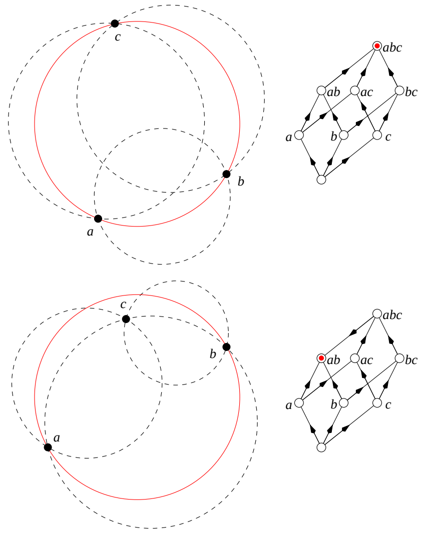
Unique Sink Orientations of Cubes

Why
Properties
How to find the Sink



Emo Welzl
ETH Zürich

Linear Programming

Given $A \in \mathbf{R}^{n\times d}$, $b \in \mathbf{R}^n$, $c \in \mathbf{R}^d$, solve

$$\max_{x \in \mathbf{R}^d} cx \quad \text{subject to} \quad Ax \geq b\ .$$

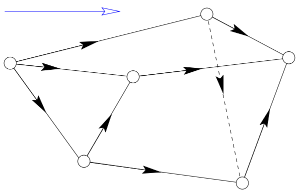
Polynomial time solvable (in n, d, and coding length L) [Khachian, Karmarkar]

Is there a polynomial (in n and d) algorithm over the real numbers?

deterministic $d^{O(d)}n$, randomized $d^2n + e^{O(\sqrt{d \log d})}$

Steve Smale in 'Mathematical Problems for the Next Century,' (1998): "*It is my belief that this problem number nine is the main unsolved problem of linear programming theory. The use of real number algorithms plays a natural role in a subject where the most important measure of cost is the number of arithmetic operations.*"

Abstract Objective Functions



If we orient the edges of a polytope $\mathcal{P}$ according to a generic linear function (the road map of the simplex algorithm) then

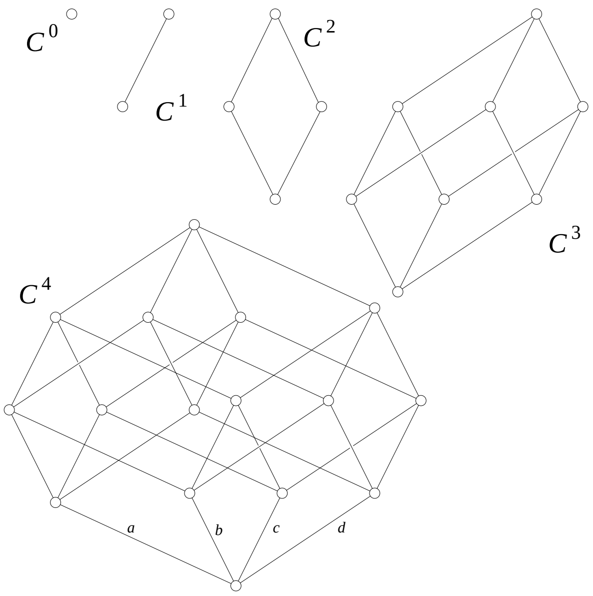
(1) every face has a unique sink, and

(2) there are no cycles.

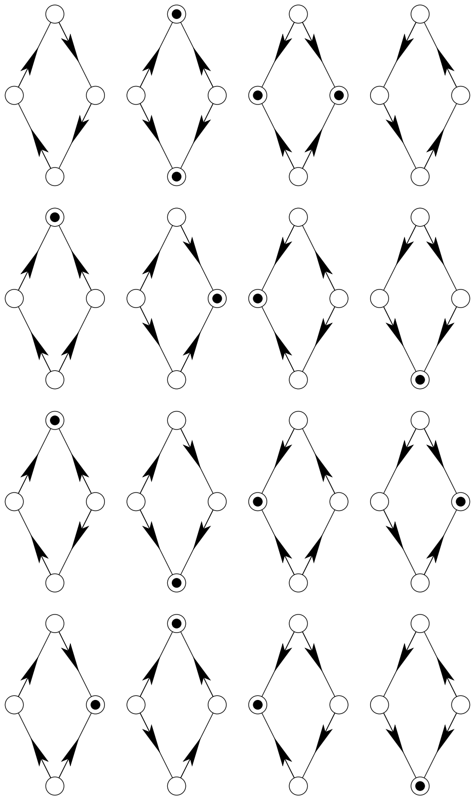
Every orientation satisfying (1) and (2) is called abstract objective function (AOF). [Kalai]

Here we restrict our attention to cubes and relax the acyclicity condition (2).

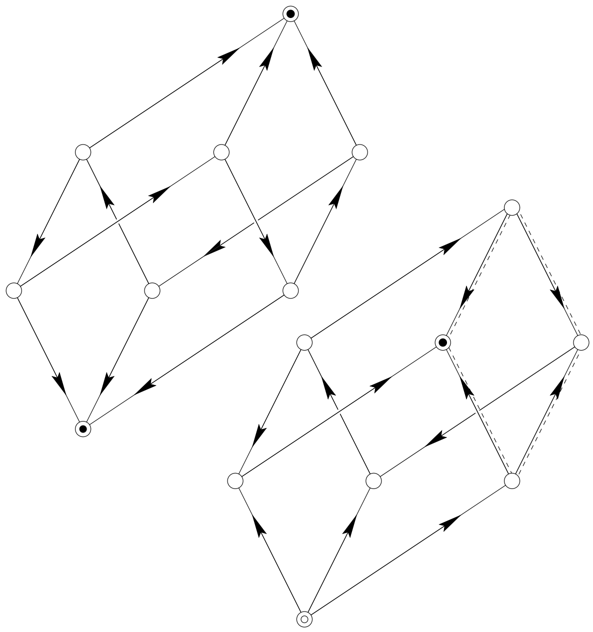
Cubes



All orientations of a 2-cube:



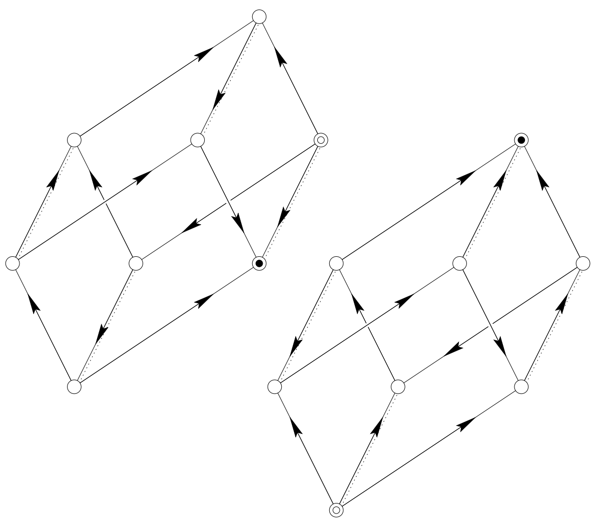
Orientations of 3-cubes:



Sink: Vertex with no outgoing edges.

Unique sink orientation (USO) of n-cube:

Every face has a *unique* sink.

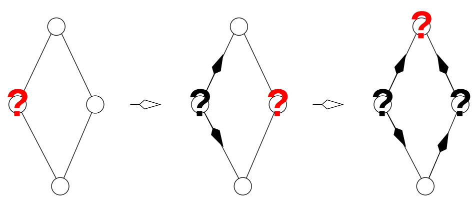


Goal: Find the sink in an implicitly given unique sink orientation, which can be accessed by vertex evaluations.

Vertex evaluation: Returns orientations of all incident edges.

How many vertices have to be evaluated in a u.s.o. of an n-cube, until we have evaluated the sink?

$t(n)$ deterministic
$\tilde{t}(n)$ expected for randomized



3 Steps until we hit the sink ($t(2) = 3$).

USO's occur as

- orientations of slanted geometric cubes relative to a generic linear function, e.g. Klee-Minty cube (acyclic),

- models for certain linear complementarity problems (can have cycles).
[Stickney, Watson'78]

- LP (over the reals/unit cost model) is polynomial time reducible to finding the sink in a USO. [Gärtner'02]

$$\mathrm{LP}(d, n) \rightsquigarrow \mathrm{USO}(n)$$

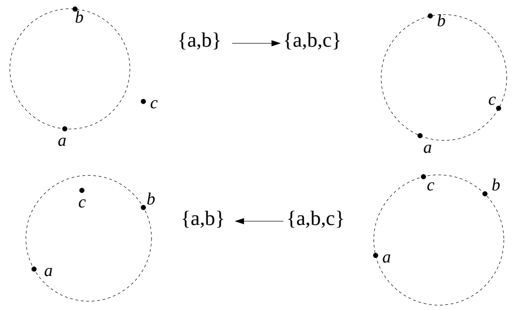
($t(n)$ plus time to compute the queries!)

Smallest enclosing ball of set P of $d+1$ generic affinely independent points in $\mathbf{R}^d$.

Consider boolean lattice on 2^P with orientation: For $P \ni x \notin Q \subseteq P$,

$$Q \longleftarrow Q \cup \{x\} \text{ iff } x \in \beta(Q) .$$

$\beta(Q)$ smallest ball with Q on boundary.

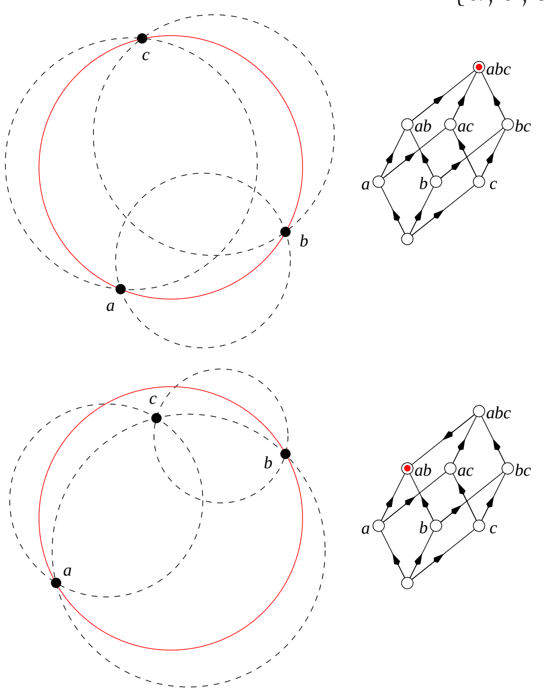


Gives USO (can have cycles) proof omitted.

If you have $n > d+1$ points, embed the ambient space in $\mathbf{R}^{n-1}$ and perturb.

S is the sink iff $\beta(S)$ is the smallest ball enclosing P.

$$P = \{a, b, c\}$$



What is known? $t(n)$??

- Relation to *linear complementarity problems* with P-matrix.
[Stickney, Watson'78]

- *Random edge* can take $\frac{(n-1)!}{2}$ steps.
[Morris'00]

- Analysis of random edge for Klee-Minty cube. [Gärtner, Henk, Ziegler'98]

- Random edge on decomposable orientations is $O(n^2)$. [Williamson Hoke'88]

- $\tilde{t}_{acyc}(n) \leq e^{2\sqrt{n}}$. [Gärtner'95]

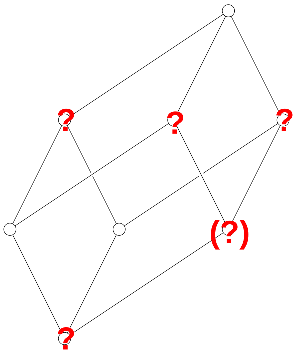
If an orientation of an n-cube is given by some function on the vertices, edges directed from larger to smaller value, such that there is a unique sink (no local minimum; acyclic, not necessarily USO), then there is a *tight* order $\sqrt{2}^n$ bound for finding the sink with a randomized algorithm [Aldous'83].

A first observation.

$$t(n) \le 2^{n-1} + 1$$

C^n is bipartite. Evaluate all vertices in one color class. These vertices cover all edges; thus we know all orientations. Perform one extra evaluation of the sink, if it lies in the other color class.

Works also if all we require is a unique sink, and it is tight for such orientations.



$$\tilde{t}(n) \le \frac{3}{4}(2^{n-1} + 1)$$

We want to show

$$t(n) \le t(k) \cdot t(n-k).$$

E.g., with $t(3) \le 2^2 + 1 = 5$, this gives

$$t(n) \le t(3)^{\lceil n/3 \rceil} = O(\sqrt[3]{5}^{\,n}) = O(1.71^n)$$

Similarly,

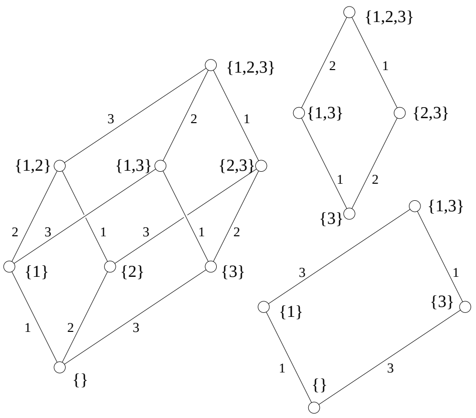
$$\tilde{t}(n) \le \tilde{t}(k) \cdot \tilde{t}(n-k)$$

Combined with $\tilde{t}(1) \le 3/2$, we get

$$\tilde{t}(n) \le \tilde{t}(1)^n = O(1.5^n)$$

A . . . a finite set.

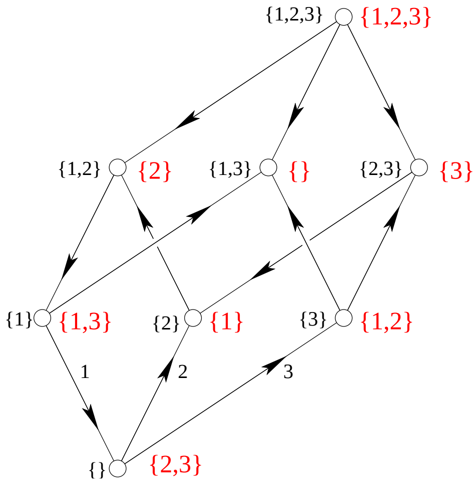
C^A is the edge labeled graph with
  vertices: 2^A ;
  edges: $\{\{u, u \oplus \{a\}\} \mid u \in 2^A, a \in A\}$;
  labeling: $\lambda(\{u, u \oplus \{a\}\}) = a$.



3-Cube $C^{\{1,2,3\}}$, and two of its 2-faces.

Out-map s_ψ of orientation ψ of C^A:

vertex $v \mapsto$ labels of outgoing edges
$2^A \to 2^A$



v is sink iff $s_\psi(v) = \emptyset$.

Characterization of out-maps s of USO:

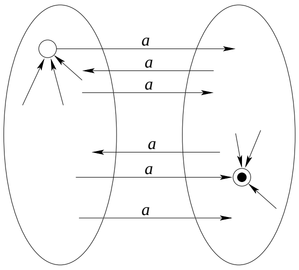
$$(s(u) \oplus s(v)) \cap (u \oplus v) \neq \emptyset$$

for all vertices $u \neq v$.

In particular, s is injective (and thus a permutation); characterizes USOs for acyclic orientations – not so in general.

Proof of injectivity for USO.
Reorientation of all edges with a given label a preserves USO

Antipodal facets separated by a-edges.



If $s(u) = s(v)$, we reorient all a-edges with $a \in s(u)$. Now u and v are sinks; ↯.

Another view:

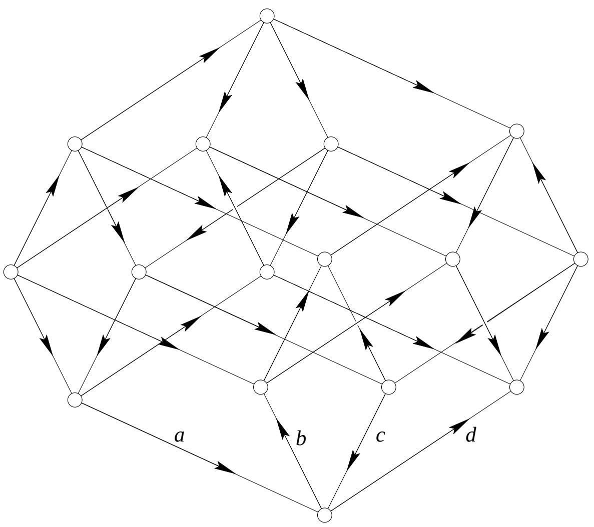
For a function

$$s : 2^A \to 2^A \qquad \text{or } \mathbf{F}_2^n \to \mathbf{F}_2^n \qquad \text{or } \{0,1\}^n \to \{0,1\}^n$$

with the property

$$(s(u) \oplus s(v)) \cap (u \oplus v) \neq \emptyset \quad \text{for } u \neq v,$$

determine $s^{-1}(\emptyset)$ via as few function evaluations as possible.

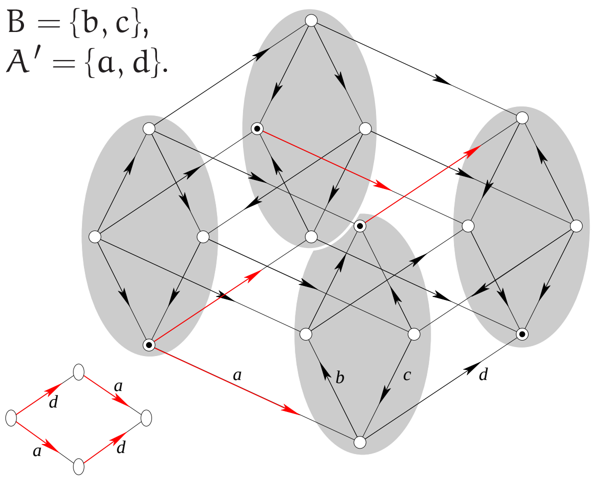


Inherited Orientation of ψ on C^A relative to $B \subseteq A$.

Consider the out map on all sinks of B-labeled faces of C^A. Gives a mapping

$$s' : \quad 2^{A'} \to 2^{A'}, \quad A' := A \setminus B\ .$$

$B = \{b, c\}$,
$A' = \{a, d\}$.



If ψ is a USO, then s' comes from a USO!

Choose B with $|B| = k$, and perform a search for the sink in the inherited $(n-k)$-cube defined by s'.

This needs $t(n-k)$ 'hyper'-vertex evaluations in the inherited cube. Each such evaluation amounts to evaluating the sink in a k-face in at most $t(k)$ steps; thus $t(k) \cdot t(n-k)$ vertex evaluations suffice.

But we can do better.

The Fibonacci Seesaw

We assume that we have two antipodal i-faces F and G with their sinks u and v, resp., already evaluated. If $i = n-1$, we are done (u or v is sink). Otherwise, choose $b \in s(u) \oplus s(v)$. Assume $b \in s(v)$.

We can extend F along b to a $(i+1)$-face F' of which u is the sink. In order to find the sink of its antipodal face G' (which has G as facet), we have to find the sink of an i-face (antipodal facet of G in G'); takes $t(i)$ evaluations.

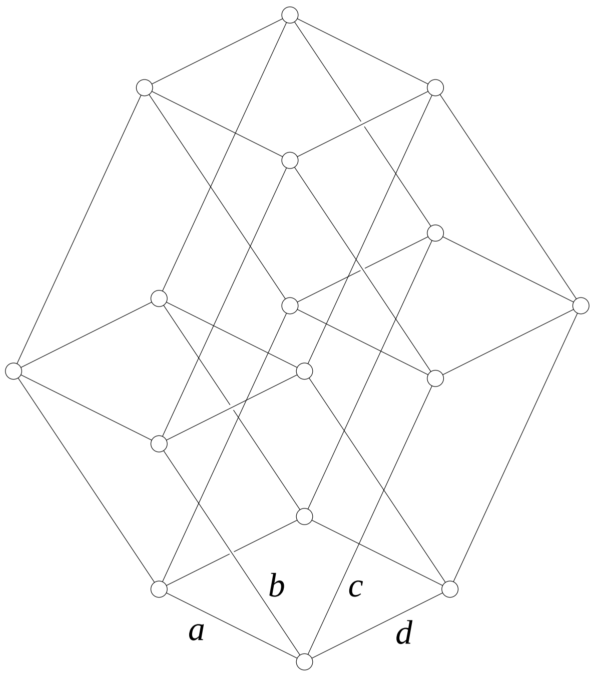
$$\xrightarrow{2} 0 \rightarrow \cdots i \xrightarrow{t(i)} (i+1) \cdots \rightarrow (n-1)$$

$$t(n) \leq 2 + t(0) + t(1) + \cdots t(n-2)$$

Thus $t(n) \leq F_{n+2} = O(1.619^n)$,

$$F_0 = 0,\ F_1 = 1,\ F_{k+2} = F_{k+1} + F_k.$$

Fibonacci Seesaw – Live



A combination of

SEESAW and SEVEN STEPS TO HEAVEN

for 4-cubes ($7 < F_6 = 8$) gives

$$t(n) = O(1.606^n)\ .$$

[Szabó,W.'01]

Yields best known *deterministic* bound for computing smallest ball enclosing $d+1$ points in $\mathbf{R}^d$, and for LPs with $2d$ constraints in d variables.

Any deterministic algorithm requires $\Omega(n^2/\log n)$ vertex evaluations in the worst case. [Schurr,Szabó'02]

- Can acyclicity be exploited for deterministic algorithms?

USO's for LP

$A \in \mathbf{R}^{n\times d}$, $b \in \mathbf{R}^n$, $c \in \mathbf{R}^d$, solve

$$\max_{x\in\mathbf{R}^d} cx \quad \text{subject to} \quad Ax \geq b\ .$$

If $n = 2d$, the feasible region has at most

$$O(\binom{\lceil 3d/2\rceil}{\lfloor d/2\rfloor}) = O(2^{H(1/3)3/2+o(1))d}) = O(2.599^d)$$

vertices, and there are instances that attain this bound.

If we apply Gärtner's reduction and the best deterministic USO strategy, then the LP can be solved in time

$$O(1.606^{2d}) = O(2.579^d)\ .$$

Best randomized: $O(e^{O(\sqrt{d})})$ expected.

Randomized algorithms.

Since $\tilde{t}(2) = 43/20$, as can be shown, $\tilde{t}(n) \leq \tilde{t}(2)^{\lceil n/2\rceil} = O(\sqrt{\tfrac{43}{20}}^{\,n}) = O(1.47^n)$.

Solving a Linear Program on a computer, G. Rote generated an optimal randomized algorithm for 3-cubes:

$$\tilde{t}(3) = \frac{4074633}{1369468} \approx 2.976\ .$$

Employing the multiplicative bound gives

$$\tilde{t}(n) = O(1.438^n)\ .$$

No lower bound known.

The number of USOs of the n-cube is $2^{\Theta((\log n)\,2^n)}$ (compared to $2^{n\,2^n}$ for all orientations) [Matoušek'02]

Problems

Improve (upper/lower) bounds ..., exploit acyclicity for better deterministic upper bound ...

Extract extra properties of USO's that come from LP or from smallest enclosing ball problem.