Unique Sink Orientations of Grids

Bernd Gärtner · Walter D. Morris Jr. · Leo Rüst

Abstract We introduce unique sink orientations of grids as digraph models for many well-studied problems, including linear programming over products of simplices, generalized linear complementarity problems over P-matrices (PGLCP), and simple stochastic games.

We investigate the combinatorial structure of such orientations and develop randomized algorithms for finding the sink. We show that the orientations arising from PGLCP satisfy the Holt-Klee condition known to hold for polytope digraphs, and we give the first expected linear-time algorithms for solving PGLCP with a fixed number of blocks.

1 Introduction

A *grid* is a graph whose vertex set is the Cartesian product of n finite sets, with edges joining all pairs of vertices that differ in exactly one component, see Fig. 1a.

B. Gärtner · L. Rüst (✉)
Institute of Theoretical Computer Science, ETH Zürich, 8092 Zürich, Switzerland
e-mail: ruestle@inf.ethz.ch

B. Gärtner
e-mail: gaertner@inf.ethz.ch

W.D. Morris Jr.
Department of Mathematical Sciences, George Mason University, MS 3F2, Fairfax,
VA 22030-4444, USA
e-mail: wmorris@gmu.edu

Alternatively, we can view a grid as the skeleton (vertex-edge graph) of a specific polytope, namely a product of simplices.

If all sets have size two, we get the graph of the *n*-cube. A *face* or *subgrid* is any induced subgraph spanned by the Cartesian product of subsets of the original sets. An orientation ψ of the grid is called a *unique sink orientation* (USO) if every face has a unique sink with respect to ψ. Figure 1b depicts a USO of the $(3 \times 2 \times 2)$-grid. In particular, the grid itself must have a unique sink. Grid USO may contain directed cycles, as the 3-cube in Fig. 2 shows.

The significance of USO on grids comes from the fact that they form a simple combinatorial framework subsuming a number of well-studied problems. We show in this paper that the problem of solving a *generalized linear complementarity problem* over a P-matrix (PGLCP), as introduced by Cottle & Dantzig [9], can be recast as the problem of finding the unique sink of an implicitly given grid USO. As special cases, this includes the well-known standard linear complementarity problems over P-matrices (PLCP) [10] and problems reducing to them [18, 39], linear programming (LP) over products of simplices, and LP over combinatorial cubes. In the LP applications, we get acyclic unique sink orientations (AUSO).

Two major open problems motivate our research. On the one hand, it is unknown whether polynomial-time algorithms exist for PLCP or PGLCP, even though both problems are unlikely to be NP-hard. Megiddo has shown that hardness of PLCP would imply NP = co-NP [37], and his proof extends to PGLCP easily. LP, on the other hand, is solvable in polynomial time (a celebrated result of Khachyian [31]), but a *strongly* polynomial algorithm is not known, even if we are dealing only with LP over combinatorial cubes. Candidates for strongly polynomial algorithms must be *combinatorial* in the sense that the number of arithmetic operations they perform depends only on the combinatorial structure of the LP but not on the actual numbers that encode it. The AUSO approach attempts to extract the combinatorial structure behind
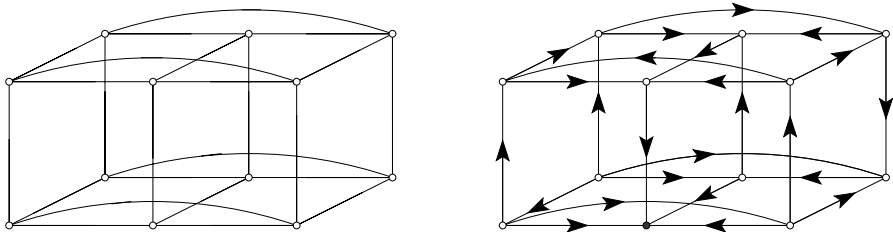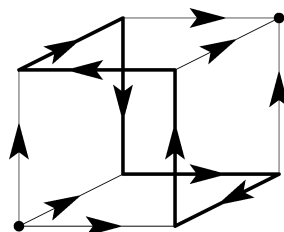

Fig. 1 **a** The $(3 \times 2 \times 2)$-grid, **b** a unique sink orientation

Fig. 2 A cyclic USO

LP; in this paper, we generalize to the combinatorics of PGLCP. A polynomial-time algorithm for finding the sink of a grid USO (using an oracle that returns the orientation of a given edge) would solve both problems in strongly polynomial time. The USO generalization reveals some (algorithmically useful) hidden structure, leading to new results for PGLCP.

The AUSO framework also covers a generalization of LP resulting from the replacement of linear objective functions with *abstract objective functions* (AOF) [1, 29], or *completely unimodal numberings* of vertices [26, 48]. On general polytopes, these concepts are dual to the notion of *shellings* [26], and they have successfully been applied to the theory of polytope (di)graphs and linear programming [28, 30]. A further generalization leads to the natural concept of *violator spaces*, and grid USO can be shown to form nontrivial models of (possibly cyclic) violator spaces [23].

The case of grid AUSO—equivalently, AOF on products of simplices—has been treated in the planar case ($n = 2$), with interesting connections to *arrangements of pseudolines* [15, 46].

For general n, grid AUSO have been studied by Björklund et al. as a combinatorial framework for the problem of computing optimal infinite game strategies [2, 3]. The games considered include parity, mean-payoff, and *simple stochastic games* (SSG). Whether parity games (the easiest among the three) can be solved in polynomial time is an important open question [25]. In particular, the complexity of very natural *strategy improvement* algorithms for these classes of games is still open. For SSG, such an algorithm is the *Hoffman-Karp* algorithm [8].

By the reductions of Björklund et al. [2, 3] (and the earlier reduction of Ludwig for binary SSG [34]), we can interpret the Hoffman-Karp algorithm as a sink-finding method for grid AUSO. In the USO world, this method is known as the *bottom-antipodal* algorithm, and there are exponential lower bounds for its performance [40]. In order to understand whether or not these bounds can be turned into lower bounds for the Hoffman-Karp algorithm, it is important to understand the class of grid AUSO that may arise from SSG.

An alternative reduction from SSG to grid USO follows from our results, together with a recent reduction of SSG to PGLCP [18]. In this way, we may get cyclic USO, but in return, they will satisfy the *Holt-Klee* condition, a combinatorial property shared by a diminishing fraction of all USO.

Unique sink orientations of cubes (not necessarily acyclic) were first considered by Stickney and Watson as digraph models for PLCP [43]. Most remarkably, Szabó and Welzl gave algorithms for finding the sink of an n-cube USO by looking at only $O(c^n)$ vertices and edges, for some c strictly smaller than 2 [45]. This in particular yields the first combinatorial algorithms for PLCP with nontrivial runtime bounds. Cube USO are also useful as combinatorial models for geometric optimization problems, in particular the problem of computing the smallest enclosing ball of a set of points [19, 45], or a set of balls [16]. In the point case, this problem can be reduced to PLCP with a positive definite matrix [19], but the ball case leads to more general USO. In both cases, cyclic orientations may arise.

Unique sink orientations of the graphs of general polytopes are dual to *exact signings* studied by Kleinschmidt and Onn [33].

Fig. 3 Two-dimensional projected USO over one-dimensional hypervertices (*shaded*) in a three-dimensional grid. The projected USO is defined through the arcs incident to the hypervertex sinks (*white*); in this case, we obtain the projected USO of Fig. 4
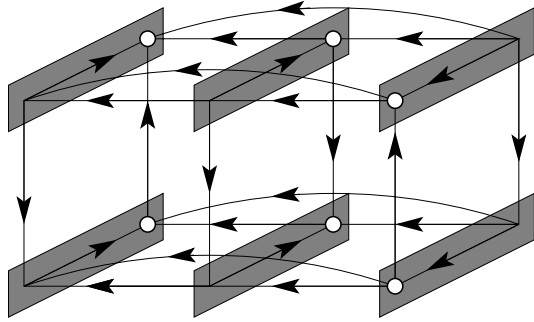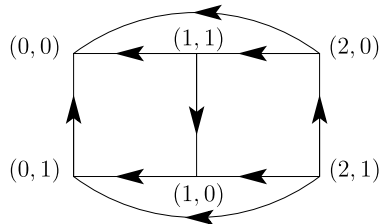
Fig. 4 The refined index. For each vertex, the two components correspond to the outgoing edges in the horizontal and vertical directions

1.1 Statement of Results

In this paper, we generalize results known to hold in some of the above special cases, and we prove new structural and algorithmic results of particular significance for the theory of PLCP and PGLCP. Probably the most surprising insight is that all these results hold even under the presence of directed cycles in the orientation.

1.1.1 Projected USO and Refined Index

A *hypervertex* is a face consisting of all vertices whose coordinates agree with a given vertex on some fixed subset I of the n grid dimensions. An $|I|$-dimensional *projected grid* can be imposed on the set of hypervertices defined by I in a natural way. Given a USO of the grid, the *projected orientation* of the projected grid is obtained from the orientations of the outgoing edges incident to the *sinks* of the hypervertices. We prove that the projected orientation is USO again. Figure 3 shows an example.

While this result is an easy generalization of a corresponding result for cubes [45], its implications are more interesting. In Sect. 2.2 we formally define the *refined index* and, based on the projected structure, we are able to prove that the refined index is a bijection for any grid USO. With respect to a given USO, the refined index maps every vertex to an n-tuple of natural numbers, the entry at position i corresponding to the number of outgoing edges in dimension i of the grid. Figure 4 depicts a USO along with all refined index values.

This bijection was previously known to hold for cubical grids (in which case it easily follows from the USO property [45]), and for acyclic grids [2], with a proof that does not extend to USO with cycles. An earlier proof for the planar case is due to Tschirschnitz [46]. For USO coming from LP over a product of simplices, or from

a PGLCP, the bijection can be established using simple geometric arguments. In the planar setting, it is the basis for a tight analysis of the RANDOMEDGE simplex algorithm [22].

Our inductive proof of the bijection for grid USO differs fundamentally from the proofs of the special cases. The reason is that the projected USO—which we use in our proof—will in general not come from PGLCP, even if the original grid USO does (we give a concrete example for this). Here, the generalization to USO allows for new approaches that are not necessarily viable in the concrete setting.

1.1.2 Algorithms

Given an oracle that returns the orientation of any given edge in a grid USO, it is natural to ask how many oracle calls are required in order to find the global sink of the USO. In all concrete instances, this oracle can easily be implemented in polynomial time, meaning that the number of oracle calls is a good measure of complexity. We develop two simple *randomized* algorithms whose expected number of calls to the oracle are of the order $f(n)N$, with N being the sum of sizes of the n sets whose product forms the grid, and with $f(n) \approx n!$.

The two algorithms are very similar in spirit; the first one generalizes an algorithm for the n-cube [21, 45], and the vertices visited by the algorithm generally do not form a path in the grid. The expected runtime only depends on the parameters of the grid, but not on the concrete USO. The second algorithm follows a directed path to the global sink. Compared to the first algorithm, its expected worst-case runtime is slightly higher, but it has the potential of being much faster if the starting vertex is already close to the sink, or if directed paths tend to be short in the USO under consideration.

In particular, if n is fixed, we get *linear-time* algorithms. Specialized to PGLCP, this corresponds to the case in which we have a fixed number of *blocks*; in this situation, we get the first algorithms whose expected complexity is linear in the number of variables—this is optimal. Even if n is not considered to be fixed, our algorithms improve over the trivial bound resulting from the evaluation of all edge orientations. To the best of our knowledge, no previously published algorithm could be proven to perform substantially better than the trivial approach, in the worst case. A reduction from PGLCP to PLCP was claimed [12] that would yield a quadratic bound of $O(N^2 + \mathrm{plcp}(n))$, where $\mathrm{plcp}(n)$ is the time necessary to solve a PLCP instance of size n. However, the proof contains an error, and there is a counterexample to the correctness of the reduction [13].

In the acyclic case, linear-time algorithms for fixed n are known. This follows from the fact that the problem of finding the sink in an AUSO can be formulated as an *LP-type problem* [36] in a natural way [2]. In the LP-type framework, $f(n)N$ oracle calls suffice, even deterministically [5]. The currently best algorithm for the acyclic case combines two randomized algorithms [7, 36] and requires an expected number of $O(Nn + f(n))$ oracle calls, where $f(n) = \exp(O(\sqrt{n \log n}))$ is a *subexponential* function [20]. No subexponential methods are known for USO that contain cycles.

Fig. 5 Two AUSO violating the Holt-Klee condition
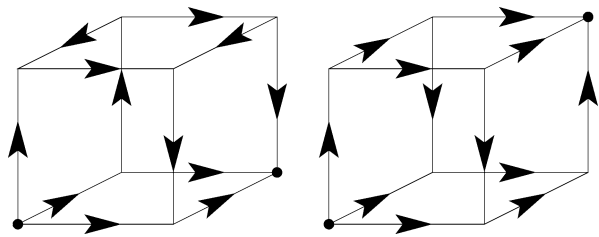
1.1.3 The Holt-Klee Condition

Unique sink orientations coming from LP over products of simplices satisfy an interesting combinatorial property: consider any subgrid along with its unique sink and unique source.[1] The *Holt-Klee* (HK) condition states that there are as many vertex-disjoint directed paths between source and sink as there are neighbors of the source (equivalently, the sink) in the subgrid [27]. This is obviously the maximum possible number of such paths. The HK condition does not hold for general AUSO; there are two nonequivalent AUSO of the 3-cube with only two vertex-disjoint directed paths from source to sink, see Fig. 5.

The HK condition is important, as it is the only known simple combinatorial condition that can distinguish geometric (LP- or PGLCP-induced) from abstract situations.

We show that all grid USO coming from PGLCP do satisfy the HK condition (we also say that they are HK), even in the presence of cycles. The proof works in the dual and more general setting of *complete pointed fans* (defined in Sect. 4.3). The result emphasizes the geometric nature of PGLCP and establishes a new combinatorial way of checking whether a given USO can be realized as PGLCP. This result is new also in the context of PLCP with its wide range of known theory and applications [10].

Even if a given USO is HK, the above result might allow us to conclude that it is not PGLCP-induced. For this, we identify certain classes of *edge flips* that transform PGLCP instances into other PGLCP instances. If we manage to obtain a non-HK USO through such flips, we know that the USO we started with is not PGLCP-induced.

The HK condition is necessary but not sufficient: there exist AUSO of the 4-cube which satisfy the HK condition but do not come from PLCP [38]. Asymptotically, for n tending to infinity, almost all n-cube AUSO that are HK are not PLCP-induced [11]. The stronger condition we get by including edge flips is not sufficient, either. Still, we show that it properly strengthens HK: there is a 4-cube AUSO which satisfies HK but can be shown to be not PLCP-induced through the edge flipping criterion.

The HK condition is involved in a prime example of how combinatorial information can be exploited algorithmically. Matoušek has constructed a class of n-cube AUSO with the property that the RANDOMFACET simplex algorithm [29] has superpolynomial expected runtime on a randomly chosen instance from the class [35]. This runtime is of the order $\exp(\Omega(\sqrt{n}))$ and matches the best known upper bound of $\exp(O(\sqrt{n}))$ for the expected performance of this algorithm on general n-cube

---

[1]The existence of a unique source follows from the USO axioms, see Sect. 2.2.

AUSO [17]. While this result settles the complexity of RANDOMFACET on n-cube AUSO, the question remained open whether the performance analysis can be improved when the AUSO is induced by an actual LP over a combinatorial cube. At least within Matoušek's class, this is the case. It can be shown that RANDOMFACET needs an expected number of only $O(n^2)$ pivot steps on all n-cube AUSO in the class that satisfy the HK condition [17]. It is open whether this result extends to larger classes of AUSO.

A general grid USO in dimension $n = 2$ is known to be HK if and only if no subgrid orientation is isomorphic to the orientation of Fig. 4 which we call the *double twist* [15, 46]. We extend this result to $n = 3$, proving that in order to satisfy the HK condition, it is necessary and sufficient to avoid the double twist as well as the two cubes of Fig. 5. For $n = 4$, we give an example of a minimal grid that violates HK but is not a cube.

1.1.4 The Planar Case

We show that no directed cycles can occur in a grid USO of dimension $n = 2$, meaning that the cube of Fig. 2 is in fact the smallest possible cyclic example. If the orientation is HK, this follows from the interpretation in terms of a pseudoline arrangement [15].

2 Basics

Throughout this paper, we fix two natural numbers $N \geq n \geq 1$ and an ordered partition

$$\Pi = (\Pi_1, \ldots, \Pi_n)$$

of the set $[N] := \{1, \ldots, N\}$ into n nonempty subsets. We also refer to Π_i as the *block i*.

A subset $J \subseteq [N]$ is called a Π-*vertex* (or simply *vertex*) if $|J \cap \Pi_i| = 1$ for all i. We also set

$$J_I = \bigcup_{i \in I} (J \cap \Pi_i), \quad I \subseteq [n].$$

In other words, J_I comprises the $|I|$ elements of J in the blocks indexed by I.

Let V be the set of vertices. The *grid* spanned by $S \subseteq [N]$ is the undirected graph

$$G(S) = (V(S), E(S)),$$

with

$$V(S) := \{J \in V \mid J \subseteq S\},$$

$$E(S) := \{\{J, J'\} \subseteq V(S) \mid |J \oplus J'| = 2\},$$

where $\oplus$ denotes symmetric difference. The vertices of $G(S)$ correspond to the elements of the Cartesian product

$$\prod_{i=1}^{n} S_i, \quad S_i := S \cap \Pi_i.$$

(Note that $J_i = J_{\{i\}}$.) Edges join pairs of vertices J, J' that differ in exactly one coordinate.

If $|S_i| \leq 2$ for all i, $G(S)$ is the graph of some k-cube, $k \leq n$. If $S_i = \emptyset$ for some i, $G(S)$ is the empty graph.

A *face* or *subgrid* of $G(S)$ is any graph of the form $G(S')$, for $S' \subseteq S$. Throughout, we abbreviate $G([N])$ as G and $E([N])$ as E.

Definition 2.1 Let ψ be an orientation of G.[2] ψ is called *unique sink orientation* (USO) if all subgraphs of ψ induced by nonempty faces of G have unique sinks. If ψ contains the arc (J, J'), we also write $J \xrightarrow{\psi} J'$.

In the remainder of this section, we introduce PGLCP as a generalization of LP over products of simplices, and we prove that any *generic* PGLCP instance gives rise to a USO. Subsequently, we prove further properties of general grid USO.

2.1 Grid LP and Generalized LCP

Consider a linear program in the variables $x = (x_1, \ldots, x_N)^T$, of the form

$$\begin{aligned}
\text{minimize} \quad & c^T x \\
\text{subject to} \quad & Ax = b, \\
& x \geq 0,
\end{aligned} \tag{1}$$

where $A \in \mathbb{R}^{n \times N}, b \in \mathbb{R}^n, c \in \mathbb{R}^N$. For $S \subseteq [N]$, let A_S denote the submatrix of columns indexed by S. Furthermore assume that every vertex $J \in V$ is a nondegenerate basis in (1), meaning that A_J is invertible and $A_J^{-1} b > 0$. We say that the LP is Π-*compatible*, and we call A_J a *representative submatrix*. We will assume that the ordering of the columns in A_J is compatible with Π, meaning that the ith column of A_J comes from $A_{\Pi_i}, i \in [n]$.

Using Cramer's rule, the following is not hard to establish (we omit the proof).

Observation 2.2 *Consider a Π-compatible LP of the form* (1). *Then*

(i) *All determinants* $\det(A_J)$ *of representative submatrices have the same* (*nonzero*) *sign, and*
(ii) *The vertices* $J \in V$ *are exactly the bases of the LP* (1).

---

[2]Formally, ψ is a digraph containing either the arc (J, J'), or the arc (J', J), for each edge $\{J, J'\} \in E$.

For any $c \in \mathbb{R}^N$, a canonical Π-compatible LP is obtained by setting $b_i = 1$ and

$$A_{ij} := \begin{cases} 1, & j \in \Pi_i, \\ 0, & \text{otherwise}, \end{cases} \quad i \in [n], \ j \in [N].$$

In this case, all representative submatrices are equal to the identity matrix I_n, and the feasible region is the product of n simplices, where the ith simplex is defined in the space of variables x_j, $j \in \Pi_i$, via the constraints

$$\sum_{j \in \Pi_i} x_j = 1, \quad x_j \geq 0. \tag{2}$$

In general, the feasible region of a Π-compatible LP is combinatorially equivalent to the product of n simplices with a total of N facets: as a consequence of Observation 2.2(ii), the vertex-edge graph of the feasible region is a grid. By nondegeneracy, the feasible region is therefore a simple polytope, sharing its graph with the product of simplices defined in (2). Because the combinatorial structure of a simple polytope is determined by its graph [4, 28], the statement follows.

A unique sink orientation of the grid G is obtained from a Π-compatible LP with *generic* objective function vector c, meaning that $c^T x$ is not constant on any edge. In this case, an edge can be directed towards its vertex of lower objective function value. The resulting orientation is a USO, with the unique sink of the face $G(S)$ corresponding to the unique optimal basis of the linear program

$$\begin{aligned} \text{minimize} \quad & c_S^T x_S \\ \text{subject to} \quad & A_S x_S = b, \\ & x_S \geq 0 \end{aligned}$$

in the $|S|$ variables $x_S := (x_j)_{j \in S}$.

The edge orientations in the USO are determined by *reduced cost coefficients* (see any description of the simplex method, for example the one in Chvátal's book [6]). More precisely, if J is a basis, the row vector

$$\bar{c}(J) := c^T - c_J^T A_J^{-1} A \tag{3}$$

is the *reduced cost vector* associated with J. Note that $\bar{c}(J)_J = 0$, and that $\bar{c}(J)_j \neq 0$ for $j \notin J$, if c is generic. Then, if J' is adjacent to J, with j being the unique index in $J' \setminus J$, we have

$$J \to J' \quad \Leftrightarrow \quad \bar{c}(J)_j < 0. \tag{4}$$

The existence of a unique sink K therefore provides us with a vector $y^T := c_K^T A_K^{-1}$ (a dual feasible solution) such that

(i) $c^T \geq y^T A$, and
(ii) For every $i \in [n]$, there is a $j \in \Pi_i$ with $c_j = (y^T A)_j$.

Note that the second set of constraints forces y^T to be of the form $c_J^T A_J^{-1}$ for some vertex J. By the unique sink property, K is the only choice for J that also guarantees property (i) and so there is a unique vector y that fulfills (i) and (ii).

If A and c do not come from a Π-compatible LP as in (1), a vector y satisfying properties (i) and (ii) need not exist. However, restricting A to be a matrix that—in a suitable sense—respects the partition Π, we arrive at the following generalization of Π-compatible LP.

Definition 2.3 Let $A \in \mathbb{R}^{n \times N}$ and $c \in \mathbb{R}^N$ such that A has *property* P, meaning that condition (i) of Observation 2.2 holds. The P-*generalized linear complementarity problem* (PGLCP) is the problem of finding a vector $y \in \mathbb{R}^n$ such that

$$c^T \geq y^T A, \tag{5}$$

and with the property that for every $i \in [n]$, there is a $j \in \Pi_i$ satisfying

$$c_j = (y^T A)_j. \tag{6}$$

Intuitively, PGLCP is LP of the form (1) 'without a right-hand side'. It will turn out that under this generalization, uniqueness of solution as well as the USO formulation persist. Because already the cyclic cube USO of Fig. 2 arises from a PGLCP (Stickney & Watson [43] give an example in an equivalent dual setting, see also Sect. 4.3), the generalization is proper: given a PGLCP instance (A, c), it is not always possible to find a right-hand side b such that A, b, c form a Π-compatible LP.

Theorem 2.4 *Let $A \in \mathbb{R}^{n \times N}$ and $c \in \mathbb{R}^N$ define a* PGLCP *instance.*

 (i) *There exists a unique vector $y \in \mathbb{R}^n$ satisfying conditions (5) and (6).*
(ii) *If c is generic (which means that $\bar{c}(J)_j \neq 0$ for $j \notin J \in V$), the edge orientations given by (4) define a unique sink orientation of the grid G.*

Proof Fix a vertex K, define

$$M^T := A_K^{-1} A,$$
$$q^T := \bar{c}(K) = c^T - c_K^T A_K^{-1} A,$$

and consider the problem of finding $z \in \mathbb{R}^n$, $w \in \mathbb{R}^N$ such that

$$w - Mz = q, \tag{7}$$

$$\prod_{j \in \Pi_i} w_j = 0, \quad i \in [n], \tag{8}$$

and

$$w, z \geq 0. \tag{9}$$

By definition, $q_k = 0$ and $(M^T)_k = \mathbf{e}_i$ for k being the unique element in $K \cap \Pi_i$, where $\mathbf{e}_i$ is the ith unit vector. Consequently, every solution of (7) must satisfy $z_i = w_k$ for $i \in [n]$. This means that (8) may be replaced with

$$z_i \prod_{j \in \Pi_i} w_j = 0, \quad i \in [n]. \tag{10}$$

Equations (7), (9) and (10) define the generalized linear complementarity problem in the setting of Cottle & Dantzig, who show that a feasible solution (w, z) exists if every representative submatrix of M^T is a P-*matrix* [9]. A matrix is a P-matrix if all determinants of principal minors are positive [10]. In our case, M^T satisfies this, which easily follows from the fact that M^T has property P *and* contains a representative identity matrix (see Definition 2.3).

To prove the existence of y in part (i), we choose a solution (w, z) of (7), (8) and (9). Then we set $y^T = (c_K^T - z^T) A_K^{-1}$ and get $w^T = c^T - y^T A$, using (7). Conditions (8) and (9) then ensure that y is the desired solution.

For a uniqueness proof, assume that distinct y and y' solve (5) and (6). Then the vector $(y' - y)^T A$ satisfies the property that in every block Π_i we have a zero element (if both $y^T A$ and $y'^T A$ agree with c^T on the same component in Π_i), or there is a positive and a negative element (if $y^T A$ and $y'^T A$ agree with c^T on different components; remember that $c^T - y^T A, c^T - y'^T A \geq 0$). If R and S are representative submatrices of A, then $R^{-1}S$ has to be a P-matrix if A is to have property P. We construct representative submatrices R and S of A for which this fails, as follows. For each i, if $(y' - y)^T A$ has a zero entry in block i, then choose columns R_i and S_i to be the column of A corresponding to this entry (so that $(y' - y)^T R_i = (y' - y)^T S_i = 0$).

Else, let columns R_i and S_i be columns of block i of A for which the corresponding components of $(y' - y)^T A$ have different nonzero signs (this means that $(y' - y)^T R_i \cdot (y' - y)^T S_i < 0$).

It follows that $((y' - y)^T R)_i ((y' - y)^T R(R^{-1}S))_i \leq 0$ for each i. The matrix $R^{-1}S$ is said to *reverse the sign* of the nonzero vector $(y' - y)^T R$. This implies (see [10], Thm. 3.3.4) that $R^{-1}S$ is not a P-matrix which contradicts property P of A.

Statement (ii) is a corollary of (i), because c being generic implies that the vector y from (i) can be expressed in the form $y^T = c_J^T A_J^{-1}$ for exactly one J. This set J is then a sink of G in the orientation defined by (4); moreover, there cannot be a second sink J', since that would correspond to a second solution $y'^T = c_{J'}^T A_{J'}^{-1}$ to (5) and (6), a contradiction to (i).

The fact that this orientation defines a USO easily follows: applying the above arguments to the PGLCP instance (A_S, c_S), we can prove the existence of a unique sink in the subgrid $G(S)$. ∎

Given a PGLCP instance (A, c) and $i \in [n]$, the replacement of submatrix A_{Π_i} with $-A_{\Pi_i}$ preserves property P. Let A' be the resulting matrix, and let c' be the vector resulting from c by replacing the subvector c_{Π_i} with its negative. It is easily seen from (3) and (4) that the USO induced by (A, c) and (A', c') differ exactly in the orientations of the edges along dimension i. This can be summarized in the following

Theorem 2.5 *Let ψ be a grid orientation, choose $i \in [n]$, and let ϕ be the orientation resulting from ψ by flipping all arcs of the form*

$$(J, J'), \quad J \oplus J' \subseteq \Pi_i.$$

Then ψ is a PGLCP-induced USO if and only if ϕ is a PGLCP-induced USO.

Theorem 2.12 proves that the class of general grid USO is closed under this flipping operation as well.

2.2 USO Properties

In this subsection, we study general grid USO according to Definition 2.1 and develop some basic theory. The material will also be used in later sections.

2.2.1 Outmap and h-Vector

Any USO can be specified by associating each vertex J with its outgoing edges. Given J and $j \in [N] \setminus J$, we define $J \triangleright j$ to be the unique vertex $J' \subseteq J \cup \{j\}$ which is different from J. If $j \in \Pi_i$, we have $J' = (J \setminus \Pi_i) \cup \{j\}$. We call J' the *neighbor* of J *in direction* j. Note that J is a neighbor of J' in some direction different from j.

Given an orientation ψ, the function $s_\psi : V \to 2^{[N]}$, defined via

$$s_\psi(J) := \{j \in [N] \setminus J \mid J \xrightarrow{\psi} J \triangleright j\}, \tag{11}$$

is called the *outmap* of ψ. Björklund et al. consider the outmap for acyclic grid USO and call it VID-function (vector of improving directions) [2]. Szabó and Welzl [45] deal with outmaps for cube USO; formally, these are different from ours, even when we specialize to the cube case, because Szabó and Welzl identify the n-cube vertices with the subsets of an n-element set, while we identify them with representative n-subsets of a $2n$-element set. Still, we can generalize the characterization of USO outmaps by Szabó and Welzl, using one more ingredient.

Definition 2.6 Let $s : V \to 2^{[N]}$. The vector $h(s) = (h_0(s), \ldots, h_{N-n}(s))$, defined via

$$h_k(s) = \#\{J \in V \mid |s(J)| = k\}$$

is called the *h-vector* of s. If $s = s_\psi$ for some orientation ψ on G, we also refer to $h(s)$ as $h(\psi)$.

The following result is well-known for the acyclic case [2, 26]. We prove it for general grid USO. The proof is the same as for the acyclic case, though, and we provide it for completeness.

Theorem 2.7 $h(\psi) = h(\psi')$ *for any two USO* ψ, ψ' *of* G.

Proof The *degree* of a nonempty subgrid $G(S)$ is the number $|S| - n$, or the degree of any vertex in $G(S)$. Let f_i denote the number of subgrids of degree i, $i = 0, \ldots, N - n$.

On the one hand, each subgrid has a unique sink w.r.t. ψ, so the number of sinks in degree-i subgrids is f_i. On the other hand, J is a sink in $G(S)$ if and only if $J \xleftarrow{\psi} J \triangleright j$ for all $j \in S \setminus J$. It follows that a vertex of indegree k appears as a sink in exactly $\binom{k}{i}$ degree-i subgrids.

Since the number of vertices of indegree k is $h_{N-n-k}(\psi)$, we obtain the equations

$$f_i = \sum_{k=0}^{N-n} \binom{k}{i} h_{N-n-k}(\psi), \quad i = 0, \ldots, N - n.$$

We write this as $f = U \cdot \bar{h}(\psi)$, where $f = (f_0, \ldots, f_{N-n})^T$, $\bar{h}(\psi) = (h_{N-n}(\psi), \ldots, h_0(\psi))^T$, and U is a square upper-triangular matrix with ones on the diagonal. Therefore, U is invertible, and we have that for any ψ

$$\bar{h}(\psi) = U^{-1} f := h(\Pi),$$

a quantity not depending on ψ. $\qquad\qquad\square$

As all USO on a grid have the same h-vector, we denote it by $h(\Pi)$, emphasizing the fact that it depends on the parameters of the grid only. An immediate corollary of this theorem is that the h-vector is symmetric, meaning that $h_k(\Pi) = h_{N-n-k}(\Pi)$ for all $k \in \{0, \ldots, N-n\}$. To see this, choose ψ as some PGLCP-induced USO; repeated application of Theorem 2.5 guarantees that the orientation ψ' in which all arcs are flipped is again PGLCP-induced, so that $h_k(\psi) = h_{N-n-k}(\psi')$ holds. In particular, w.r.t. any given USO, all nonempty subgrids of G also have unique *sources*.

Here is the characterization of functions s that are of the form s_ψ, for ψ being USO.

Lemma 2.8 *Let $s : V \to 2^{[N]}$ satisfy $s(J) \cap J = \emptyset$ for all $J \in V$. The function s is the outmap of a USO of G if and only if*

(i) $(s(J) \oplus s(J')) \cap (J \oplus J') \neq \emptyset$, *for all $J \neq J'$, and*
(ii) $h(s) = h(\Pi)$.

Proof If s is the outmap of a USO, (ii) follows from Theorem 2.7. To see (i), consider the cube $G(J \cup J')$ spanned by J and J', and assume w.l.o.g. that J is not the sink of that cube. Then there is $j \in J' \setminus J \subseteq J \oplus J'$ such that $j \in s(J)$, and because $j \notin s(J')$, (i) follows.

For the other direction, assume that s is a function satisfying (i) and (ii). s defines a set of arcs

$$D = \{(J, J \triangleright j) \mid J \in V, j \in s(J)\}.$$

We first show that D contains exactly one arc for each grid edge $\{J, J \triangleright j\}$, meaning that D defines a proper orientation ψ. From (i) (applied with $J' = J \triangleright j$), it follows that each edge generates at least one arc. On the other hand, we have

$$|D| = \sum_k k h_k(s) \stackrel{\text{(ii)}}{=} \sum_k k h_k(\Pi) = |E|,$$

because $h(\Pi)$ is defined via a proper orientation. It follows that each edge gets a unique orientation.

It remains to prove that ψ is USO. (i) implies that no face has more than one sink w.r.t. ψ: assuming there are sinks $J \neq J'$ in some face $G(S)$, it follows that $s(J) \cap S = s(J') \cap S = \emptyset$, which implies $(s(J) \oplus s(J')) \cap S = \emptyset$. Together with $J, J' \subseteq S$ (which implies $J \oplus J' \subseteq S$), we get a violation of (i).

Condition (ii) then guarantees that every face has exactly one sink. To see this, note that every J which is counted for $h_k(s)$ is the sink in exactly

$$2^{N-n-k}$$

faces, spanned by J and some subset of its incoming edges. In other words, the expression

$$\sum_k 2^{N-n-k} h_k(s) = \sum_k 2^{N-n-k} h_k(\Pi)$$

counts the total number of vertex-face pairs (J, F) such that J is a sink in F w.r.t. ψ. By the previous equation, this number equals the total number of nonempty faces, because in any USO, all nonempty faces have unique sinks. This again means that in ψ, each nonempty face has *on average* one sink, which—together with the previously derived fact that all faces have at most one sink—implies that ψ is USO. $\qquad\square$

We remark that condition (ii) in the lemma is necessary; unlike in the cube case [45], it is not implied by (i).

2.2.2 Projected USO

Let $I \subseteq [n]$. Two vertices $J, J' \in V$ are called *I-equivalent* if and only if $J_I = J'_I$. In other words, J and J' coincide after collapsing all dimensions $i \notin I$, or after *projecting* to the dimensions indexed by I.

This relation is an equivalence relation, and we call the equivalence classes of it the *hypervertices* generated by I. For example, the hypervertices generated by $I = [n]$ correspond to the original vertices, while $I = \emptyset$ induces one hypervertex containing all the vertices. Figure 3 depicts a partition of a grid into 1-dimensional hypervertices.

Definition 2.9 For $I \subseteq [n]$ and $J \in V$,

$$J^I := \{J' \in V \mid J_I = J'_I\}$$

is the hypervertex generated by I and J.

$$V^I := \{J^I \mid J \in V\}$$

denotes the set of all these hypervertices.

I defines for any subset $S \subseteq \bigcup_{i \in I} \Pi_i$ a *projected grid* $G^I(S) = (V^I(S), E^I(S))$ on the hypervertices, via

$$V^I(S) := \{J^I \in V^I \mid J_I \subseteq S\},$$
$$E^I(S) := \{\{J^I, J'^I\} \subseteq V^I(S) \mid |J_I \oplus J'_I| = 2\}.$$

By definition of a hypervertex, this graph is well-defined and isomorphic to the grid defined by the ordered partition $\Pi^I(S) = (S_i)_{i \in I}$. It is not hard to see that all edges in $E^I(S)$ are of the form

$$\{J^I, (J \triangleright j)^I\}, \quad j \in \bigcup_{i \in I} S_i \setminus J.$$

To obtain $G^I(S)$, one starts with the face $G(S \cup \bigcup_{i \in [n] \setminus I} \Pi_i)$ of the original grid and collapses all dimensions $i \in [n] \setminus I$. Vertices of the original grid that only differ w.r.t. collapsed dimensions are identified with each other, resulting in a single hypervertex

$$J^I = V\left(J_I \cup \bigcup_{i \in [n] \setminus I} \Pi_i\right).$$

As before, if $S = \bigcup_{i \in I} \Pi_i$, we may omit the parameter S from G^I, V^I and E^I. We also abuse notation and identify a hypervertex with the face of the original grid that it corresponds to. We are now prepared to introduce the important concept of a *projected orientation*.

Definition 2.10 Let ψ be an orientation of the grid G, $I \subseteq [n]$. The projected orientation ψ^I is defined on G^I via

$$J^I \xrightarrow{\psi^I} (J \rhd j)^I \quad \Leftrightarrow \quad K \xrightarrow{\psi} K \rhd j,$$

where K is the sink (w.r.t. ψ) of the face J^I.

In Fig. 3, the hypervertex sinks K correspond to the vertices in white.

Lemma 2.11 *If ψ is USO, then ψ^I is USO.*

Proof Any nonempty face of G^I corresponds to a projected grid $G^I(S)$, obtained by collapsing dimensions in the face $F = G(S \cup \bigcup_{i \in [n] \setminus I} \Pi_i)$ of the grid G. The hypervertex J^I that contains the sink of F w.r.t. ψ is a sink in ψ^I. Conversely, any sink w.r.t. ψ^I contains a vertex that is a sink in F w.r.t. ψ. This implies the USO property. □

Theorem 2.12 *Let ψ be a grid orientation, choose $i \in [n]$, and let ϕ be the orientation resulting from ψ by flipping all arcs of the form*

$$(J, J'), \quad J \oplus J' \subseteq \Pi_i.$$

Then ψ is USO if and only if ϕ is USO.

Proof Let $I = \{i\}$. By construction, ϕ^I arises from ψ^I by flipping the orientations of *all* edges of the 1-dimensional projected grid G^I. Assuming that ψ is USO, ψ^I is USO by the previous lemma, and from the existence of unique sources w.r.t. ψ^I (see the discussion after Theorem 2.7), it follows that ϕ^I is USO as well.

It remains to show that ϕ is USO. The only candidates for a sink of G w.r.t. ϕ are the unique sinks of the $|\Pi_i|$ hypervertices into which G is partitioned. Note that the orientations within these hypervertices are unaffected by the reorientation $\psi \to \phi$. But since ϕ^I is USO, *exactly one* of these hypervertices has no outgoing edge along dimension i w.r.t. ϕ, so its sink is the unique sink of G w.r.t. ϕ.

The same can be shown for any face of G, so the USO property of ϕ follows. □

2.2.3 Refined Index

The outmap value $s_\psi(J) \subseteq [N]$ of a vertex J w.r.t. some USO ψ is partitioned according to the dimensions of the grid (the sets Π_i). Accordingly, the outdegree of J can be refined to an n-vector of dimensional outdegrees, as follows.

Definition 2.13 Let ψ be a grid USO. The function

$$r_\psi : V \to \prod_{i=1}^{n} \{0, \ldots, |\Pi_i| - 1\},$$

with

$$r_\psi(J) = (|s_\psi(J) \cap \Pi_1|, \ldots, |s_\psi(J) \cap \Pi_n|), \quad J \in V$$

is called the *refined index* of ψ.

Unlike the outmap, the refined index is a mapping between two sets of the same size, so it is natural to ask whether this mapping is a bijection. This is true in the cube case (where the refined index is just a different way of writing the outmap) [45], and it also holds for grid AUSO [2], where Björklund et al. use the term *signature* (SIG). The proof of the latter result does not generalize to the case of general USO. We show that the refined index is a bijection for general USO, using the concept of projected orientations.

Theorem 2.14 *Let ψ be a grid USO. The refined index r_ψ is a bijection.*

Proof We proceed by induction on n. For $n = 1$, Theorem 2.7 implies $h_k(\psi) = 1$ for all $k \in \{0, \ldots, |\Pi_1| - 1\}$ (take any 1-dimensional USO to see this). In other words, for any k in this range, there is exactly one vertex whose outmap has size k. The result follows.

For $n > 1$, we set $I = [n] \setminus \{i\}$, for i arbitrarily chosen from $[n]$ and consider the $(n-1)$-dimensional grid G^I over the 1-dimensional hypervertices J^I, $J \in V$. Now fix some value $k \in \{0, \ldots, |\Pi_i| - 1\}$. Since the hypervertices are of dimension 1, we may w.l.o.g. choose the generating vertex J of any hypervertex J^I to be the unique vertex in J^I such that $|s_\psi(J) \cap \Pi_i| = k$.

Claim The orientation ϕ of G^I, defined via

$$J^I \overset{\phi}{\to} (J \rhd j)^I \quad \Leftrightarrow \quad J \overset{\psi}{\to} J \rhd j$$

is USO, see Fig. 6.

Note that for $k = 0$, the claim is Lemma 2.11 applied to the special case $I = [n] \setminus \{i\}$. For $k > 0$, we prove that G^I contains exactly one sink w.r.t. ϕ; the argument for a face $G^I(S)$ is the same, after restricting to the partition $\Pi^I(S)$.

First, let us assume that G^I contains two distinct sinks J^I, J'^I w.r.t. ϕ. This means that both J and J' have no outgoing edges along the dimensions indexed by $I = [n] \setminus \{i\}$, but k outgoing edges along dimension i, w.r.t. ψ.
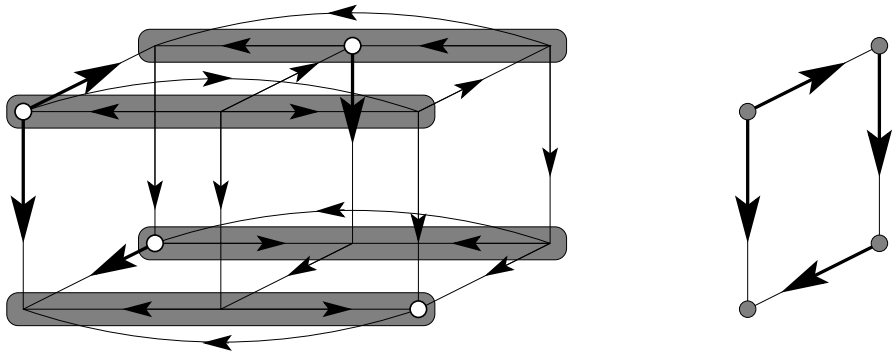
Fig. 6 A USO ψ on the left and its 'projected' USO ϕ generated by vertices (*white*) having refined index 1 w.r.t. their hypervertices (*shaded*)

If $J_i = J_i'$, then J and J' are in the same 'slice' (facet) $G(([N] \setminus \Pi_i) \cup J_i)$ of G w.r.t. dimension i, meaning that this slice contains two distinct sinks J, J' w.r.t. ψ, which is impossible. If $J_i \neq J_i'$, we consider the 1-dimensional grid $G^{\{i\}}$, in which J and J' are the sinks (w.r.t. ψ) of their respective $(n-1)$-dimensional hypervertices. However, both hypervertices have outdegree k in the projected USO $\psi^{\{i\}}$, contradicting $h_k(\psi^{\{i\}}) = 1$. The case in which G^I has no sink w.r.t. ϕ leads to $h_k(\psi^{\{i\}}) = 0$, a contradiction as well.

Having established the claim, we can resume the main proof. It is sufficient to show that no two distinct vertices $J, J' \in V$ have the same refined index value. If $r_\psi(J)_i \neq r_\psi(J')_i$, we are done. Otherwise, $r_\psi(J)_i = r_\psi(J')_i = k \in \{0, \ldots, \Pi_i - 1\}$, and the claim provides us with a USO ϕ on G^I for which $r_\phi(J^I) \neq r_\phi(J'^I)$ holds by induction (note that $J^I \neq J'^I$: as a consequence of $r_\psi(J)_i = r_\psi(J')_i$, J and J' cannot be in the same 1-dimensional hypervertex). By definition of ϕ,

$$r_\phi(J^I)_{i'} = r_\psi(J)_{i'}, \qquad r_\phi(J'^I)_{i'} = r_\psi(J')_{i'}, \quad i' \neq i,$$

and $r_\psi(J) \neq r_\psi(J')$ follows. $\square$

With this result, vertex K in Definition 2.10 of the projected orientation is the unique vertex in J^I satisfying $r_\psi(K)_i = 0, i \in [n] \setminus I$. The refined index bijection more generally implies that this definition also yields a USO if in all J^I, K is chosen as the unique vertex with $r_\psi(K)_i = k_i, i \in [n] \setminus I$, where the $k_i \in \{0, \ldots, |\Pi_i| - 1\}$ are any fixed numbers. The orientation ϕ derived in the proof of Theorem 2.14 is an instance of such a generalized projected USO.

3 Algorithms

In this section, we develop algorithms for finding the sink of a given grid USO, implicitly specified by either an *edge evaluation* or a *vertex evaluation* oracle. The edge evaluation oracle must be able to return the orientation of any given grid edge, while

the vertex evaluation oracle returns the outmap value of any given vertex. Our complexity measure will be the maximum (expected) number of oracle calls needed to find the sink in the worst case. In concrete instances, both oracles can typically be implemented to run in polynomial time. In the case of PGLCP, for example, an edge evaluation must return the sign of a single coefficient of the reduced cost vector (3), while a vertex evaluation computes all signs. Obviously, the necessary number of edge evaluations cannot exceed the number of vertex evaluations by a factor of more than $N - n$, but better bounds may be (and actually are) possible. We remark that for the cube case, Szabó and Welzl [45] only count vertex evaluations.

Under both oracles, the additional computations performed by our (very simple) algorithms do not generate asymptotic overhead, and this justifies our measure of complexity. In particular, any USO algorithm which calls the oracle only a polynomial number of times is actually a *strongly* polynomial algorithm for LP and PGLCP. Moreover, the complexity of a single edge evaluation typically only depends on n but not on N (in the PGLCP case, this complexity is $O(n^3)$ at most). Therefore, if n is considered to be a constant, any bound on the number of edge evaluations determines the complexity of the algorithm up to a constant factor. This explains why counting edge evaluations is our primary concern.

3.1 The PRODUCT Algorithm

This algorithm generalizes the *product algorithm* of Szabó and Welzl from cubes [45] to grids, with a slight twist: while in the n-cube, all dimensions are equivalent with respect to their size (which is two), a general grid may consist of large blocks Π_i (meaning that there are many elements in Π_i) and small blocks (with few elements in Π_i). The analysis will reveal why our algorithm gives priority to the large blocks. The basic idea is the following.

Choose $i \in [n]$ such that $|\Pi_i| > 1$, and choose $j \in \Pi_i$. Then recursively compute the sink K of the subgrid $G([N] \setminus \{j\})$, where $j \in \Pi_i$. If the edge incident to K in direction j is incoming, we have already found the global sink, otherwise we need to search the lower-dimensional slice $G(([N] \setminus \Pi_i) \cup \{j\})$ recursively. A generic call to PRODUCT finds the sink of a nonempty face $G(S)$. The correctness proof of the algorithm is a straightforward induction, using the USO properties. By the end of this subsection, it will also become clear why the algorithm is called PRODUCT. Recall that $S_i = S \cap \Pi_i$.

Algorithm 3.1

```
PRODUCT(S):
    (* Input: USO ψ on G(S), given by edge evaluation oracle *)
    (* Output: unique sink of G(S) w.r.t. ψ *)
    (* Invariant: S contains a vertex of G *)
    IF is_vertex(S) THEN
        RETURN S
    ELSE
        choose i ∈ [n] such that |S_i| ≥ |S_k| for all k ≠ i
        choose j ∈ S_i uniformly at random
```

```
      K := PRODUCT(S \ {j})
      K' := K ▷ j
      IF K' →ψ K THEN
          RETURN K
      ELSE
          RETURN PRODUCT((S \ S_i) ∪ {j})
      END
  END
```

To prepare the analysis of the algorithm, we need one more

Definition 3.2 Let $S \subseteq [N]$ be a set containing a vertex and let $z = \max_{i=1}^n |S_i|$ be the size of a largest block. The $(z-1)$-vector $(a_2, \ldots, a_z)$, defined through

$$a_t = |\{i \in [n] \mid |S_i| = t\}|$$

is called the *characteristic* of S. If $z = 1$ (meaning that S is a vertex itself), the characteristic is the empty vector ( ).

For example, the n-cube has characteristic (n), and the grid in our first Fig. 1 has characteristic $(2, 1)$. It can easily be shown by induction that the expected number of edge evaluations in PRODUCT(S) only depends on the characteristic of S but (maybe surprisingly) not on the input USO ψ. We can even compute the exact expectation.

Theorem 3.3 *For $z \geq 1$, let $T_e(a_2, \ldots, a_z)$ denote the expected number of edge evaluations in a call to PRODUCT(S), where S has characteristic $(a_2, \ldots, a_z)$. Then*

$$T_e(a_2, \ldots, a_z) = T_e(a_2, \ldots, a_{z-1} + 1, a_z - 1) + 1$$
$$+ \frac{1}{z} T_e(a_2, \ldots, a_{z-1}, a_z - 1), \quad z > 1, \qquad (12)$$

with

$$T_e(a_2, \ldots, a_{z-1}, 0) := T_e(a_2, \ldots, a_{z-1}) \quad and \quad T_e() = 0.$$

The solution to this recurrence is

$$T_e(a_2, \ldots, a_z) = \prod_{k=2}^{z} H_k^{a_k} + \sum_{k=2}^{z} \prod_{\ell=k}^{z} (H_\ell - H_{k-1} + 1)^{a_\ell} - z. \qquad (13)$$

Here, $H_k = \sum_{t=1}^{k} 1/t$ is the kth Harmonic number.

Proof It is clear that $(a_2, \ldots, a_{z-1} + 1, a_z - 1)$ and $(a_2, \ldots, a_{z-1}, a_z - 1)$ are the characteristics of the grids handled in the recursive calls. Moreover, the second recursive call is executed if and only if the global sink contains the chosen element j. This happens with probability $1/z$. The recurrence follows. The closed form (13) can be checked by induction.   □

As an example, consider $T_e(a_2)$ (the cube case). The theorem yields

$$T_e(a_2) = 2\left(\frac{3}{2}\right)^{a_2} - 2,$$

which is still exponential but much smaller than the total number $a_2 2^{a_2-1}$ of cube edges.

For fixed z, (13) is maximized if $a_z = n$. This corresponds to the characteristic $(0, \ldots, 0, n)$ of the $(z \times \cdots \times z)$-grid. It follows that

$$T_e(a_2, \ldots, a_z) \le H_z^n + \sum_{k=2}^{z}(H_z - H_{k-1} + 1)^n - z. \tag{14}$$

The middle term

$$f(n, z) := \sum_{k=2}^{z}(H_z - H_{k-1} + 1)^n \tag{15}$$

asymptotically dominates the bound in (14), and an estimate of $f(n, z) \le (z-1)H_z^n \approx z \ln^n z$ immediately follows. The next result shows that the bound is actually *linear* in z.

Lemma 3.4

$$T_e(a_2, \ldots, a_z) \le (\lfloor en! \rfloor - 1)z + H_z^n.$$

Proof Using the estimate

$$\sum_{t=\ell}^{u} g(t) \le \int_{\ell-1}^{u} g(x)dx \tag{16}$$

for any decreasing function g such that the integral exists, we can bound (15) as follows (the first inequality applies (16) with $\sum_{t=k}^{z} 1/t$).

$$
\begin{aligned}
f(n, z) &= \sum_{k=2}^{z}(H_z - H_{k-1} + 1)^n \le \sum_{k=2}^{z}(\ln z - \ln(k-1) + 1)^n \\
&= \sum_{k=1}^{z-1}\left(1 - \ln\frac{k}{z}\right)^n < \sum_{k=1}^{z}\left(1 - \ln\frac{k}{z}\right)^n \\
&\overset{(16)}{\le} \int_0^z \left(1 - \ln\frac{k}{z}\right)^n dk = z\int_0^1 (1 - \ln x)^n dx := z I_n.
\end{aligned}
$$

Integration by parts yields the recurrence relation

$$I_n = 1 + n I_{n-1},$$

with $I_0 = 1$. This solves to $I_n = \lfloor en! \rfloor$ for $n > 0$ [42]. The statement follows. $\square$

This means that algorithm PRODUCT solves any PGLCP instance with a fixed number n of blocks in expected time $O(z) = O(N)$ which is asymptotically optimal.

Algorithm 3.1 can easily be rewritten to work under a *vertex* evaluation oracle. For this, we simply need to call the oracle for the set S immediately before we RETURN S. This has the effect that all face sinks computed by recursive calls of the algorithm get evaluated before they are returned, meaning that all tests '$K' \xrightarrow{\psi} K$' come for free.

The resulting expected number $T_v(a_2, \ldots, a_z)$ of vertex evaluations satisfies the recurrence

$$T_v(a_2, \ldots, a_z) = T_v(a_2, \ldots, a_{z-1} + 1, a_z - 1) + \frac{1}{z} T_v(a_2, \ldots, a_{z-1}, a_z - 1), \quad (17)$$

with

$$T_v(a_2, \ldots, a_{z-1}, 0) := T_v(a_2, \ldots, a_{z-1}) \quad \text{and} \quad T_v() = 1.$$

Its solution is

$$T_v(a_2, \ldots, a_z) = \prod_{k=2}^{z} H_k^{a_k}. \quad (18)$$

For the cube with characteristic (a_2), we get

$$T_v(a_2) = H_2^{a_2} = \left(\frac{3}{2}\right)^{a_2}.$$

$T_v(a_2, \ldots, a_z)$ is maximized if $a_z = n$, from which we obtain the upper bound

$$T_v(a_2, \ldots, a_z) \leq H_z^n.$$

For fixed n, the expected number of vertex evaluations required to find the sink is therefore only *polylogarithmic* in z, equivalently in N. In order to turn this bound into an actual runtime bound, an $O(N)$ factor has to be included, though, which makes the algorithm superlinear. The reason is that under the vertex evaluation oracle, *all* edges incident to a vertex are evaluated, regardless of whether their orientations are needed later or not. Our previous analysis in terms of edge evaluations avoids this.

Algorithm 3.1 is a close relative of algorithms due to Seidel (for linear programming with n variables and N constraints) [41] and Welzl (for finding the smallest enclosing ball of a set of N points in dimension n) [47]. If $N \leq n + 1$, the smallest enclosing ball problem is known to induce a cube USO [19, 45], but for larger N, the problem has in general no apparent grid structure. The complexity bounds for Welzl's algorithm are very similar to ours and hold as upper bounds. In our scenario, the additional grid structure makes it possible to determine the complexity exactly.

Instead of choosing a largest block S_i in the vertex evaluation version of the algorithm, other rules could be applied. For example, we might choose a smallest block, or even a random one. It turns out that the expected number of vertex evaluations does not depend on the particular rule and always equals the expression in (18). In order to explain this, let us adopt a somewhat different view of the algorithm under the vertex evaluation oracle.

Let i be the largest (or any other) block and set $I = \{i\}$. We can consider the grid G as being the product of the projected grids G^I and $G^{[n]\setminus I}$. Algorithm PRODUCT can then be considered as an algorithm that *evaluates* the sink of the one-dimensional projected USO ψ^I of G^I. For this, it chooses a random hypervertex J^I in G^I (more precisely, the one where $J_i = j$) and recursively evaluates the sink K^I among the remaining $|\Pi_i| - 1$ hypervertices.[3] If K^I is not the sink w.r.t. ψ^I yet, one more evaluation of the true sink $(K \rhd j)^I = J^I$ is necessary. For each vertex evaluation w.r.t. ψ^I, the algorithm evaluates the sink of a hypervertex J^I (which has the structure of $G^{[n]\setminus I}$) w.r.t. the original USO ψ. The total (expected) number of vertex evaluations is therefore the *product* of the respective numbers for the grids G^I and $G^{[n]\setminus I}$. For G^I (a one-dimensional grid), this number is easily seen to be $H_{|\Pi_i|}$, from which (18) follows by induction.

Interestingly, the expected number of edge evaluations *does* depend on the choice of i, and a bad choice can lead to a complexity which is asymptotically worse than what we found in Theorem 3.3. For instance, always choosing i corresponding to the smallest block yields an expected number of

$$T_e(a_2, \ldots, a_z) = \sum_{k=1}^{z-1} k \left(\prod_{\ell=2}^{k} H_\ell^{a_\ell} \right) \sum_{m=0}^{a_{k+1}-1} H_{k+1}^m$$

edge evaluations, which is again largest for $a_z = n$. In this case we get

$$(z-1) \sum_{m=0}^{a_z-1} H_z^m = \frac{z-1}{H_z - 1}(H_z^n - 1),$$

which is superlinear in z and asymptotically not better than the obvious upper bound obtained from multiplying (18) by $N - n$. We believe (although we cannot prove it formally) that our choice of i in Algorithm 3.1 leads to the smallest possible expected number of edge evaluations.

3.2 The Algorithm RANDOMFACET

It is certainly pleasing from a mathematical point of view that the complexity of the PRODUCT algorithm does not depend on the USO ψ and can be evaluated exactly (Theorem 3.3). On the other hand, this means that the algorithm fails to deal with easy orientations ψ in the appropriate way. Consider for example the *uniform* USO ψ, given by

$$J \xrightarrow{\psi} (J \cup \{j\}) \setminus \{j'\} \quad \Leftrightarrow \quad j < j'.$$

The sink is the vertex J which collects from each block Π_i the smallest element. Moreover, any directed path in ψ has length at most $N - n$. It follows that any path-following algorithm will find the sink quickly, while the algorithm PRODUCT will be

---

[3]In this recursive call, the algorithm might proceed according to some dimension different from i, but inductively, we can argue that this does not influence the expected number of vertex evaluations, so we might as well stick to dimension i.

comparatively slow. In this subsection, we present a path-following algorithm whose expected worst-case complexity is not much larger than that of algorithm PRODUCT (but it has the potential of being much faster in practice). Its runtime is again $O(N)$ for fixed n.

The RANDOMFACET algorithm shares its basic idea with the PRODUCT algorithm. However, in addition to the current set S, it maintains a current vertex $J \in V(S)$ which may be replaced at some point by a neighbor of J along an outgoing edge.

In order to guarantee the invariant $J \in V(S)$, the element j which gets removed from S for the first recursive call must not be in J. This restricts j to the set $S_i \setminus J_i$, where the PRODUCT algorithm was free to choose $j \in S_i$. As before, the correctness of the algorithm is easily established through the USO axioms.

Algorithm 3.5
RANDOMFACET(J, S):
    (* Input: USO ψ on $G(S)$, given by edge evaluation oracle *)
    (* Output: unique sink of $G(S)$ w.r.t. ψ *)
    (* Invariant: $V \ni J \subseteq S$ *)
    IF $S = J$ THEN
        RETURN J
    ELSE
        choose $i \in [n]$ such that $|S_i| \geq |S_k|$ for all $k \neq i$
        choose $j \in S_i \setminus J_i$ uniformly at random
        $K := $ RANDOMFACET($J, S \setminus \{j\}$)
        $K' := K \rhd j$
        IF $K' \xrightarrow{\psi} K$ THEN
            RETURN K
        ELSE
            RETURN RANDOMFACET($K', (S \setminus S_i) \cup \{j\}$)
        END
    END

The following recurrence relation is derived as in the case of the PRODUCT algorithm, where the probability of the second recursive call is now *at most* $1/(z-1)$. The closed formula for the upper bound can again be proved inductively.

Theorem 3.6 *For $z \geq 1$, let $T'_e(a_2, \ldots, a_z)$ denote the expected number of edge evaluations in a call to* RANDOMFACET(S), *where S has characteristic $(a_2, \ldots, a_z)$. Then*

$$T'_e(a_2, \ldots, a_z) \leq T'_e(a_2, \ldots, a_{z-1}+1, a_z-1) + 1$$

$$+ \frac{1}{z-1} T'_e(a_2, \ldots, a_{z-1}, a_z-1), \quad z > 1, \qquad (19)$$

with

$$T'_e(a_2, \ldots, a_{z-1}, 0) := T'_e(a_2, \ldots, a_{z-1}) \quad and \quad T'_e() = 0.$$

An upper bound is

$$T'_e(a_2, \ldots, a_z) \le \sum_{k=1}^{z-1} \prod_{\ell=k}^{z-1} (H_\ell - H_{k-1} + 1)^{a_{\ell+1}} - z + 1. \tag{20}$$

As an example, let us consider the cube case again. The theorem yields

$$T'_e(a_2) \le 2^{a_2} - 1, \tag{21}$$

which is not better than the trivial upper bound that holds for any path-following algorithm. The reason is that j is 'randomly chosen' from a set of size one, so there is no randomness left. In fact, if we break ties in the choice of i by choosing i, for example, as the largest index for which $|S_i| = 2$, the bound in (21) is tight. A worst-case input is the *Klee-Minty cube*, an n-cube AUSO with a directed Hamiltonian path through all vertices. This AUSO is actually generated by an LP and was the first example to show that the simplex method with Dantzig's pivot rule may require an exponential number of pivot steps [32].

If we break ties *randomly*, the situation changes. For all cube AUSO, a *subexponential* upper bound of

$$e^{2\sqrt{a_2}} - 1$$

on the expected number of edge evaluations can then be shown [17]. It is open whether this bound (or any bound of the form $2^{o(a_2)}$) holds for general cube USO.

As in the case of the PRODUCT algorithm, the complexity is maximized if $a_z = n$, and an upper bound of

$$\sum_{k=1}^{z-1} (H_{z-1} - H_{k-1} + 1)^n - z + 1 = (H_{z-1} + 1)^n + \sum_{k=2}^{z} (H_{z-1} - H_{k-1} + 1)^n - z \tag{22}$$

on $T'_e(a_2, \ldots, a_z)$ holds. Comparing this with the expected number of edge evaluations (14) of PRODUCT, we see that for large z, both algorithms have approximately the same expected worst-case complexity which we have shown to be linear in z. However, only RANDOMFACET has the potential of being faster than the upper bound in practice.

The algorithm RANDOMFACET is a close relative of an algorithm by Matoušek, Sharir and Welzl for *LP-type problems* [36]. This combinatorial framework covers grid AUSO but also general LP and other (acyclic) problems which have no grid structure. While the PRODUCT algorithm is insensitive to directed cycles, RANDOM-FACET (more precisely, its present analysis) suffers to some extent: the exponential behavior in n (see Lemma 3.4 which also applies to RANDOMFACET with minor modifications) is much worse than the subexponential bounds one gets for LP-type problems [36]. In other words, not only for the cube ($z = 2$) but for any grid with small z, cycles make a huge difference. On the other hand, if z (and N) are large compared to n, our analysis of RANDOMFACET shows that cycles are much less problematic.

(A) (B) (C)
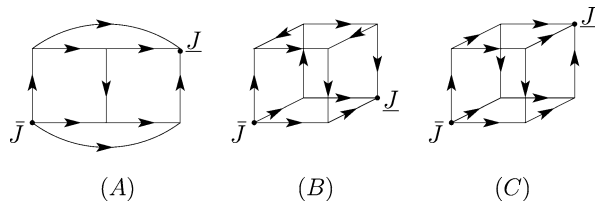
All complexity bounds we have developed in this section are of the form $f(n)N$, where $f(n)$ is exponential in n. We note that one can reduce these complexities to $O(Nn) + f(n)$, extending techniques by Clarkson [7] to deal with cyclic orientations [23].

4 The Holt-Klee Condition

A grid USO is said to be *Holt-Klee* (HK) if there is a set of $N - n$ vertex-disjoint paths from source to sink, and if in addition, every nonempty subgrid is HK. In this section, we prove a number of HK-related results for general USO as well as for specific USO coming from PGLCP.

4.1 Three-dimensional Grids

It is easy to prove [15, 46] that a 2-dimensional grid-USO is HK if and only if the double twist of Fig. 7A does not occur as a subgrid. We derive a similar statement for $n = 3$.

Theorem 4.1 *For $n = 3$, a grid USO ψ is HK if and only if no subgrid orientation is isomorphic to any of the three* forbidden subgrids *depicted in Fig.* 7.

Proof Assume the grid has dimensions $d_1 \times d_2 \times d_3$. It is clear that no forbidden subgrid can occur if the orientation is HK, since the forbidden subgrids in Fig. 7 are themselves not HK. For the other direction, it suffices to show that $N - n$ paths can be found in the full grid G; the same arguments then apply to the subgrids.

We distinguish three cases, depending on the positions of the global source $\bar{J}$ and the global sink $\underline{J}$ relative to each other.

Case 1 Source and sink span a 1-dimensional face. One can easily check that in this case, *any* grid USO has $N - n$ paths from source to sink. The proof only requires the 2×2 subgrids to be properly oriented.

Case 2 Source and sink span a 2-dimensional face. Without loss of generality, we can assume that source and sink agree in their third coordinate, see Fig. 8. The edge labeled 1 in Fig. 8 (left) must be oriented as indicated, otherwise the forbidden subgrid (A) occurs. In total, there are $(d_1 - 1) + (d_2 - 1)$ vertex-disjoint ways of drawing the figure, one for each neighbor of $\bar{J}$ in the face spanned by $\bar{J}$ and $\underline{J}$. The remaining $d_3 - 1$ paths are obtained by going through the third dimension, see Fig. 8 (right)

Fig. 8 Some of the $N - 3$ paths where sink and source span a 2-dimensional face
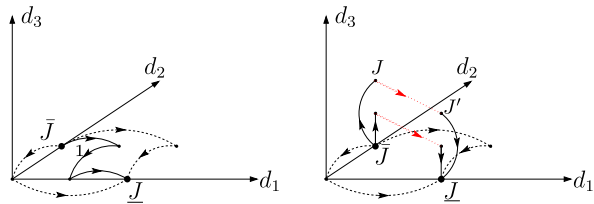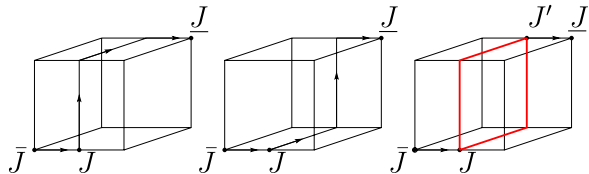

Fig. 9 Possible paths from $\bar{J}$ to $\underline{J}$ and the layer spanned by J and J', dividing the grid into two subgrids

where two of them are shown. Because the 3-cube spanned by $\bar{J}, \underline{J}, J, J'$, must be HK due to the absence of the forbidden cube (B) in Fig. 7, there is always a path from $\bar{J}$ to $\underline{J}$ in the upper facet of that cube (indicated by the dotted arrow). Here (and below) we use the fact that the two cubes of Fig. 7 are the *only* 3-cube USO that are not HK.

Case 3 Source and sink span a 3-dimensional face. For $i = 1, 2, 3$, $\bar{J}$ has $d_i - 2$ neighbors in dimension i which do not share a coordinate with $\underline{J}$. For any such neighbor J, we prove that there is a path from $\bar{J}$ to $\underline{J}$ through J, with the property that all intermediate vertices on the path share the i-th coordinate with J, see Fig. 9. This gives $N - 6$ vertex-disjoint paths, and as the forbidden subgrid (C) is not allowed to occur, we get three more paths in the cube spanned by $\bar{J}$ and $\underline{J}$.

Let us consider the *layer* spanned by the vertices J and J' in Fig. 9. If there is no path from J to J' in that layer, we may assume w.l.o.g. that the edges of the layer are oriented as in Fig. 10, case (i) or (ii). We will derive a contradiction in both cases.

The contradiction in case (i) is obtained as follows: edges 1 and 2 must be as given in order to avoid 2-cube orientations that are not USO ('bad' 2-cubes). Edge 3 prevents the right 3-cube from having two sinks and edge 4 circumvents the forbidden subgrid (A). Edges 5 and 6 again avoid bad 2-cubes and finally edge 7, avoiding a cyclic row, yields the forbidden subgrid (A).

The contradiction in case (ii) requires slightly more work: edges 1 and 2 avoid bad 2-cubes. Assume as in case (iii) of Fig. 10 that edge 3 is oriented upwards. Then edges a and b must have the same orientation. We must orient them to the left, as otherwise we get the forbidden subgrid (A). Choosing c and d such that bad 2-cubes are avoided results in the right 3-cube being the forbidden subgrid (B). So edge 3 must be oriented downwards.

Assume as in case (iv) that edge 4 is oriented upwards. Again, edges a and b must have the same orientation. In order to avoid (A), they are oriented to the left. Edge c prevents the left 3-cube from having two sources and d avoids a bad 2-cube. This forces the left 3-cube to be (B), which means that edge 4 must be oriented downwards.
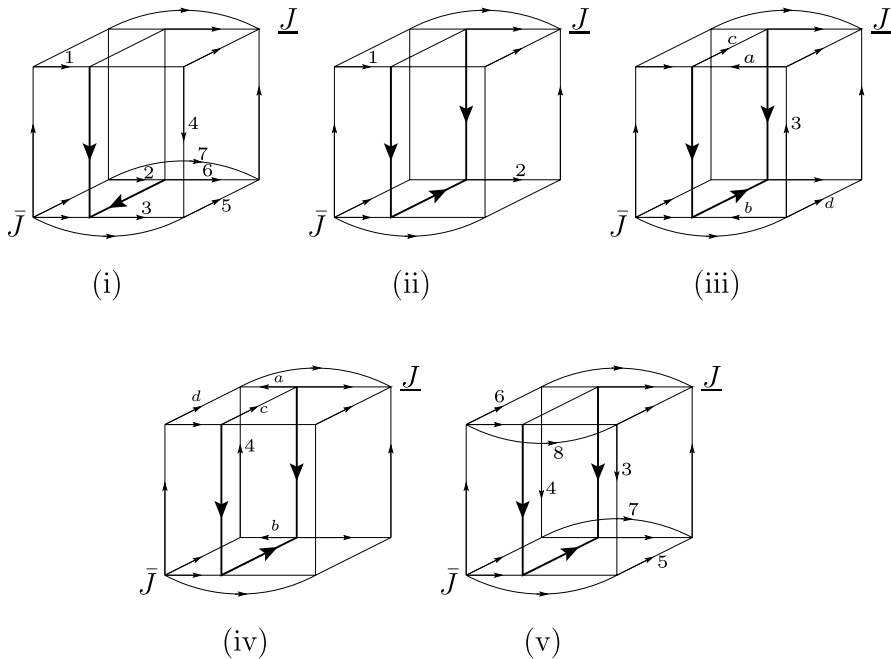
Fig. 10 Contradictions if there is no path from $\bar{J}$ to $\underline{J}$ through the layer of J, J'

Knowing the orientations of the edges 3 and 4 lets us fix edges 5–8 in case (v) (avoiding bad 2-cubes), resulting in the forbidden subgrid (C) for the 3-cube spanned by $\bar{J}$ and $\underline{J}$. □

4.2 Higher-dimensional Grids

While we have shown in the previous subsection that a 3-dimensional grid is HK if we exclude the double twist as well as the non-HK *cubes*, this does not hold anymore in dimension 4.

The smallest counterexample is the orientation of the $(3 \times 2 \times 2 \times 2)$-grid depicted in Fig. 11 (the big arrow indicates the common orientation of all edges between the left and right $3 \times 2 \times 2$ faces). The orientation itself is not HK (try to build two vertex-disjoint paths from source to sink, starting with edges 1 and 2), but all its faces are. In other words, this orientation is a minimal, non-cubical obstruction for the Holt-Klee condition in dimension $n = 4$. The example can be extended to yield a minimal non-cubical obstruction in any dimension $n \geq 4$.

An open question is whether there is a finite family of forbidden subgrids for given n whose absence makes any n-dimensional grid-USO HK.

4.3 PGLCP-induced Orientations

We show in this section that the grid USO coming from PGLCP have the Holt-Klee property. We actually prove this for a larger class of digraphs, those defined by *com-*

Fig. 11 The $(3 \times 2 \times 2 \times 2)$-grid not being HK
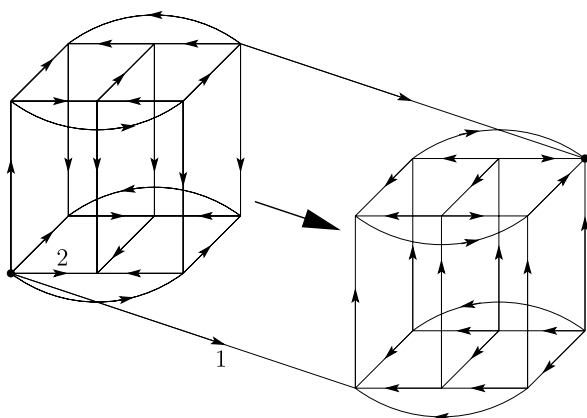
plete pointed fans. We will first show how the PGLCP case fits into this larger class, yielding interesting examples of non-polyhedral fans.

Following Sect. 2.1, we are given an $n \times N$ matrix A, an N-vector c and a partition Π of $[N]$ into n nonempty subsets. We seek a $y \in \mathbb{R}^n$ so that

$$c^T \geq y^T A \tag{23}$$

and

$$\text{for every } i \in [n], \text{ there is some } j \in \Pi_i \text{ with } c_j = (y^T A)_j. \tag{24}$$

The matrix A is assumed to have property P (in particular, it has full rank n). This implies that there is a unique vector $y \in \mathbb{R}^n$ satisfying conditions (23) and (24), see Theorem 2.4.

Let $\hat{A}$ be an $(N - n) \times N$ matrix for which the row space is the orthogonal complement in $\mathbb{R}^N$ of the row space of A. A set of columns of $\hat{A}$ indexed by a set K will be called *complementary* if $|\Pi_i \backslash K| \geq 1$ for all $i \in [n]$. A cone generated by a complementary set of columns of $\hat{A}$ will be called a *complementary cone*.

Note that the index set of a largest complementary set of $(N - n)$ columns is always the complement in $[N]$ of a Π-vertex. From property P and the construction of $\hat{A}$ it is not hard to deduce that such a complementary set of columns of $\hat{A}$ is linearly independent, so the complementary cones are all simplicial.

Here is the relation between this new setup and our original PGLCP setup (Sect. 2.1).

Claim Fix A and c. The following statements hold for all $y \in \mathbb{R}^n$.

(i) y satisfies condition (23) if and only if $q := \hat{A}c$ is contained in the cone generated by the columns of $\hat{A}$ that correspond to the nonzero components of $c - A^T y$.

(ii) y satisfies condition (24) if and only if $q := \hat{A}c$ is a linear combination of the complementary set of columns of $\hat{A}$ that correspond to the nonzero components of $c - A^T y$.

Putting (i) and (ii) together shows that the PGLCP with A and c can equivalently be expressed dually as the problem of finding $x \in \mathbb{R}^N$ such that $\hat{A}x = q, x \geq 0$, and with the complementarity property that for every $i \in [n]$, there is a $j \in \Pi_i$ with $x_j = 0$. We remark that this generalizes our reduction to PGLCP in the version of Cottle & Dantzig [9] from Sect. 2.1.

To prove the claim, we first fix y. We note that $\hat{A}(c - A^T y) = \hat{A}c = q$, by the choice of $\hat{A}$. If (23) holds, q is in the cone generated by the columns of $\hat{A}$ that correspond to the nonzero components of $c - A^T y$, and if (24) holds, q is a linear combination of complementary columns.

For the other direction, let x satisfy $\hat{A}x = q$. We get $\hat{A}(x - c) = q - q = 0$, so $(x - c)^T$ is in the row space of A and can be expressed as $-y^T A$ for some y. This gives $x^T = c^T - y^T A$. If $x \geq 0$, then (23) holds, and if the nonzero components of x correspond to a complementary set of columns of $\hat{A}$, we get (24).

The existence of the PGLCP solution for all c implies that the complementary cones cover $\mathbb{R}^{N-n}$, and the uniqueness implies that the intersection of two complementary cones is a complementary cone. The complementary cones therefore form a *complete pointed simplicial fan*. In proving our result for fans, we will replace, for the time being, the dimension $N - n$ by d. We follow the notation of Ziegler [49].

Definition 4.2 A *fan* in $\mathbb{R}^d$ is a family

$$\mathcal{F} = \{C_1, C_2, \ldots, C_t\}$$

of nonempty polyhedral cones, so that

 (i) Every nonempty face of a cone in $\mathcal{F}$ is also a cone in $\mathcal{F}$.
(ii) The intersection of any two cones in $\mathcal{F}$ is a face of both.

A fan $\mathcal{F}$ is *complete* if the union of its cones is $\mathbb{R}^d$. It is *pointed* if the zero vector is one of its cones. From now on, we will use the term fan to denote a complete pointed fan. A $d \times N$ matrix $\hat{A}$ is said to *realize* a fan $\mathcal{F}$ if the columns of $\hat{A}$ are a set of generators for the 1-dimensional faces of $\mathcal{F}$.

The *dual graph* $G_{\mathcal{F}}$ of a fan has as its vertex set the set of d-dimensional cones of $\mathcal{F}$, with two cones joined by an edge if their intersection is a $(d-1)$-dimensional face of $\mathcal{F}$.

For our PGLCP application, note that two $(N - n)$-dimensional complementary cones meet in an $(N - n - 1)$-dimensional complementary cone if and only if the complements of their index sets are adjacent in the graph G defined in Sect. 2. Therefore this dual graph is a grid graph.

A vector $q \in \mathbb{R}^d$ is said to be in general position with respect to $\mathcal{F}$ if it is not contained in any hyperplane that contains a $(d-1)$-dimensional cone of $\mathcal{F}$. If a vector q is in general position with respect to a fan $\mathcal{F}$, we can define an orientation $\Gamma_{q,\mathcal{F}}$ of the dual graph, in which an edge joining cones C and C' is oriented from C to C' if $C \backslash C'$ and q are on opposite sides of the hyperplane containing $C \cap C'$. The digraph $\Gamma_{q,\mathcal{F}}$ has a unique sink and source, which are the faces of $\mathcal{F}$ that contain q and $-q$.

We now show that this orientation coincides with the one defined in Sect. 2.1, if we start with the fan $\mathcal{F}$ consisting of all complementary cones determined by the matrix $\hat{A}$.

A generic vector c from our PGLCP application yields a vector $q \in \mathbb{R}^d$ in general position by the formula $q = \hat{A}c$. To determine the orientation of the edge joining cones C and C', we express q as a linear combination of the complementary columns that generate C. If this set of columns is indexed by K, we get $q = \hat{A}x = \hat{A}_K x_K$. This yields a unique vector x, since $\hat{A}_K$ is an invertible square matrix. Let x_k be the component of x corresponding to the unique element of K that is not in K', the index set of the complementary columns that generate C'. It is now easy to see that the edge is oriented from C to C' if and only if $x_k < 0$.

On the other hand, according to part (ii) of the above Claim, we can write $x^T = c^T - y^T A$, where y satisfies the complementarity condition (24). It follows that the grid orientation defined here is the same as that defined in Sect. 2.1 through the reduced costs in (4).

A set of directed paths from the source to the sink of a grid USO will be called *vertex-disjoint* if no two of the paths share any vertices other than the source or sink.

Theorem 4.3 *There is a set of d vertex disjoint directed paths in $\Gamma_{q,\mathcal{F}}$ from the source to the sink.*

The previous discussion immediately implies

Corollary 4.4 *Any PGLCP-induced grid USO ψ satisfies the Holt-Klee condition.*

The special case of the theorem in which $\mathcal{F}$ is the normal fan of a polytope was proved by Holt and Klee [27]. Our proof uses Menger's theorem, which was employed by Holt and Klee, but does not use the geometry of a polytope. Our proof may be seen as an alternate way to prove the Holt-Klee theorem. The graph $\Gamma_{q,\mathcal{F}}$ has been used by Kleinschmidt and Onn to prove that fans are *signable* [33]. Restricted to PGLCP-induced fans, their result simply says that the undirected grid graph Γ_F underlying $\Gamma_{q,\mathcal{F}}$ has a unique sink orientation, namely $\Gamma_{q,\mathcal{F}}$. Theorem 4.3 strengthens the result of Kleinschmidt and Onn by showing that the signings they produce have an interesting additional property.

In our PGLCP application, the complementary cones form the normal fan of a polytope when there is a vector b so that the LP: min $c^T x$ subject to $Ax = b, x \geq 0$ is Π-compatible. In this case, the associated digraph $\Gamma_{q,\mathcal{F}}$ is acyclic. The following example, taken from Stickney and Watson [43], shows that the orientations associated with PGLCP properly generalize those from Π-compatible linear programs, because its digraph $\Gamma_{q,\mathcal{F}}$ contains a cycle.

Example 4.5 Let

$$\hat{A} = \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & -2 \\ 0 & 1 & 0 & -2 & -1 & 0 \\ 0 & 0 & 1 & 0 & -2 & -1 \end{bmatrix}.$$

Set $n = 3$, choose $\Pi_i = \{i, i + 3\}$ for $i \in [n]$, and let $\mathcal{F}$ be the collection of complementary cones of $\hat{A}$. Then $\mathcal{F}$ is a fan in which the cones fit together like the faces of an octahedron. Let $q = (1, 1, 1)^T$. The digraph $\Gamma_{q,\mathcal{F}}$ is isomorphic to the one in Fig. 2.

For more exotic examples of fans, see Chap. III of Ewald's book [14]. We now return to the proof of the theorem. In the following, we assume that $\mathcal{F}$ is a complete pointed fan in $\mathbb{R}^d$ generated by the columns of a matrix $\hat{A}$, and that q is in general position with respect to $\mathcal{F}$. The following is the key lemma.

Lemma 4.6 *Let* $\mathcal{K} = \{K_1, K_2, \ldots, K_{d-1}\}$ *be* d*-dimensional cones of* $\mathcal{F}$*, and suppose that none of these cones contains* q *or* $-q$*. Then there exists a vector* w *orthogonal to* q *so that the line segments from* q *to* w *and from* $-q$ *to* w *both have empty intersection with each of the cones of* $\mathcal{K}$*.*

Proof We assume without loss of generality that q is the dth unit vector $(0, \ldots, 0, 1)$. Let $K \in \mathcal{K}$ and let $\hat{A}_K$ be the submatrix of $\hat{A}$ containing the generators of K. Now let $\overline{A}_K$ be the matrix obtained from $\hat{A}_K$ by deleting the last row. The columns of $\overline{A}_K$ are the projections of the columns of $\hat{A}_K$ onto the hyperplane H_q orthogonal to q, and the cone generated by these columns is the projection of the cone K.

Because neither q nor $-q$ is in K, the systems $\hat{A}_K x = q, x \geq 0$ and $\hat{A}_K x = -q, x \geq 0$ have no solution. The pointedness of the fan implies that the system $\hat{A}_K x = 0, x \geq 0$ has no nontrivial solution. It follows that the system $\overline{A}_K x = 0, x \geq 0, x \neq 0$ has no solution. By Gordan's Theorem [24] there exists a vector z_K so that $z_K^T \overline{A}_K > 0$. For such a z_K, any sufficiently small perturbation of it will also satisfy the inequality. Therefore we can find a linearly independent set $\{z_{K_1}, z_{K_2}, \ldots, z_{K_{d-1}}\}$ so that $z_{K_i}^T \overline{A}_{K_i} > 0$, $i = 1, \ldots, d - 1$. This implies $z_{K_i}^T w > 0$ for all nonzero vectors w in the cone generated by $\overline{A}_{K_i}$.

Now let Z be a $(d - 1) \times (d - 1)$ matrix that has as its rows the vectors $z_{K_i}^T$, $i = 1, \ldots, d - 1$. Stiemke's Theorem [44] says that the system $Zw \leq 0, Zw \neq 0$ has a solution if and only if the system $y^T Z = 0, y > 0$ has no solution. But the matrix Z is nonsingular, so the second system has no solution. Therefore there must be a vector $w \in H_q$ such that $z_{K_i}^T w \leq 0$ for all $i = 1, \ldots, d - 1$. This vector w is nonzero (because Zw is nonzero) and thus not contained in any of the cones $\overline{A}_K$ for $K \in \mathcal{K}$.

The vector w constructed in this way satisfies the requirements of the lemma, because any intersection of a cone of $\mathcal{K}$ with the line segment from q to w or the line segment from $-q$ to w would project to the cone generated by w and the cone generated by $\overline{A}_K$, contradicting the choice of w. In the proof of Theorem 4.3 we will also need that the set of w satisfying the requirements of the lemma is open: since the cones $\overline{A}_K$ are closed, the intersection of their complements in H_q is an open set. $\square$

Now we are ready to prove the main theorem.

Proof of Theorem 4.3 The directed vertex version of Menger's theorem states that there will be d disjoint directed paths from the source of $\Gamma_{q,\mathcal{F}}$ to the sink if and only if there do not exist $d - 1$ vertices of the graph other than the source and the sink that

cover all directed paths from the source to the sink. A set of $d-1$ vertices of $\Gamma_{q,\mathcal{F}}$ other than the source and the sink corresponds to a set $\mathcal{K} = \{K_1, K_2, \ldots, K_{d-1}\}$ as in the lemma. Because the set of vectors w satisfying the conditions of the lemma is open, we can choose a w for which the line segments from w to q and from w to $-q$ do not meet any cones of $\mathcal{F}$ of dimension less than $d-1$. We claim that the sequence of d-dimensional cones met by the directed line segment from $-q$ to w, followed by the sequence of d-dimensional cones met by the directed line segment from w to q, corresponds to a directed path from the source to the sink of $\Gamma_{q,\mathcal{F}}$. Suppose C_i and C_j are two d-dimensional cones of $\mathcal{F}$, and that the directed line segment from $-q$ to w crosses, in order, $(C_i, C_i \cap C_j, C_j)$. Then $C_i \backslash (C_i \cap C_j)$ and q are on opposite sides of the hyperplane spanned by $C_i \cap C_j$, so the edge of $\Gamma_{q,\mathcal{F}}$ connecting C_i and C_j is oriented from C_i to C_j. Similarly, the edges connecting cones met by the directed line segment from w to q are oriented consistently with the direction of the line segment. □

4.4 Projected USO and Edge Flipping

In this part, we present two examples showing that in general, the Holt-Klee property is lost under the operations of taking projected USO (Definition 2.9 and Lemma 2.11) and flipping edges (Theorem 2.12). For the projected USO, consider the LP

$$\begin{aligned}
\max \quad & z = \tfrac{101}{100}x_1 + \tfrac{99}{100}x_3 + x_5 + \tfrac{1}{2}x_6 \\
\text{subject to} \quad & x_1 + x_2 = 1, \\
& 2x_1 + x_3 + x_4 = 3, \\
& 2x_1 + 2x_3 + x_5 + x_6 + x_7 = 9, \\
& x_1, \ldots, x_7 \geq 0.
\end{aligned}$$

Let $n = 3$, $N = 7$ and $\Pi_1 = \{1, 2\}$, $\Pi_2 = \{3, 4\}$, $\Pi_3 = \{5, 6, 7\}$. Then the LP is seen to be Π-compatible and yields the unique sink orientation of Fig. 3 which can be verified to satisfy the HK condition.

We have already seen that the choice of the hypervertices as in the figure generates a projected orientation isomorphic to the double twist of Fig. 7A, which is not HK. Consequently, Corollary 4.4 tells us that projected USO coming from PGLCP-induced USO are not necessarily PGLCP-induced.

For the edge flipping operation, consider the 4-cube orientation of Fig. 12. Flipping all edges between the two 3-cubes in the figure leads to a USO that is not HK. Therefore, Corollary 4.4 together with Theorem 2.5 let us conclude that the HK-orientation of Fig. 12 is not PGLCP-induced. We therefore have a new and purely combinatorial necessary condition for PGLCP-induced orientations: all 2^n reorientations arising from edge flips according to Theorem 2.5 must be HK. This condition is not sufficient: there is an earlier example of a 4-cube USO which is HK but not PGLCP-induced, although all its reorientations are HK as well [38].
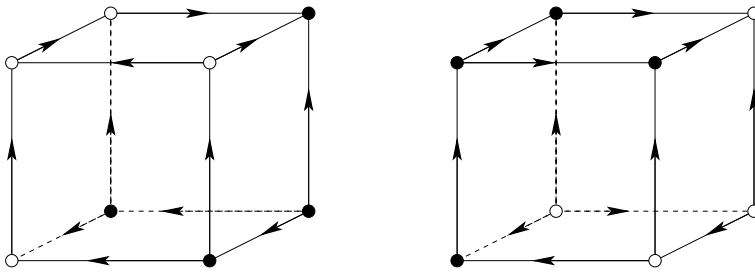
Fig. 12 An AUSO of the 4-cube satisfying the Holt-Klee condition. The arcs between the two depicted 3-cube facets go from the dark to the light vertices. Flipping the latter arcs leads to a USO not satisfying HK

5 Two-dimensional Grids

In this section we prove that no directed cycles can occur in a grid USO of dimension $n = 2$. Previously, this was only known to hold for USO of 2-dimensional grids that satisfy the Holt-Klee condition [15, 46].

Assuming that cycles are possible, let $\mathcal{C}$ be a shortest one. $\mathcal{C}$ will contain at most two vertices from each grid row and column, otherwise, a shortcut is possible. This implies that $\mathcal{C}$ alternates between vertical and horizontal edges, and that it has even length. This length must be at least 6, because we would get a cyclic 2-cube orientation otherwise, in direct contradiction to the USO property. After shuffling rows and columns, we may therefore assume that $\mathcal{C}$ has the form of a *staircase*, see Fig. 13. In the following, we restrict the grid to the rows and columns (at least three each) touched by $\mathcal{C}$.

From the earlier refined index result (Theorem 2.14), we can deduce the existence of a vertex J with all horizontal incident edges incoming, and all vertical incident edges outgoing. J must be a vertex of $\mathcal{C}$, otherwise we would immediately find a shorter cycle by connecting $\mathcal{C}$ to its two closest staircase vertices in vertical and horizontal direction. By swapping rows and columns again, we may assume that J is the vertex formed by the second row and the third column of the grid, and subsequently we focus on the 3×3-subgrid with 'center' J, see Fig. 13.

By definition of J, we already know the orientations of two off-staircase edges, see Fig. 14 (left). We now claim that this partial orientation cannot be completed to a USO, yielding the desired contradiction. For this, we show that the USO properties together with the assumption that $\mathcal{C}$ is a shortest cycle determine a partial orientation that cannot be completed to a USO. Figure 14 (right) depicts the order in which we 'fix' the orientations.

In order to avoid bad 2-cubes, the edges labeled 1 and 2 must be oriented as in the figure. The orientations of edges 3 and 4 prevent shortcuts of the cycle that bypass J. Then, having edge 5 the other way around would (prematurely) fix edge 7 through a suitable 2-cube, resulting in a shortcut of $\mathcal{C}$ along the boundary of the grid. Edge 6 keeps the first column cycle-free. The orientation of edge 7 then avoids a bad 2-cube again, and in order for the last row to remain acyclic, edge 8 must be as in the figure. The contradiction occurs in form of two sinks K and K' in the 2-cube spanned by them.

Fig. 13 The shortest cycle in a 2-dimensional grid and the subgrid defined by the special vertex J
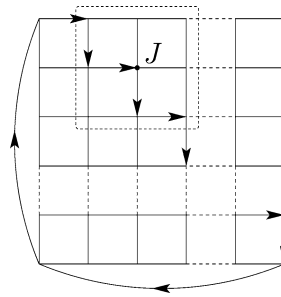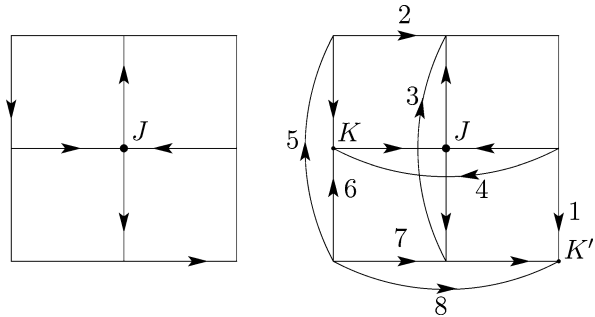
Fig. 14 Contradiction in the subgrid

We have proved the following

Theorem 5.1 *Any unique sink orientation of a 2-dimensional grid is acyclic.*

References

1. Adler, I., Saigal, R.: Long monotone paths in abstract polytopes. Math. Oper. Res. **1**(1), 89–95 (1976)
2. Björklund, H., Sandberg, S., Vorobyov, S.: Randomized subexponential algorithms for parity games. Technical Report TR-2003-019, Department of Information Technology, Uppsala University (2003)
3. Björklund, H., Sandberg, S., Vorobyov, S.: A combinatorial strongly subexponential strategy improvement algorithm for mean payoff games. In: Proc. 29th International Symposium on Mathematical Foundations of Computer Science (MFCS). Lecture Notes in Computer Science, vol. 3153, pp. 673–685. Springer, Berlin (2004)
4. Blind, R., Mani-Levitska, P.: On puzzles and polytope isomorphisms. Aequ. Math. **34**, 287–297 (1987)
5. Chazelle, B., Matoušek, J.: On linear-time deterministic algorithms for optimization problems in fixed dimension. J. Algorithms **21**, 579–597 (1996)
6. Chvátal, V.: Linear Programming. Freeman, New York (1983)
7. Clarkson, K.L.: Las Vegas algorithms for linear and integer programming. J. ACM **42**, 488–499 (1995)
8. Condon, A.: On algorithms for simple stochastic games. In: Cai, J.-Y. (ed.) Advances in Computational Complexity Theory. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, vol. 13, pp. 51–73. American Mathematical Society, Providence (1993)
9. Cottle, R.W., Dantzig, G.B.: A generalization of the linear complementarity problem. J. Comb. Theory **8**, 79–90 (1970)

10. Cottle, R.W., Pang, J., Stone, R.E.: The Linear Complementarity Problem. Academic, San Diego (1992)
11. Develin, M.: LP-orientations of cubes and crosspolytopes. Adv. Geom. **4**, 459–468 (2004)
12. Ebiefung, A.A.: Existence theory and Q-matrix characterization for the generalized linear complementarity problem. Linear Algebra Appl. **223/224**, 155–169 (1995)
13. Ebiefung, A.A.: Existence theory and Q-matrix characterization for the generalized linear complementarity problem: Revisited. Manuscript (2005)
14. Ewald, G.: Combinatorial Convexity and Algebraic Geometry. Springer, Berlin (1996)
15. Felsner, S., Gärtner, B., Tschirschnitz, F.: Grid orientations, $(d, d + 2)$-polytopes, and arrangements of pseudolines. Discrete Comput. Geom. **34**(3), 411–437 (2005)
16. Fischer, K., Gärtner, B.: The smallest enclosing ball of balls: combinatorial structure and algorithms. Int. J. Comput. Geom. Appl. **14**(4&5), 341–378 (2004)
17. Gärtner, B.: The Random-Facet simplex algorithm on combinatorial cubes. Random Struct. Algorithms **20**(3), (2002)
18. Gärtner, B., Rüst, L.: Simple stochastic games and P-matrix generalized linear complementarity problems. In: Proc. 15th International Symposium on Fundamentals of Computation Theory (FCT). Lecture Notes in Computer Science, vol. 3623, pp. 209–220. Springer, Berlin (2005)
19. Gärtner, B., Schurr, I.: Linear programming and unique sink orientations. In: Proc. 17th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 749–757, 2006
20. Gärtner, B., Welzl, E.: Linear programming—randomization and abstract frameworks. In: Proc. 13th Ann. ACM Symp. Theoretical Aspects of Computer Science. Lecture Notes in Computer Science, vol. 1046, pp. 669–687. Springer, Berlin (1996)
21. Gärtner, B., Welzl, E.: Explicit and implicit enforcing—randomized optimization. In: Lecture Notes of the Graduate Program Computational Discrete Mathematics. Lecture Notes in Computer Science, vol. 2122, pp. 26–49. Springer, Berlin (2001)
22. Gärtner, B., Solymosi, J., Tschirschnitz, F., Valtr, P., Welzl, E.: One line and n points. Random Struct. Algorithms **23**(4), 453–471 (2003) (preliminary version at STOC 2001)
23. Gärtner, B., Matoušek, J., Rüst, L., Škovroň, P.: Violator spaces: structure and algorithms. In: Proc. 14th Annual European Symposium on Algorithms (ESA). Lecture Notes in Computer Science. Springer, Berlin (2006)
24. Gordan, P.: Über die Auflösung linearer Gleichungen mit reellen Coefficienten. Math. Ann. **6**, 23–28 (1873)
25. Grädel, E., Thomas, W., Wilke, T.: Automata, Logics, and Infinite Games. Lecture Notes in Computer Science, vol. 2500. Springer, Berlin (2002)
26. Hoke, K.W.: Completely unimodal numberings of a simple polytope. Discrete Appl. Math. **20**, 69–81 (1988)
27. Holt, F., Klee, V.: A proof of the strict monotone 4-step conjecture. In: Chazelle, J., Goodman, J.B., Pollack, R. (eds.) Advances in Discrete and Computational Geometry. Contemporary Mathematics. Am. Math. Soc., Providence (1998)
28. Kalai, G.: A simple way to tell a simple polytope from its graph. J. Comb. Theory **49**, 381–383 (1988)
29. Kalai, G.: Linear programming, the simplex algorithm and simple polytopes. Math. Program. **79**, 217–233 (1997)
30. Kalai, G., Kleitman, D.J.: A quasi-polynomial bound for the diameter of graphs of polyhedra. Bull. Am. Math. Soc. **26**, 315–316 (1992)
31. Khachiyan, L.G.: Polynomial algorithms in linear programming. USSR Comput. Math. Math. Phys. **20**, 53–72 (1980)
32. Klee, V., Minty, G.J.: How good is the simplex algorithm? In: Shisha, O. (ed.) Inequalities III, pp. 159–175. Academic, San Diego (1972)
33. Kleinschmidt, P., Onn, S.: Signable posets and partitionable simplicial complexes. Discrete Comput. Geom. **15**, 443–466 (1996)
34. Ludwig, W.: A subexponential randomized algorithm for the simple stochastic game problem. Inform. Comput. **117**, 151–155 (1995)
35. Matoušek, J.: Lower bounds for a subexponential optimization algorithm. Random Struct. Algorithms **5**(4), 591–607 (1994)
36. Matoušek, J., Sharir, M., Welzl, E.: A subexponential bound for linear programming. Algorithmica **16**, 498–516 (1996)
37. Megiddo, N.: A note on the complexity of P-matrix LCP and computing an equilibrium. Technical Report, IBM Almaden Research Center, San Jose (1988)

38. Morris, W.D. Jr.: Distinguishing cube orientations arising from linear programs. Manuscript (2002)
39. Schäfer, U.: A linear complementarity problem with a P-matrix. SIAM Rev. **46**(2), 189–201 (2004)
40. Schurr, I., Szabó, T.: Jumping doesn't help in abstract cubes. In: Proc. 11th Conference on Integer Programming and Combinatorial Optimization (IPCO). Lecture Notes in Computer Science, vol. 3509, pp. 225–235. Springer, Berlin (2005)
41. Seidel, R.: Small-dimensional linear programming and convex hulls made easy. Discrete Comput. Geom. **6**, 423–434 (1991)
42. Sloane, N.J.A.: Sequence A000522. The On-Line Encyclopedia of Integer Sequences. http://www.research.att.com/~njas/sequences/ (2004)
43. Stickney, A., Watson, L.: Digraph models of Bard-type algorithms for the linear complementarity problem. Math. Oper. Res. **3**, 322–333 (1978)
44. Stiemke, E.: Über positive Lösungen homogener linearer Gleichungen. J. Reine Angew. Math. **76**, 340–342 (1915)
45. Szabó, T., Welzl, E.: Unique sink orientations of cubes. In: Proc. 42nd IEEE Symp. on Foundations of Comput. Sci., pp. 547–555, 2000
46. Tschirschnitz, F.: LP-related properties of polytopes with few facets. PhD Thesis, ETH Zürich (2003)
47. Welzl, E.: Smallest enclosing disks (balls and ellipsoids). In: Maurer, H. (ed.) New Results and New Trends in Computer Science. Lecture Notes in Computer Science, vol. 555, pp. 359–370. Springer, Berlin (1991)
48. Wiedemann, D.: Unimodal set-functions. Congr. Numer. **50**, 165–169 (1985)
49. Ziegler, G.M.: Lectures on Polytopes. Springer, Berlin (1995)