Unique Sink Orientations of Cubes

Tibor Szabó* Emo Welzl
Theoretical Computer Science, ETH Zürich
CH-8092 Zürich, Switzerland
{szabo,emo}@inf.ethz.ch

September 4, 2001

Abstract

Suppose we are given (the edge graph of) an n-dimensional hypercube with its edges oriented so that every face has a unique sink. Such an orientation is called a *unique sink orientation*, and we are interested in finding the unique sink of the whole cube, when the orientation is given implicitly. The basic operation available is the so-called *vertex evaluation*, where we can access an arbitrary vertex of the cube, for which we obtain the orientations of the incident edges.

Unique sink orientations occur when the edges of a deformed geometric n-dimensional cube (i.e., a polytope with the combinatorial structure of a cube) are oriented according to some generic linear function. These orientations are easily seen to be acyclic. The main motivation for studying unique sink orientations are certain linear complementarity problems, which allow this combinatorial abstraction (due to Alan Stickney and Layne Watson), where orientations *with cycles* can arise. Similarly, some quadratic optimization problems, like computing the smallest enclosing ball of a finite point set, can be formulated as finding a sink in a unique sink orientation (with cycles possible).

For acyclic unique sink orientations, randomized procedures due to Bernd Gärtner with an expected number of at most $e^{2\sqrt{n}}$ vertex evaluations have been known [3, 4]. For the general case, a simple randomized $(3/2)^n$ procedure exists (without explicit mention in the literature). We present new algorithms, a deterministic $O(1.61^n)$ procedure and a randomized $O((43/20)^{n/2}) = O(1.47^n)$ procedure for unique sink orientations. An interesting aspect of these algorithms is that they do not proceed on a path to the sink (in a simplex-like fashion), but they exploit the potential of random access (in the sense of arbitrary access) to any vertex of the cube. We consider this feature the main contribution of the paper.

We believe that unique sink orientations have a rich structure, and there is ample space for improvement on the bounds given above.

1 Introduction

Basic Notations and Definitions. We like to think of combinatorial cubes as the Boolean lattice. More specifically, for sets A and B let $[A, B] := \{X \mid A \subseteq X \subseteq B\}$. Given finite sets $A \subseteq B$, the cube $C = C^{[A,B]}$ is the edge-labeled graph with vertex set $\mathrm{vert} C := [A, B]$, edge set

$$\{\{u, u \oplus \{a\}\} \mid u \in \mathrm{vert} C, a \in B \setminus A\} ,$$

and edge labeling

$$\lambda : \quad \{u, u \oplus \{a\}\} \mapsto a .$$

Here $\oplus$ denotes the symmetric difference of two sets. The set $B \setminus A$ is called the *carrier of* C, denoted by $\mathrm{carr} C$. The cardinality of $\mathrm{carr} C$ is the *dimension of* C, denoted by $\dim C$. The subgraphs of C induced by sets $[X, Y]$, for $A \subseteq X \subseteq Y \subseteq B$, are cubes. They are called the faces of C, and we use i-face short for face of dimension i; 0-faces are *vertices*, 1-faces are *edges*, $(\dim C - 1)$-faces are *facets*. n*-Cube* is short for n-dimensional cube. Two faces F and F' induced by vertex sets $[X, Y]$ and $[X', Y']$, resp., are called *antipodal*, if $X' = B \setminus (Y \setminus A)$ and $Y' = B \setminus (X \setminus A)$; hence, $\dim F = \dim F'$ and $\mathrm{carr} F = \mathrm{carr} F'$.

*Supported by the joint Berlin/Zurich graduate program Combinatorics, Geometry, and Computation (CGC), financed by German Science Foundation (DFG) and ETH Zurich, and by NSF grant DMS 99-70270.

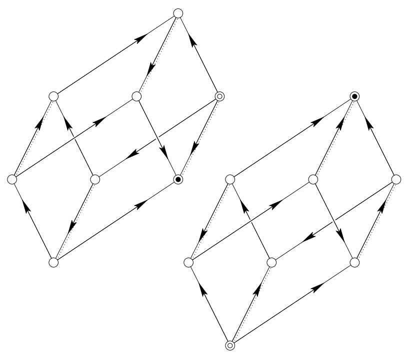

Figure 1: Two unique sink orientations of the 3-cube (the right one with a cycle).

An orientation of the edges of C is called *unique sink orientation of* C if every face[1] has a unique sink w.r.t. this orientation. A sink is a vertex without outgoing edges. See Figure 1 for examples of unique sink orientations.

The Problem. Given a unique sink orientation of an n-dimensional cube C, we want to find its unique sink. We assume that the orientation is given implicitly, and we can learn about the orientation by *vertex evaluations*, where we can access an arbitrary vertex of the cube, for which we obtain the orientations of the incident edges. The goal is to design algorithms that evaluate the sink of the cube with a small number of vertex evaluations, preferably much smaller than going through all 2^n vertices.

For a reason to become apparent only later, we are always required to perform an evaluation on the sink, even if we know already which vertex it is. Let us denote by $t(n)$ the smallest number, such that there is a procedure that evaluates the sink of any unique sink orientation of an n-cube with at most $t(n)$ vertex evaluations. We have $t(0) = 1$ and $t(1) = 2$, where we recall once more that we have to evaluate the sink in any case. In general $t(n) \leq 2^{n-1} + 1$ for $n \geq 1$: There are 2^{n-1} vertices that cover all edges (2-color the graph, and take one color-class). After evaluation of such a set of 2^{n-1} vertices we know the orientations of all edges, and so we know the sink, which we have to evaluate, if we haven't already done so.

We are also interested in randomized algorithms. Let $\tilde{t}(n)$ be the smallest number, such that there is a randomized procedure that evaluates the sink of any unique sink orientation of an n-cube with an expected number of at most $\tilde{t}(n)$ vertex evaluations. $\tilde{t}(0) = 1$, $\tilde{t}(1) = 3/2$, and, in general, $\tilde{t}(n) \leq 3(2^{n-1} + 1)/4$ is easy to obtain: First choose one of the two color classes, each with equal probability; then scan through the vertices in this color class in random order until the sink is hit; of course, with probability $1/2$, the sink is in the other color class, when we have to do one extra evaluation.

No bounds of the order $o(2^n)$ were known for $t(n)$. For $\tilde{t}(n)$ a bound of $(3/2)^n$ is relatively easy to obtain, although it seems not be mentioned explicitly in the literature (the procedure ForceOrNot in [9] can be interpreted as such an algorithm yielding this bound). One observation to make is that, in general, unique sink orientations are not acyclic (see Figure 1). Bernd Gärtner [3] has given a randomized procedure with a bound of $e^{2\sqrt{n}}$ for acyclic orientations, but as for deterministic bounds, nothing is known even for acyclic orientations.

The goal of this paper is to show that the problem has sufficient structure to derive nontrivial bounds. In particular, we show that $t(n) = O(1.61^n)$ and $\tilde{t}(n) = O(1.47^n)$. No lower bounds are known; so even upper bounds polynomial in n seem to be possible at this point.

Recently, Volker Kaibel [10] has independently derived a deterministic $O(1.73^n)$ algorithm.

Motivation. Unique sink orientations occur when the edges of a deformed geometric n-dimensional cube (i.e., a polytope with the combinatorial structure of a cube) are oriented according to some generic linear function. These orientations are easily seen to be acyclic. Motivated by the simplex algorithm, acyclic unique sink orientations have been studied for general polytopes, where they are called *abstract objective functions* [6, 11] or *completely unimodal numberings* [17]. Probably the most famous such orientation of a cube is the one obtained via the *Klee-Minty cube*. A randomized simplex variant (random edge) for that specific orientation has been quite intensively investigated, see e.g. [5], although even here no complete solution is known.

The main motivation for studying unique sink orientations are certain *linear complementarity problems* [1] (those defined by so-called P-matrices), which allow this combinatorial abstraction due to Stickney and Watson [16]; here

[1] Note that the whole cube is a face of itself!

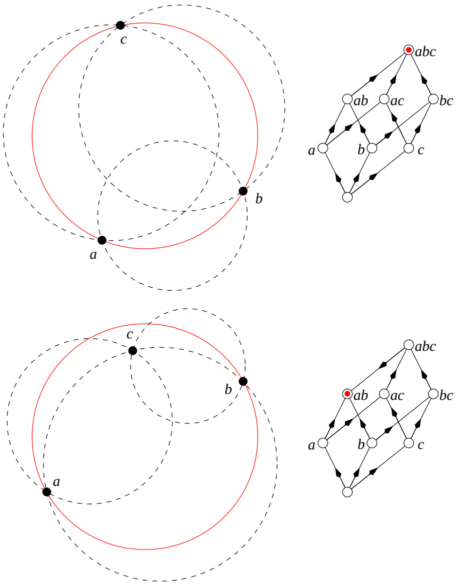

Figure 2: Unique sink orientations from smallest enclosing balls.

orientations *with cycles* do arise.

Similarly, some quadratic optimization problems, like *computing the smallest enclosing ball* of a finite point set, can be formulated as finding a sink in a unique sink orientation (with cycles possible). It is perhaps instructive to see how such a transformation can be done. Suppose we are given a set P of n affinely independent points in $(n-1)$-space[2], and we are interested in computing the smallest enclosing ball of P. This ball is unique, and there is a subset S_0 of P, such that the smallest enclosing ball of P is the smallest ball with S_0 on its boundary. Computing the smallest ball $\beta(Q)$ with a given set Q of affinely independent points on its boundary is easy: We compute the unique such ball in the affine hull of Q and its center is the center of the requested ball. So how can we obtain S_0?

Consider the cube $C^{[\emptyset,P]}$ and define an orientation of its edges as follows (for an illustration of the planar case see Figure 2): Given $x \in Q \subseteq P$, we let the edge between $Q \setminus \{x\}$ and Q be directed towards $Q \setminus \{x\}$ iff $x \in \beta(Q \setminus \{x\})$. This implies that if S is a sink in this orientation, then (i) $P \subseteq \beta(S)$ and (ii) $x \notin \beta(S \setminus \{x\})$ for all $x \in S$. In other words, $\beta(S)$ satisfies all constraints in P, and releasing any point in S from the boundary will lead to a violation of such a constraint. It can be shown that S is a sink of the cube iff $\beta(S)$ is the smallest enclosing ball of P (although that does not follow from our discussion here). Moreover, it can be shown that the orientation we defined is a unique sink orientation. Details, together with the fact that such an orientation can be cyclic in a sufficiently high dimension, are described in [7].

How do our findings relate to that problem? In the unit cost model[3] no deterministic bounds other than the trivial one of $O(2^n \mathrm{poly}(n))$ was known for computing the smallest ball of n points in $(n-1)$-space. Since the necessary vertex evaluations can be performed in polynomial time, we have obtained an $O(1.61^n \mathrm{poly}(n))$ algorithm for that problem. Our randomized bound is not relevant here, since a randomized procedures of complexity $e^{O(\sqrt{n})}$ is known [2]. They are relevant in the context of linear complementarity problems, though: For P-matrix problems nothing better than $O(2^n)$ was known [13].

Findings. To the best of our knowledge all previous algorithms searching for the sink work in a simplex-like fashion. Start at some vertex, and successively proceed to a neighbor along some outgoing edge until a sink is found. Examples are Murty's algorithm ([13], cf. [1, 16]), or the random edge rule (which chooses the next vertex uniformly at random among the eligible neighbors). Recently, Morris [12] has shown that there are unique sink orientations where the random edge rule leads to an expected number of at least $((n-1)/2)!$ steps (which exceeds the number of vertices).

In this paper we investigate the structure of unique sink orientations which leads us to algorithms that exploit the access to arbitrary vertices in the cube, as already indicated in the discussion of the simple bounds above. Typically, we start by evaluating two antipodal vertices of the cube.

Simple properties and a useful characterization of unique sink orientations are presented in Section 2. In Section 3 we show the decomposability of the problem, that allows us to conclude that $t(n) \leq t(k) \cdot t(n-k)$ for all $0 \leq k \leq n$ (and analogously for $\tilde{t}$). Using simple bounds

[2]If there are $n > d+1$ points, d the dimension of the ambient space, then we embed in $(n-1)$-space and perturb. Moreover, the problem is well understood when n is large compared to d (see e.g. [8]), while the situation when n is close to d seems to be the bottleneck.

[3]As opposed to the bit-model, where polynomial time solutions are known.

for small n, this gives already bounds of the form $O(c^n)$, $c < 2$, for arbitrary n. An alternative algorithm is described in Section 4 which evaluates the sink of an n-cube in a number of steps that is at most the Fibonacci number F_{n+2}. Tailored optimal algorithms for the 2-cube (randomized) and the 4-cube (deterministic) are presented in Sections 5 and 6; they allow further improvements for n-cubes.

2 Properties

Let ψ be an orientation of a cube C, and let $R \subseteq \mathrm{carr}C$. By $\psi^{(R)}$ we denote the orientation obtained from ψ by switching the directions of all edges labeled by some $a \in R$. For $a \in \mathrm{carr}C$, we use $\psi^{(a)}$ short for $\psi^{(\{a\})}$.

Lemma 2.1 *If ψ is the unique sink orientation of a cube and $R \subseteq \mathrm{carr}C$, then $\psi^{(R)}$ is a unique sink orientation.*

Proof. It suffices to show that $\psi^{(a)}$ is a unique sink orientation for any $a \in \mathrm{carr}C$. Now consider the two antipodal facets F and G of C that are connected by edges labeled by a. Let u be the sink of F and let v be the sink of G. Exactly one of the two, say u, is the sink of the whole cube and thus has the incident edge labeled a ingoing; therefore, v has the incident edge labeled a outgoing. Clearly, reorientation of all a-labeled edges will make v the unique sink. The same argument applies to all faces of C (if a is not in the carrier of a face, nothing changes). □

An orientation ψ of a cube C induces the *out-map, $\mathbf{s}_\psi$, of ψ,* defined by

$$\begin{array}{rcl} \mathbf{s}_\psi : \ \mathrm{vert}C & \to & 2^{\mathrm{carr}C} \\ v & \mapsto & \left\{ \begin{array}{l} \{\lambda(e) \mid e \text{ is outgoing} \\ \quad \text{edge of } v \text{ w.r.t. } \psi\} \end{array} \right. . \end{array}$$

The out-map is just an alternative way of specifying an orientation. u is a sink w.r.t. ψ iff $\mathbf{s}_\psi(u) = \emptyset$.

Lemma 2.2 *Let ψ be a unique sink orientation. Then $\mathbf{s}_\psi$ is injective.*

Proof. Suppose $\mathbf{s}_\psi(u) = \mathbf{s}_\psi(v) = R$, then both u and v are sinks in the orientation $\psi^{(R)}$. Since $\psi^{(R)}$ is a unique sink orientation, $u = v$. □

Since $|\mathrm{vert}C| = |2^{\mathrm{carr}C}|$ the out-map must be a bijection for a unique sink orientation. That is, for every $K \subseteq \mathrm{carr}C$ there is unique vertex u with $\mathbf{s}_\psi(u) = K$. Hence, the unique sink property implies the analogous unique source property, etc. Out-maps of unique sink orientations have a simple characterization, which we state without proof.

Lemma 2.3 *A mapping s : $\mathrm{vert}C \to 2^{\mathrm{carr}C}$ is the out-map of a unique sink orientation of C iff $(\mathbf{s}(u) \oplus \mathbf{s}(v)) \cap (u \oplus v) \neq \emptyset$ for all $u, v \in \mathrm{vert}C$ with $u \neq v$.*

It is perhaps worthwhile to mention that injectivity alone is not sufficient for an out-map to come from a unique sink orientation. This is true, though, for acyclic orientations.

3 Inherited Orientations

Consider a cube $C = C^{[X,Y]}$ and let $A \subseteq \mathrm{carr}C$. Removal of all edges with labels from $\mathrm{carr}C \setminus A$ leaves connected components that are exactly the faces of C with carrier A. For every vertex v with $X \subseteq v \subseteq Y \setminus A$ there is a unique such face F_v with carrier A containing v. Now let ψ be an orientation of C such that every face with carrier A has a unique sink. Then we define a mapping

$$\begin{array}{rcl} \mathbf{s}_{\psi/A} : [X, Y \setminus A] & \to & 2^{\mathrm{carr}C \setminus A} \\ v & \mapsto & \left\{ \begin{array}{l} \text{set of labels of edges out-} \\ \text{going of the sink of } F_v \end{array} \right. \end{array}$$

which we call the *A-inherited out-map of ψ.* Note that $\mathbf{s}_{\psi/A}$: $\mathrm{vert}C' \to 2^{\mathrm{carr}C'}$ for $C' = C^{[X,Y\setminus A]}$, but, a priori, it is not clear that this is an out-map of any orientation of C'.

Lemma 3.1 *Let ψ be a unique sink orientation of a cube C, and let $A \subseteq \mathrm{carr}C$. Then $\mathbf{s}_{\psi/A}$ is the out-map of a unique sink orientation of a $(\dim C - |A|)$-cube.*

Proof. This can be derived from the characterization given in Lemma 2.3. □

Figure 3 shows an example of an orientation obtained through an inherited out-map. There we have partitioned a 4-cube into 2-faces, and these 2-faces are connected isomorphic to a 2-cube, with an inherited orientation. How can we use that for finding the sink of the 4-cube? We can perform a search on the inherited structure, an orientation of a 2-cube: That can be done with at most 3 evaluations. Note that every such vertex evaluation in the inherited structure amounts to evaluation the sink of a 2-face, which

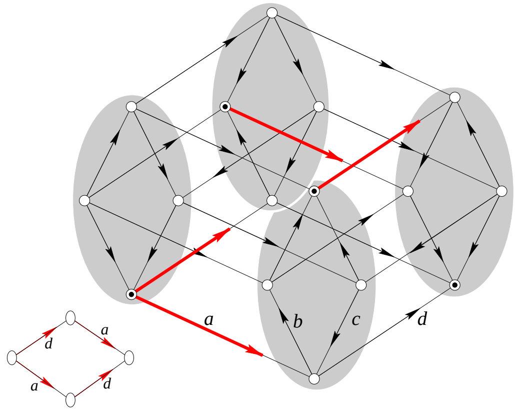

Figure 3: Orientation of a 4-cube and an inherited orientation.

takes at most 3 evaluations each. That makes at most $3 \times 3 = 9$ altogether. For the 4-cube that is not an improvement over the bound of 2^3+1 for $t(4)$, which we know already. The general principle is quite useful, though.

Lemma 3.2 *Let $0 \leq k \leq n$. Then*

$$t(n) \leq t(k) \cdot t(n-k) \quad \textit{and} \quad \tilde{t}(n) \leq \tilde{t}(k) \cdot \tilde{t}(n-k) \ .$$

Proof. (*Product Algorithm*) Given an orientation ψ of an n-cube C, choose a subset A of carr$C$ of $n-k$ elements. Now perform a search for the sink in the cube induced by the inherited orientation $\mathbf{s}_{\psi/A}$; this takes at most $t(k)$ evaluations. Each evaluation, however, amounts to finding (and evaluating!)[4] a sink of an $(n-k)$-dimensional face, and thus takes at most $t(n-k)$ steps.

Analogously for $\tilde{t}$. ◻

Corollary 3.3 *Let $0 \leq k \leq n$. Then*

$$t(n) \leq t(k)^{\lceil n/k \rceil} \quad \textit{and} \quad \tilde{t}(n) \leq \tilde{t}(k)^{\lceil n/k \rceil} \ . \ ◻$$

Recall that we have already derived $t(3) \leq 2^2+1 = 5$ which now gives $t(n) = O(\sqrt[3]{5}^{\,n}) = O(1.71^n)$. Similarly, $\tilde{t}(1) \leq 3/2$ entails $\tilde{t}(n) = O(1.5^n)$. In the rest of the paper we will further improve on this bounds.

4 The Fibonacci Seesaw

Let ψ be a unique sink orientation of a cube C. The Fibonacci Seesaw procedure maintains the following invariant for i, while increasing i from 0 to $n-1$: There are two antipodal i-faces F and G of C, with theirs sinks u and v, resp., already evaluated. This invariant can be obtained for $i=0$ by 2 vertex evaluations. If we have reached $i = n-1$, then F and G are antipodal facets and we are done, since either u or v is the sink of the whole cube C.

In order to proceed from i to $i+1$, we choose some $b \in \mathbf{s}_\psi(u) \oplus \mathbf{s}_\psi(v)$, which must exist because of the injectivity of $\mathbf{s}_\psi$. Assume that $b \in \mathbf{s}_\psi(v)$. Then $b \notin \mathbf{s}_\psi(u)$ and we can extend F along b to an $(i+1)$-face F' of which u is the sink. The face G' antipodal to F' in C is an $(i+1)$-cube, of which G is a facet. We have to search for the sink of G' in the facet of G' antipodal to G, that is, we have to search and evaluate the sink in an i-cube which takes at most $t(i)$ evaluations. Summing up, we can extend the invariant from i to $i+1$ with $t(i)$ vertex evaluations. This leads to the recursion

$$\begin{aligned} t(0) &= 1, \quad \text{and} \\ t(n) &\leq 2 + t(0) + t(1) + \cdots t(n-2), \\ & \qquad \text{for } n \geq 1. \end{aligned}$$

Thus $t(n) \leq F_{n+2} = O(1.62^n)$, for the Fibonacci numbers defined by $F_0 = 0$, $F_1 = 1$, and, for $k \geq 0$, $F_{k+2} = F_{k+1} + F_k$.

In particular, $t(4) \leq 8$. One can further improve the Fibonacci Seesaw by making use of an improved algorithm for 4-cubes. The SevenStepsToHeaven Algorithm, described in Section 6, finds the sink in a 4-cube with at most 7 evaluations.

Theorem 4.1 $t(n) \leq f(n)$, *where f is defined by the recurrence relation $f(n) = 2 + \sum_{i=0}^{n-5} f(i) + 5\,f(n-4)$, for $n \geq 4$, and initial conditions $f(0)=1, f(1)=2, f(2)=3, f(3)=5$. The solution to this recurrence relation yields $f(n) = O(1.61^n) = o(F_{n+2})$.*

Proof. (*ImprovedSeesaw Algorithm*) For n-cubes, $0 \leq n \leq 3$, we proceed in the same way as the Fibonacci Seesaw. For $n \geq 4$, we start out as in the Fibonacci Seesaw, but switch strategy when two antipodal $(n-4)$-faces have their sinks evaluated. Up to this point the algorithm has performed at most $2 + \sum_{i=0}^{n-5} t'(i)$ vertex evaluations, where $t'(i)$ is the maximum number of vertex evaluations required by our algorithm on an i-cube.

Let F and G be the two antipodal $(n-4)$-faces of which we have already evaluated the sink. Set

[4] Here our careful definition of $t(n)$ pays off.

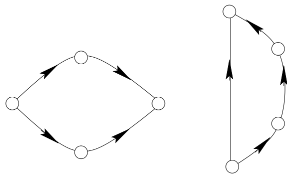

Figure 4: The two types of unique sink orientations of the 2-cube: Eye and bow.

$A := \mathrm{carr} F = \mathrm{carr} G$. Now we invoke Algorithm SevenStepsToHeaven on the inherited orientation induced by $\mathbf{s}_{\psi/A}$. This is an orientation of a 4-cube, of which we have already evaluated two antipodal vertices. To conclude our algorithm, we need to evaluate 5 more appropriately chosen $(n-4)$-faces. Hence, $t'(0) = 1$, $t'(1) = 2$, $t'(2) = 3$, $t'(3) = 5$, and, for $n \geq 4$,

$$t'(n) \leq 2 + \sum_{i=0}^{n-5} t'(i) + 5\, t'(n-4)$$

and thus $t'(n) \leq f(n)$ with $f(n)$ statisfying the recurrence relation given in the assertion of the theorem.

In order to solve the recurrence, observe that for every $n \geq 5$,

$$f(n) - f(n-1) = 5f(n-4) - 4f(n-5).$$

So $f(n)$ satisfies the recursion $f(n) = f(n-1) + 5f(n-4) - 4f(n-5)$ for every $n \geq 5$, with initial conditions $f(0) = 1, f(1) = 2, f(2) = 3, f(3) = 5, f(4) = 7$. The solution of this recurrence relation is of the form $f(n) = \sum_{i=1}^{5} \alpha_i r_i^n$, where the r_i, $i = 1, \ldots 5$, are the distinct roots of the polynomial $x^5 - x^4 - 5x + 4$, and the $\alpha_i \neq 0$ are constants [14, Section 5.2]. Thus $f(n)$ is dominated by the term r^n, where r is the root of largest absolute value. In our case this value is $1.60588\ldots$. ◻

5 2-Cube (Randomized)

There are two types of unique sink orientations of 2-cubes (see Figure 4): One, where source and sink are connected by two disjoint paths of length 2; we call it an *eye*. And one, where source and sink are connected by a path of length 1 and a path of length 3; we call it a *bow*.

Recall that $\tilde{t}(2) \leq \tilde{t}(1)^2 = 9/4$, a bound on which we will improve now.

Lemma 5.1 *In a 2-cube the sink of a unique sink orientation can be evaluated with an expected number of $\frac{43}{20}$ vertex evaluations.*

Proof. We describe three randomized strategies that will perform either better on an eye or on a bow. The right randomized mixture will lead to the claimed expectation.

We start with the common features of all three strategies. The analysis for a bow or an eye is given along with the description.

First, choose a vertex v, uniformly at random.

Case 1 Evaluation shows that v is the sink. We are done. The case happens with probability $\frac{1}{4}$ and requires 1 evaluation (both for eye and bow). *(Case 1: eye* $1/4$*; bow* $1/4$*)*

Case 2 Evaluation reveals one incoming and one outgoing edge of v, a case that happens with probability $\frac{1}{2}$ both for eye and bow. We evaluate the head u of the outgoing edge next.

If we have an eye, then u must be the sink. That is, for an eye, this case entails 2 evaluations.

If we have a bow, then either u is the sink, or if it has an outgoing edge, the sink has to be the head w of this edge. Conditioned on Case 2 occurring, both outcomes happen with equal probability for a bow. Hence, for a bow, this case requires an expected number of $\frac{2+3}{2}$ evaluations. *(Case 2: eye* 1*; bow* $5/4$*)*

Case 3 We are left with the situation that v is the source. This case has a probability of $\frac{1}{4}$ to occur. Let w be the vertex antipodal to v, and let u' and u'' be the two remaining vertices. Here is where the three strategies discriminate their further proceeding. Note that in case of an eye, the antipodal vertex w is the sink. In case of a bow, one of the two neighbors u' or u'' is the object of desire.

Strategy 1 We evaluate the vertex antipodal to v. For an eye, we are done and we used 2 evaluations altogether. For a bow, one of the two remaining vertices is the sink, but since we know now the orientations of all edges, we know the sink and evaluate it. 3 evaluations were necessary. *(Case 3, Strategy 1: eye* $1/2$*; bow* $3/4$*)* Therefore, summing up the contributions from all three cases, we get the following expectations. *(Overall, Strategy 1: eye* $7/4$*; bow* $9/4$*)*

Strategy 2 For the next vertex to evaluate we sample uniformly at random in $\{u', u''\}$. If this is not the sink, we evaluate the other vertex in $\{u', u''\}$ next. For an eye, of course, that's bad news. Neither u' nor u'' is the sink, and we end up evaluating 4 vertices. For a bow, either u' or u'' is the sink. That is, with equal probability we will succeed on the first one or we have to proceed to the other one as well. So for a bow, $\frac{2+3}{2}$ vertex evaluations had to be performed on the average.

(Case 3, Strategy 2: eye 1*; bow* 5/8*)*

Consequently, expectations can be summarized.

(Overall, Strategy 2: eye 9/4*; bow* 17/8*)*

Strategy 3 Like before, the next vertex for evaluation is chosen uniformly at random in $\{u', u''\}$. However, if that is not the sink, we continue with the antipodal vertex w. This causes 3 evaluations for an eye. The bow has 2 and 4 evaluations with equal probability, thus also 3 on the average.

(Case 3, Strategy 3: eye 3/4*; bow* 3/4*)*

We see that this strategy does not improve on Strategy 1 in both configurations, and thus it is excluded from further consideration.

What we will do, though, is to toss a biased coin and pursue Strategy 1 or 2 depending on the outcome of the experiment. If the coin lets Strategy 1 materialize with probability λ, this mixed strategy gives expectations of

$$\begin{array}{ll} \text{eye} & \frac{7}{4}\lambda + \frac{9}{4}(1-\lambda) = \frac{9}{4} - \frac{1}{2}\lambda \\ \text{bow} & \frac{9}{4}\lambda + \frac{17}{8}(1-\lambda) = \frac{17}{8} + \frac{1}{8}\lambda \end{array}$$

The maximum of these two expressions is minimized for $\lambda = \frac{1}{5}$, when it is $\frac{43}{20}$, the value claimed in the lemma. ◻

A careful inspection of the proof shows that the bound obtained is optimal for the 2-cube (that is why have included consideration of Strategy 3 which was later abandoned; details omitted). With Lemma 3.2, we have obtained a bound of $O(\sqrt{43/20}^{\,n}) = O(1.467^n)$ for the expected number of vertex evaluations.

Günter Rote [15] has determined the optimal randomized algorithm for the 3-cube by solving a linear program. It amounts to an algorithm with an expected number of $\frac{4074633}{1369468} \approx 2.976$ vertex evaluations; with Lemma 3.2, this gives a bound of $O(1.438^n)$.

6 4-Cube (Deterministic)

Next we improve on the Fibonacci Seesaw for 4-cubes. Given a cube C, a vertex u, and $A \subseteq \text{carr}C$, we let $C(u, A)$ denote the face of C with carrier A containing u.

Lemma 6.1 *In a 4-cube the sink of a unique sink orientation ψ can be evaluated with at most 7 vertex evaluations.*

Proof. (*SevenStepsToHeaven Algorithm*) We start by evaluating a pair of antipodal vertices u_1 and u_2. Since the out-map is injective, there exists a label $p \in \mathbf{s}_\psi(u_1) \oplus \mathbf{s}_\psi(u_2)$, say the incident edge of label p is outgoing for u_1, and incoming for u_2. If there is another incoming edge (with label $q \neq p$) for u_2 (see Figure 5), then we run the Product Algorithm on the orientation induced by $\mathbf{s}_{\psi/\{p,q\}}$. The evaluation of the sink u_2 of $C(u_2, \{p, q\})$ took only one step (as opposed to the usual three we have to account for in a 2-cube). The evaluation of u_1 was not wasted, as it is in the antipodal 2-face. So we gained two steps compared to the regular $3\times 3 = 9$ of the Product Algorithm, and finished in at most $1 + 3 + 3 = 7$ steps.

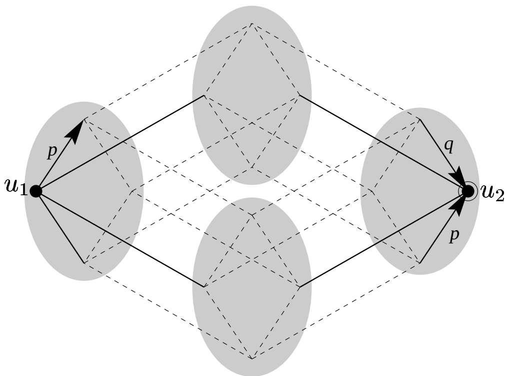

Figure 5:

Thus we can assume that $\mathbf{s}_\psi(u_2) = \text{carr}C \setminus \{p\}$ (Figure 6).

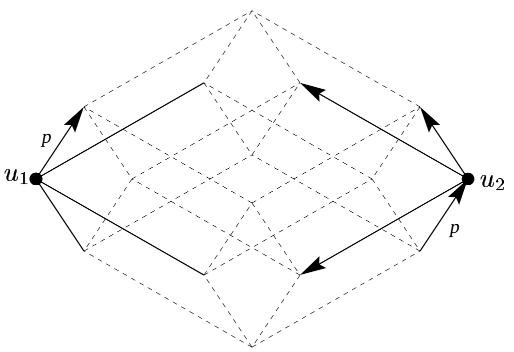

Figure 6:

Let us evaluate vertex $u_3 = u_1 \oplus \{p\}$ next. Since $\mathbf{s}_\psi(u_3) \neq \mathbf{s}_\psi(u_2)$, u_3 has another incoming edge labeled $q \neq p$. If there are at least three incoming edges for u_3 (see Figure 7), we are done: In that case u_3 is the sink of the 3-face spanned by these incoming edges, where the sink has been evaluated in 2 steps. The vertex

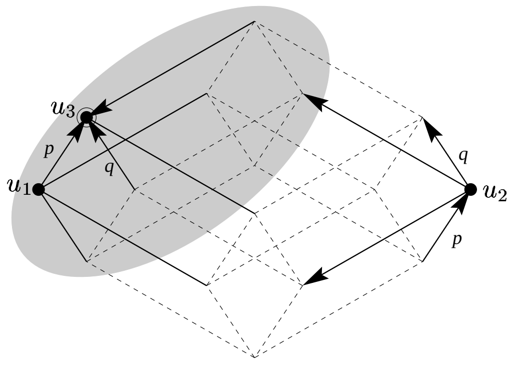

Figure 7:

u_2 is in the antipodal 3-face, where we can evaluate the sink in at most 4 further steps. Then we have sinks in two antipodal facets of C, one is the sink of the whole cube. We have used at most $2 + 5 = 7$ evaluations.

Hence, we can assume $\mathbf{s}_\psi(u_3) = \mathrm{carr} C \setminus \{p, q\}$ (Figure 8).

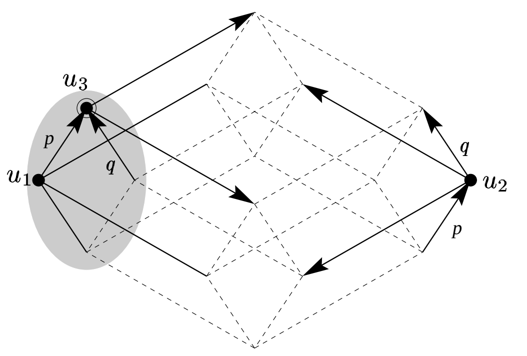

Figure 8:

Let us evaluate vertex $u_4 = u_2 \oplus \{q\}$ next. If $p \notin \mathbf{s}_\psi(u_4)$ (see Figure 9), then we are done by running the Product Algorithm on the orientation induced by $\mathbf{s}_{\psi/_{\{p,q\}}}$. We evaluated the sink u_3 of $C(u_1, \{p, q\})$ in just two steps[5], and the sink u_4 of the antipodal 2-face $C(u_2, \{p, q\})$ in two steps as well[6]. Thus at most 3 more evalu-

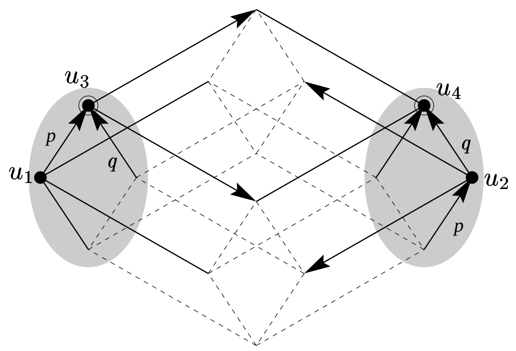

Figure 9:

ations in the appropriate 2-face will give us the global sink in at most $2 + 2 + 3 = 7$ steps altogether.

We can thus assume $p \in \mathbf{s}_\psi(u_4)$ (Figure 10). So far our algorithm went as the Fibonacci See-

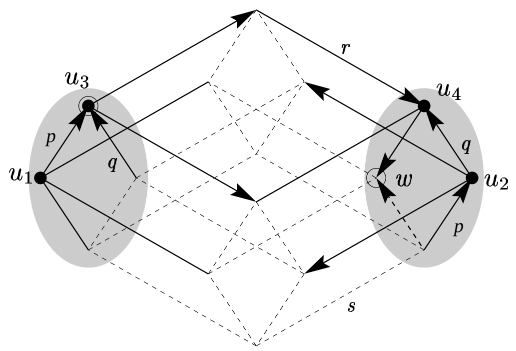

Figure 10: The edge $\{w, w \oplus \{q\}\}$ has to be directed towards w, since otherwise $C(w, \{p, q\})$ has no sink.

saw would go. The Fibonacci Seesaw now would evaluate the known sink $w = u_4 \oplus \{p\}$ of the 2-face $C(u_2, \{p, q\})$. Then it would go on and evaluate the sink of the appropriate neighboring 2-face spanned by edges labeled p and q. This evaluation of w is not very economical, as it reveals only the orientation of *two* new edges, instead of *three or four*. Here we avoid the evaluation of w by evaluating its neighbors in an appropriate order.

u_4 is the antipodal vertex to u_3 in the 2-face generated by directions $\mathrm{carr} C \setminus \{p, q\}$. u_3 is the source of this 2-face, so u_4 must have at least

[5]A birdie!

[6]Another birdie!

one incoming edge of direction $r \notin \{p, q\}$. Let s be the fourth direction, not equal to p, q, r (Figure 10).

We evaluate $u_5 = w \oplus \{s\}$. If the edge labeled s is incoming to u_5 (see Figure 11), then we know that the global sink of C is in the 2-face $C(u_5, \{p, q\})$. This sink can be found with at most two more evaluations, which gives us at most 7 altogether.

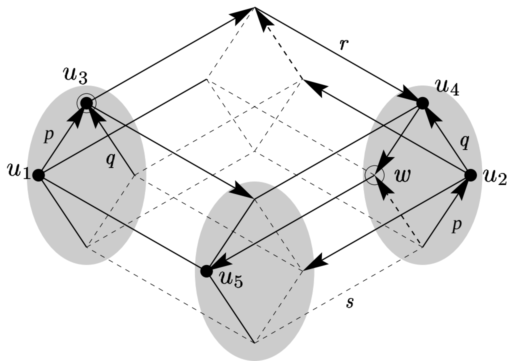

Figure 11:

If the edge labeled s is outgoing from u_5 (see Figure 12), then we know that the global sink is either w or it is the sink of the 2-face $C(w \oplus \{r\}, \{p, q\})$.

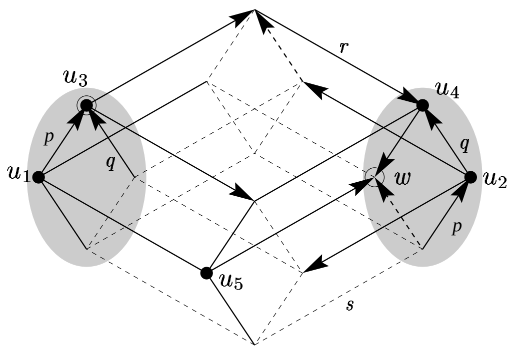

Figure 12:

Let us evaluate $u_6 = w \oplus \{r\}$, thus revealing the last unknown incident edge to w. After this evaluation we should know where the global sink is. If the edge of direction r is outgoing from u_6, then the global sink is w (Figure 13).

If $r \notin \mathbf{s}_\psi(u_6)$ (see Figure 14), then the global sink is the sink of $C(u_6, \{p, q\})$. By observing the three known edges of the 2-face $C(u_2, \{q, r\})$, we can conclude, that the edge

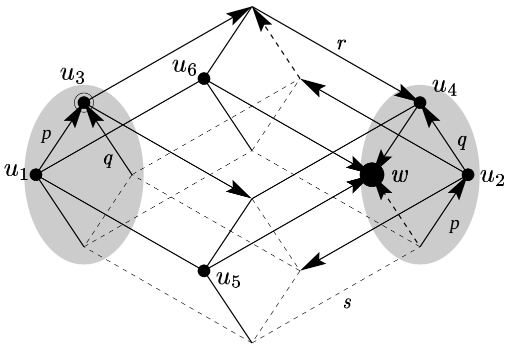

Figure 13:

$\{u_2 \oplus \{r\}, u_4 \oplus \{r\}\}$ is directed towards $u_4 \oplus \{r\}$. This, together with the knowledge of the orien-

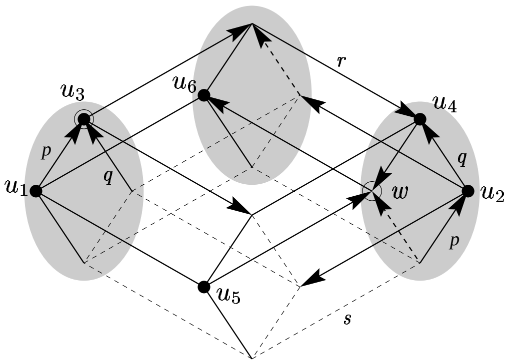

Figure 14: The edge $\{u_2 \oplus \{r\}, u_4 \oplus \{r\}\}$ must be directed towards $u_4 \oplus \{r\}$, since otherwise $C(u_2, \{q, r\})$ has two sinks.

tations of the edges incident to u_6 tells us where the sink of $C(u_6, \{p, q\})$ is, which we can evaluate in step 7. ◻

It can be shown that the bound is tight, i.e. $t(4) = 7$.

References

[1] Richard W. Cottle, Jong-Shi Pang, Richrad E. Stone: *The Linear Complementary Problem*, Academic Press (1992).

[2] Bernd Gärtner: A subexponential algorithm for abstract optimization problems, SIAM J. Comput. **24** (1995), 1018–1035.

[3] —: Combinatorial linear programming: Geometry can help, *in* "Proc. 2nd Int. Workshop

on Randomization and Approximation Techniques in Computer Science," Lecture Notes in Computer Science **1518** (1998), 82–96.

[4] —: The random-facet simplex algorithm on combinatorial cubes, submitted (2001).

[5] Bernd Gärtner, Martin Henk, Günter M. Ziegler: Randomized simplex algorithms on Klee-Minty cubes, Combinatorica **18** (1998), 349–372.

[6] Bernd Gärtner, Volker Kaibel: Abstract objective function graphs on the 3-cube: A classification by realizability, Tech. Rep 296, Department of Computer Science, ETH Zurich (1998).

[7] Bernd Gärtner, Hiroyuki Miyazawa, Lutz Kettner, Emo Welzl: ¿From geometric optimization problems to unique sink orientations of cubes, manuscript in preparation (2001).

[8] Bernd Gärtner, Emo Welzl: Linear Programming – Randomization and abstract frameworks, Proc. 13th Ann. ACM Symp. Theoretical Aspects of Computer Science (1996), 669–687.

[9] —: Explicit and Implicit Enforcing – Randomized Optimization, *in* Lectures of the Graduate Program Computational Discrete Mathematics, Lecture Notes in Computer Science **2122** (2001), 26-49.

[10] Volker Kaibel: personal communication.

[11] Gil Kalai: Linear programming, the simplex algorithm and simple polytopes, Math. Programming **79** (1997), 217–233.

[12] Walter D. Morris: Randomized principal pivot algorithms for P-matrix linear complementarity problems, Math. Programming, to appear (2001).

[13] K. G. Murty: Note on Bard-type scheme for solving the complementarity problem, Operations Research **11** (1974), 123-130.

[14] Kenneth H. Rosen: *Discrete Mathematics and its Applications,* McGraw-Hill (1999).

[15] Günter Rote: personal communication.

[16] Alan Stickney, Layne Watson: Digraph models of Bard-type algorithms for the linear complementary problem, Mathematics of Operations Research **3** (1978), 322–333.

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