "upward" edges at J for small ε), we can compute the orientations of *all* edges incident to a given vertex J in the LP-induced USO by solving at most $2|J|+2$ unconstrained quadratic programs, and hopefully much less in most cases; this is the vertex evaluation oracle. By the Karush–Kuhn–Tucker conditions, this is easy and reduces to solving linear equation systems.

Proof. We have

$$I \stackrel{\varepsilon}{\to} J \Leftrightarrow x^*_\varepsilon(J,J)_j > 0,$$

see the definition of the orientation in Theorem 4.1. Then, according to the previous theorem, the first nonzero value $g_i(J)_j$ determines the sign of $x^*_\varepsilon(J,J)_j$ for sufficiently small ε, and this is the sign that defines the orientation $I \to J$ in the limiting USO.

For the bound on $i(J,j)$ in the finite case, recall that $x^*_\varepsilon(J,J)_j$ is a rational function, and if it is nonzero, the numerator contains a monomial ε^i with $i \leq 2|J|-1$ (this is again Cramer's rule, applied to the system (5.16). It follows that

$$|x^*_\varepsilon(J,J)_j| = \Omega\left(\varepsilon^{2|J|-1}\right)$$

for $\varepsilon \to 0$, and the previous theorem implies that $i(J,j) \leq 2|J|-1$.

On the other hand, $i(J,j) = \infty$ implies $x^*_\varepsilon(J,J)_j = 0$, so (5.20) gives the right orientation also in this case.

6 LP-induced USO: The sink

Let $S \subseteq [n]$ be the sink of the LP-induced USO $\to$. From Theorem 5.1 we know that

$$x^*_\varepsilon(S,S) = \frac{c(S)}{\varepsilon} + x(S) + \varepsilon t(S) + O(\varepsilon^2), \tag{6.21}$$

which implies

$$\begin{aligned} \nabla f(x^*_\varepsilon(S,S))^T/2 &= A^T(\bar b(S) - b) \\ &+ \varepsilon(A^Ty(S) - \bar c(S)) \\ &+ O(\varepsilon^2), \end{aligned} \tag{6.22}$$

using (5.16), $Ac(S) = \mathbf{0}, Ax(S) = \bar b(S)$ and $At(S) = y(S)$, see Definition 5.1, and $c = c(S)+\bar c(S)$, see Lemma 5.2(iii).

For sufficiently small ε, S is also the sink in $\stackrel{\varepsilon}{\to}$, so $x^*_\varepsilon(S,S) = x^*_\varepsilon(\emptyset,[n])$. Using the optimality criteria (4.6), (4.8) and (4.9), we deduce

$$x^*_\varepsilon(S,S) \geq \mathbf{0}, \tag{6.23}$$

$$\nabla f(x^*_\varepsilon(S,S)) \geq \mathbf{0}, \tag{6.24}$$

$$\nabla f(x^*_\varepsilon(S,S))_j x^*_\varepsilon(S,S)_j = 0, \quad j \in [n]. \tag{6.25}$$

This implies the following

Theorem 6.1. *Consider the linear program*

$$\begin{aligned} \max \quad & \bar c(S)^T x \\ \text{s.t.} \quad & Ax = \bar b(S) \\ & x \geq 0, \end{aligned} \tag{6.26}$$

along with its dual

$$\begin{aligned} \min \quad & \bar b(S)^T y \\ \text{s.t.} \quad & A^T y \geq \bar c(S). \end{aligned} \tag{6.27}$$

For sufficiently small $\varepsilon > 0$, the following statements hold.

(i) $x(S) + c(S)/\varepsilon$ is optimal for (6.26).

(ii) $y(S) - b(S)/\varepsilon$ is optimal for (6.27).

Proof. Putting together (6.21) and (6.23) shows that $x(S)+c(S)/\varepsilon \geq \mathbf{0}$, and feasibility for (6.26) follows from the definitions of $c(S), x(S)$. Similarly, we can combine (6.22) and (6.24) to deduce that $y(S)-b(S)/\varepsilon$ is feasible for (6.27); for this recall $b(S) = b - \bar{b}(S)$ by Lemma 5.2(iii).

To prove optimality, we argue as follows. From (6.25), we see that if $x(S)_j + c(S)_j/\varepsilon > 0$, then $A_j^T(\bar{b}(S) - b) = -A_j^T b(S) = 0$ and $A_j^T y(S) - \bar{c}(S)_j = 0$, since otherwise, the lower order terms of (6.21) and (6.22) cannot contribute enough to reach complementarity in coordinate j for small ε.

The latter observation implies the complementary slackness condition for the pair of feasible solutions in (i) and (ii), and this shows that both are optimal in their respective programs.

We have shown that the sink S of the LP-induced USO gives us a primal-dual pair of optimal solutions to a modified LP (6.26). Here is what we can deduce about the *original* LP, our primary object of interest.

In the following, an *unbounded ray* for an LP is a halfline whose tail (everything except some initial segment) is feasible, and on which the objective function is unbounded.

THEOREM 6.2. *Let S be the sink of the USO induced by the LP*

$$(6.28) \qquad \text{(LP)} \quad \begin{array}{ll} \max & c^T x \\ \text{s.t.} & Ax = b \\ & x \geq 0. \end{array}$$

(i) If $\bar{b}(S) \neq b$, the LP (6.28) is infeasible. Equivalently, the LP

$$\begin{array}{ll} \max & \bar{c}(S)^T x \\ \text{s.t.} & Ax = b \\ & x \geq 0 \end{array}$$

is infeasible, and this is witnessed by the fact that

$$\{y(S) - \frac{b(S)}{\varepsilon} \mid \varepsilon > 0\}$$

is an unbounded ray of the dual problem

$$\begin{array}{ll} \min & b^T y \\ \text{s.t.} & A^T y \geq \bar{c}(S). \end{array}$$

(ii) If $\bar{b}(S) = b$ and $\bar{c}(S) \neq c$, the LP (6.28) is feasible but unbounded, and this is witnessed by the fact that

$$\{x(S) + \frac{c(S)}{2\varepsilon} \mid \varepsilon > 0\}$$

is an unbounded ray of (6.28).

(iii) If $\bar{b}(S) = b$ and $\bar{c}(S) = c$, then $x(S)$ and $y(S)$ is a pair of primal and dual optimal solutions to the LP (6.28).

Proof. By weak duality, the existence of a dual unbounded ray implies infeasibility of the primal problem, so in order to show (i) and (ii), it remains to prove that the given rays are indeed unbounded. But this follows from $c^T c(S) > 0$ and $b^T b(S) > 0$, see Lemma 5.2(ii). Property (iii) is a corollary of the previous theorem, under $b(S) = b - \bar{b}(S) = \mathbf{0}$ and $c(S) = c - \bar{c}(S) = \mathbf{0}$.

We remark that the linear programs (6.26) and (6.27) can also be defined without reference to the sink S (this follows from the KKT conditions, taking into account that $A^T(\bar{b}(S) - b) \geq \mathbf{0}$ and $c(S) \geq \mathbf{0}$).

LEMMA 6.1. *With S the sink of the LP-induced USO, we have*

$$\begin{array}{rclrcl} \bar{b}(S) & = & \text{argmin} & (b-y)^T(b-y) & & \\ & & \text{s.t.} & Ax & = & y \\ & & & x & \geq & \mathbf{0} \end{array}$$

and

$$\begin{array}{rclrcl} \bar{c}(S) & = & \text{argmin} & (c-x)^T(c-x) & & \\ & & \text{s.t.} & A^T y & \geq & x. \end{array}$$

This means, $\bar{b}(S)$ is the closest replacement for b which makes the original LP feasible, while $\bar{c}(S)$ is the closet replacement for c which makes the dual problem feasible. In this sense, the USO approach solves the feasible and bounded LP "closest" to the original one.

7 Discussion

The primal-dual pair of feasible solutions in Theorem 6.1 only depends on constantly many terms in the power series expansion of $x_\varepsilon^*(S, S)$. Although the lower order terms are "garbage" in this sense, they may still be needed in order to resolve ties when the "interesting" terms vanish. A natural question is whether fewer terms (maybe only $c(S)$ and $x(S)$?) suffice to define a USO, where ties are broken by some other rule. We currently see no systematic way of proving this, and we believe that the necessity of having to go through linearly many terms is the "price of degeneracy". In fact, tie-braking using a power series is nothing new: the *lexicographic method* for degeneracy resolution in the simplex algorithm does just that, and also in the lexicographic method, one may end up going through many lower order terms.

Breaking ties randomly is possible, but we do not consider this a pleasing solution from a combinatorial point of view: perturbing the objective function vector c to $c' = c + A^T\tilde{y}$, with $\tilde{y}$ chosen randomly from a suitable distribution, we can guarantee that ties are resolved early with high probability. Details will be given in the full paper.

All USO coming from PLCP according to Stickney and Watson [16] have recently been shown to satisfy an interesting combinatorial property, the *Holt-Klee condition* [5]. This means that any k-dimensional face has k vertex-disjoint paths from its unique source to its unique sink.[4]

Since LP-induced USO are in particular PLCP-induced USO (see the discussion at the end of Section 4), the Holt-Klee condition holds, and we are in a (probably) very small subclass of all USO. The obvious question is whether this can be exploited algorithmically, leading to improvements in Theorem 1.1. Unfortunately, we are not aware of any sink-finding algorithm that makes use of the Holt-Klee condition. In particular, we have no idea whether the Fibonacci Seesaw employed in Theorem 1.1 can be improved under this condition.

Acknowledgment

We thank Emo Welzl for pointing us to the case $n = 2m$ underlying Theorem 1.1. We also thank an anonymous SODA referee for many corrections and helpful suggestions.

References

[1] V. Chvátal. *Linear Programming.* W.H.Freeman, New York, 1983.

[2] R. W. Cottle, J. Pang, and R. E. Stone. *The Linear Complementarity Problem.* Academic Press, 1992.

[3] G. Dantzig. Linear programming and extensions, 1963.

[4] M. Develin. LP-orientations of cubes and crosspolytopes. *Advances in Geometry*, 4:459–468, 2004.

[5] B. Gärtner, W. D. Morris, and L. Rüst. Unique sink orientations of grids. In *Proc. 11th Conference on Integer Programming and Combinatorial Optimization (IPCO)*, volume 3509 of *Lecture Notes in Computer Science*, pages 210–224, 2005.

[6] G. Kalai. A subexponential randomized simplex algorithm. In *Proc. 24th annu. ACM Symp. on Theory of Computing.*, pages 475–482, 1992.

[7] L. G. Khachiyan. Polynomial algorithms in linear programming. *U.S.S.R. Comput. Math. and Math. Phys*, 20:53–72, 1980.

[8] J. Matoušek. The number of unique sink orientations of the hypercube. Combinatorica, to appear.

[9] J. Matoušek, M. Sharir, and E. Welzl. A subexponential bound for linear programming. *Algorithmica*, 16:498–516, 1996.

[10] W. D. Morris. Distinguishing cube orientations arising from linear programs. Manuscript, 2002.

[11] W. D. Morris. Randomized principal pivot algorithms for P-matrix linear complementarity problems. *Math. Programming, Ser. A*, 92:285–296, 2002.

[12] A. L. Peressini, F. E. Sullivan, and J. J. Uhl. *The Mathematics of Nonlinear Programming.* Undergraduate Texts in Mathematics. Springer-Verlag, 1988.

[13] I. Schurr and T. Szabó. Finding the sink takes some time. *Discrete Comput Geom*, 31:627–642, 2004.

[14] I. Schurr and T. Szabó. Jumping doesn't help in abstract cubes. In *Proc. 11th Conference on Integer Programming and Combinatorial Optimization (IPCO)*, volume 3509 of *Lecture Notes in Computer Science*, pages 225–235, 2005.

[15] J. Matoušek and T. Szabó. RANDOM EDGE can be exponential on abstract cubes. In *Proc. 45nd Ann. IEEE Symp. on Foundations of Computer Science (FOCS)*, pages 92–100, 2004.

[16] A. Stickney and L. Watson. Digraph models of bard-type algorithms for the linear complementary problem. *Mathematics of Operations Research*, 3:322–333, 1978.

[17] T. Szabó and E. Welzl. Unique sink orientations of cubes. In *Proc. 42^{nd} IEEE Symp. on Foundations of Comput. Sci.*, pages 547–555, 2000.

[18] M. Todd. The many facets of linear programming. *Mathematical Programming*, 91:417–436, 2002.

[19] G. M. Ziegler. *Lectures on Polytopes.* Number 152 in Graduate texts in mathematics. Springer, 1995.

[4] The existence of unique sources follows from the USO axioms.