A parallel algorithm for computing the flow complex

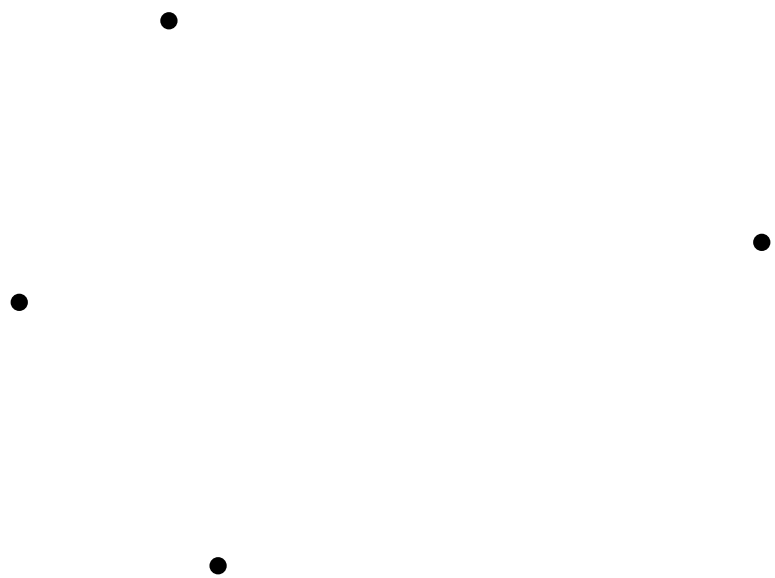
Joachim Giesen Lars Kühne

Fakultät für Mathematik und Informatik
Friedrich-Schiller Universität, Jena
Germany

joachim.giesen@uni-jena.de
lars.kuehne@uni-jena.de

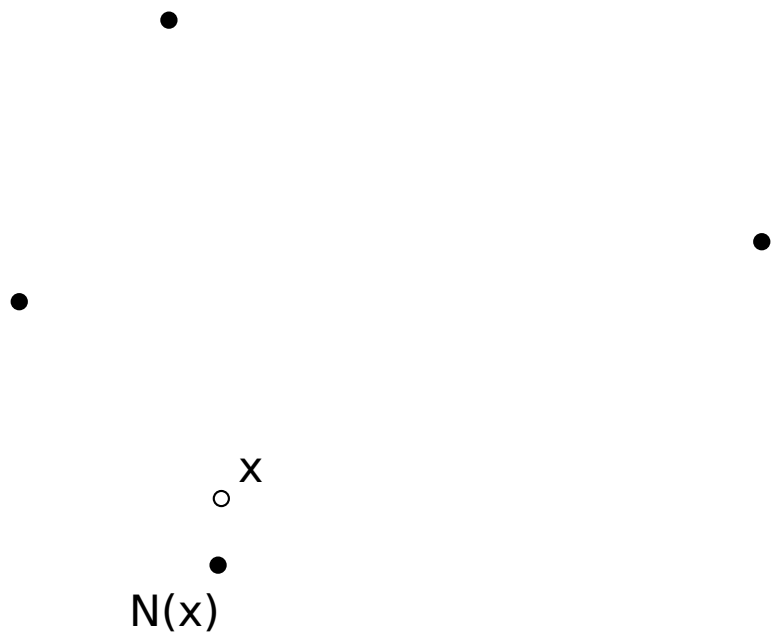
December 14, 2012

The point set



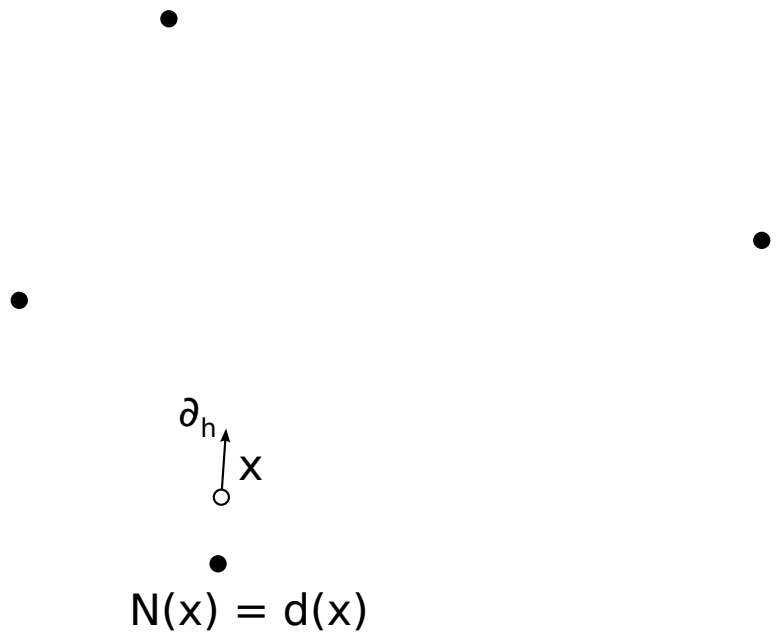
Let $P \subset \mathbb{R}^d$ be the point set in general position, with at least $d + 1$ points.

The distance function



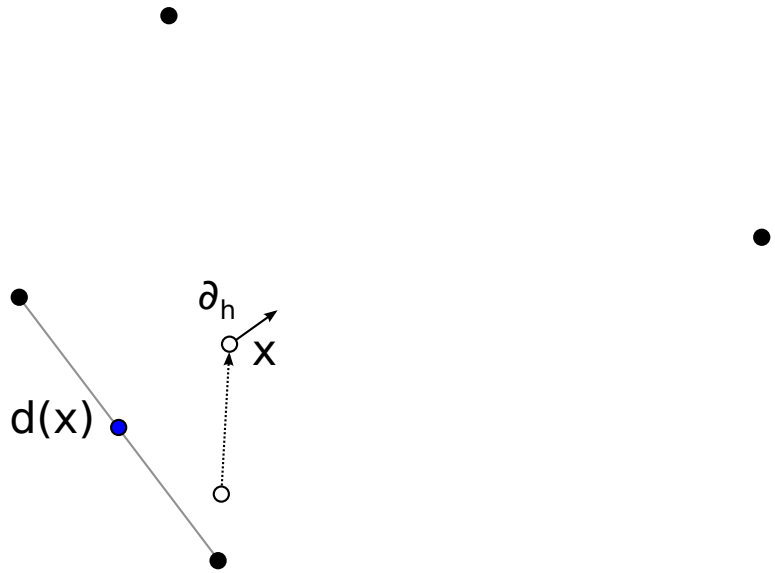
- distance function $h : \mathbb{R}^d \ni x \mapsto \min_{p \in P} \|x - p\|$
- Neighbors of x $N(x) = \{p \in P | h(x) = \|x - p\|\}$

Gradient of the distance function



- driver $d(x)$ is the center of the smallest enclosing ball of $N(x)$
- gradient of $h(x)$, $\partial_h(x) = \frac{x - d(x)}{h(x)}$

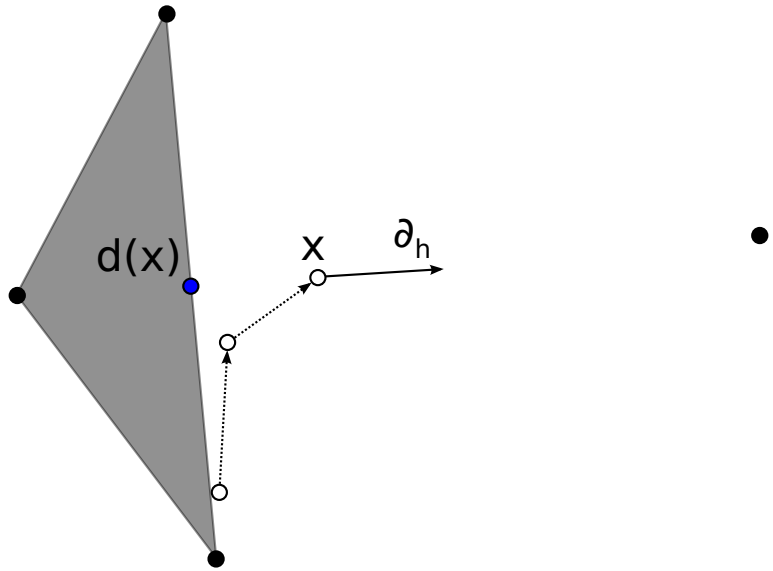
The flow



- flow $\phi : [0, \infty) \times \mathbb{R}^d \to \mathbb{R}^d$
- defined by $\phi(0, x) = x$, and

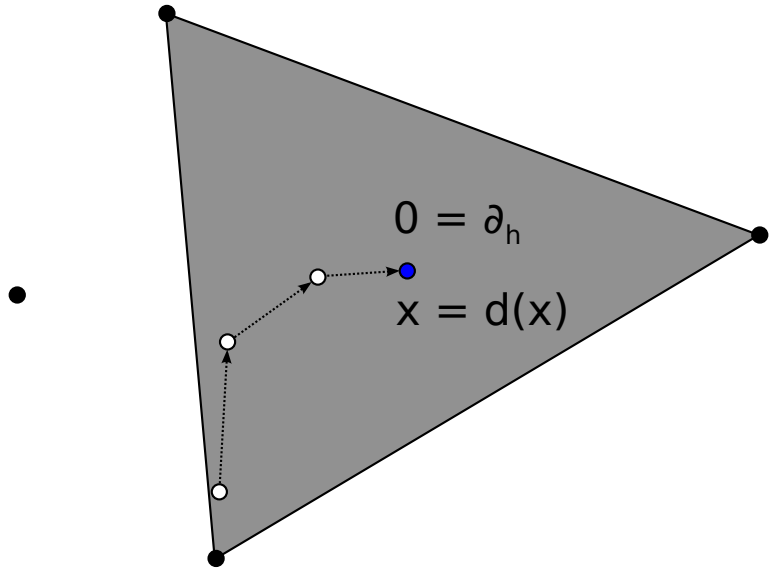
$$\lim_{t \downarrow t_0} \frac{\phi(t, x) - \phi(t_0, x)}{t - t_0} = \partial_h(\phi(t_0, x))$$

The Flow line



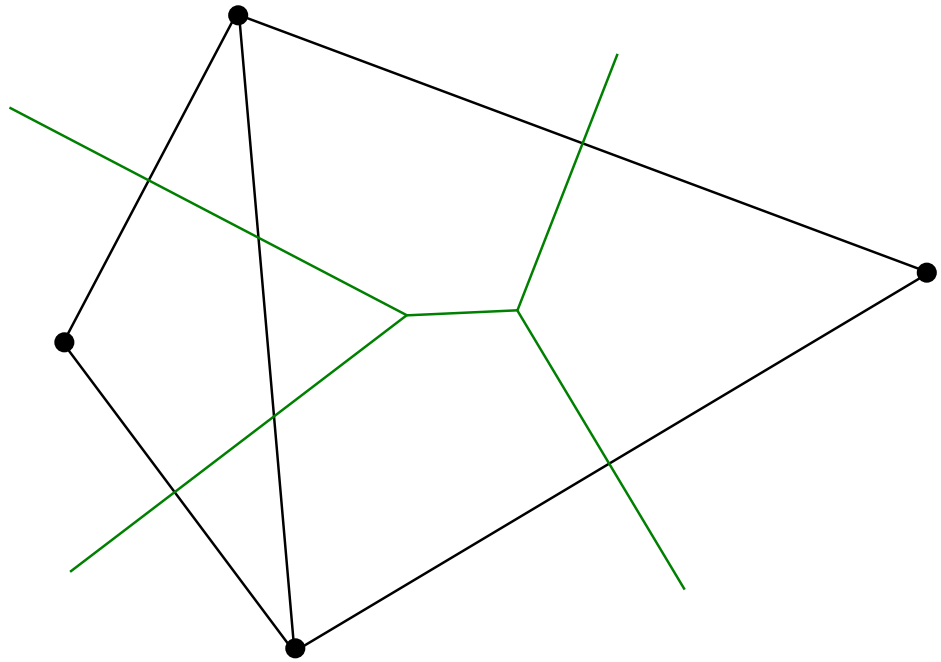
- orbit/flow-line of a point x: $\phi(x) = \{\phi(t, x) | t \geq 0\}$

Critical point

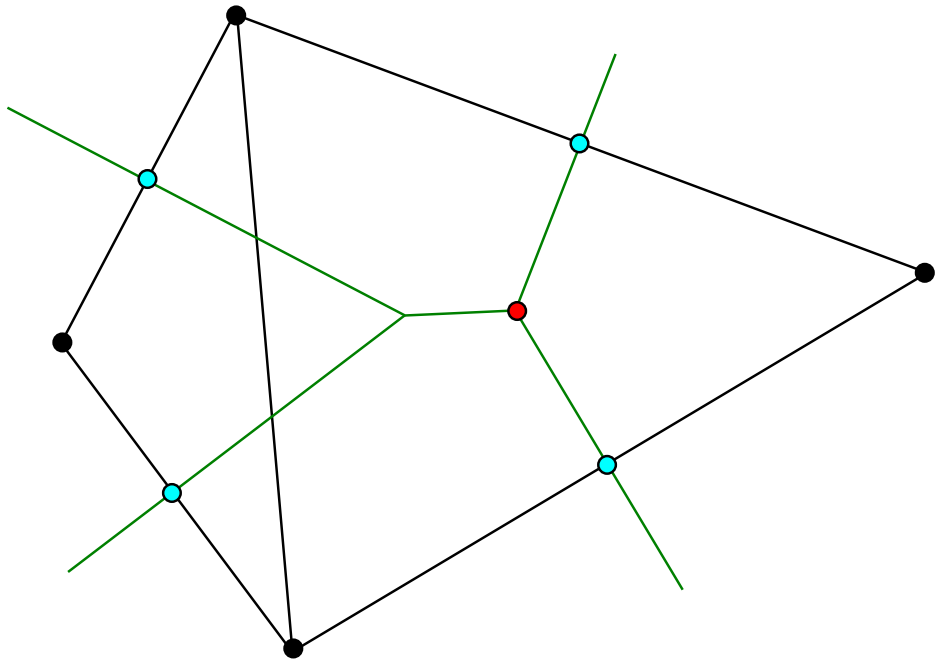


- $x \in \mathbb{R}^d$: $\partial_h(x) = 0$
- $x \in conv(N(x))$
- index of a critical point is the dimension of the affine hull of $N(x)$
- Minima with $i(x) = 0$, and Maxima with $i(x) = d$
- the remaining critical points are saddle points

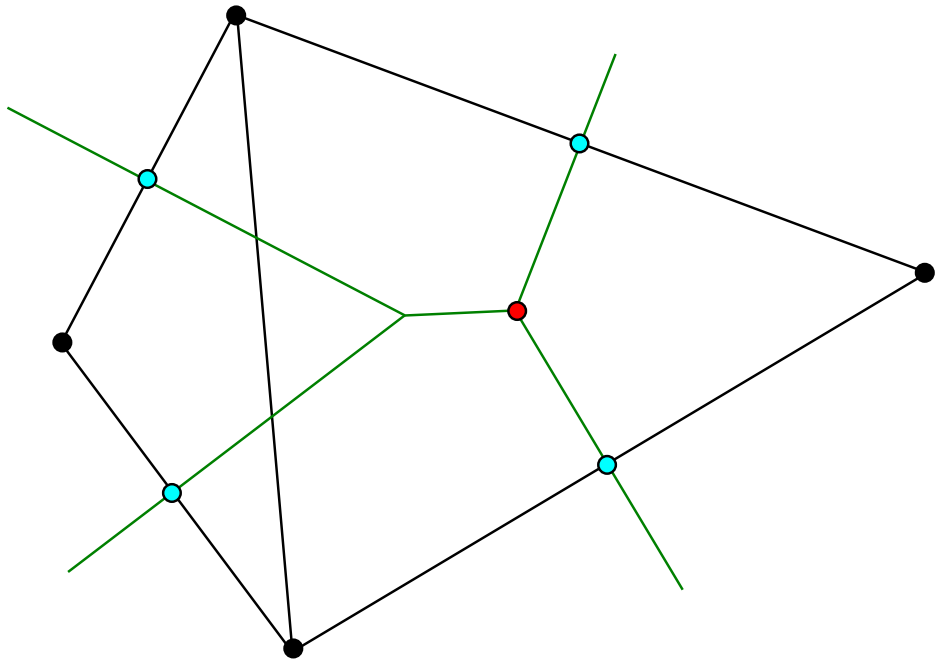
Flow complex



Flow complex



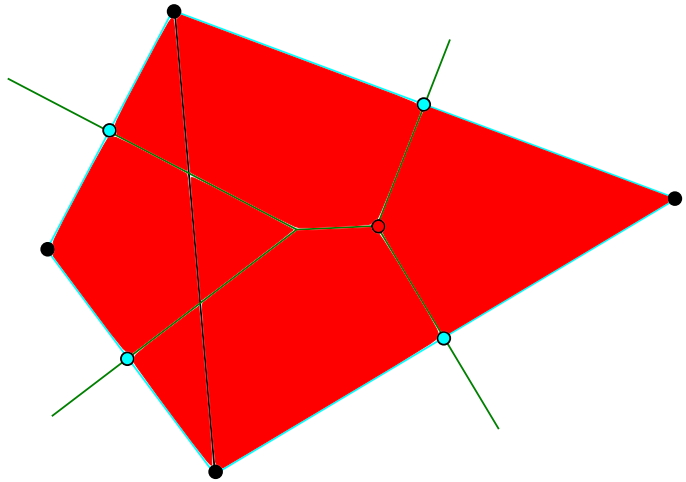
Flow complex



$$\sum_{i=0}^{d} (-1)^i \cdot n_i = 1$$

- n_i is the number of critical points of index i

Stable and unstable manifolds



- *stable manifold* of x contains the points, that flow into x

$$S(x) = \{y \in \mathbb{R}^d | \lim_{t \downarrow \infty} \phi(t, y) = x\}$$

- given a neighborhood U around x

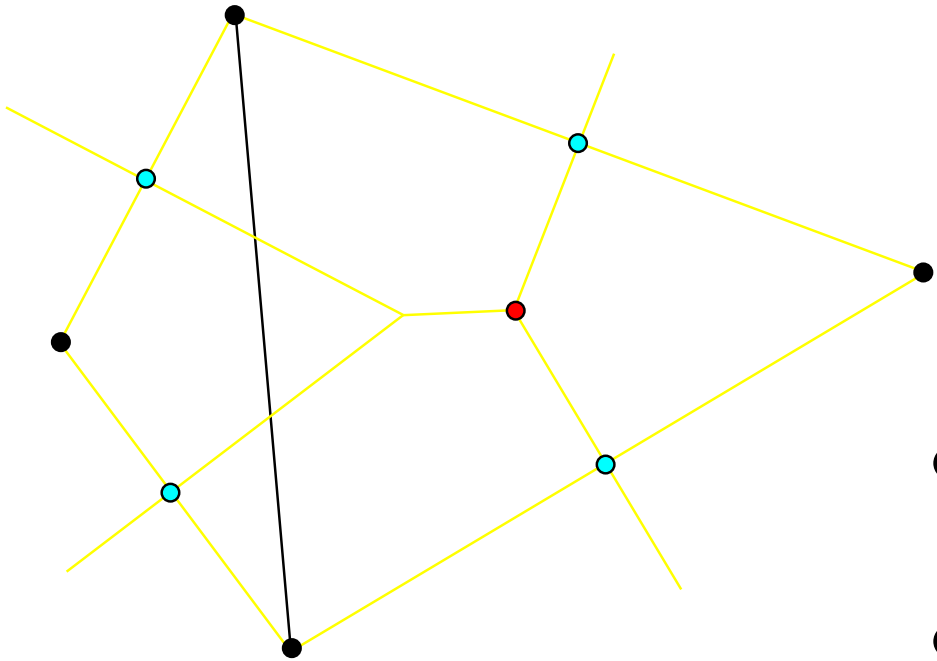
$$V(U) = \{y \in \mathbb{R}^d | \exists z \in U, t \geq 0 : \phi(t, z) = y\}$$

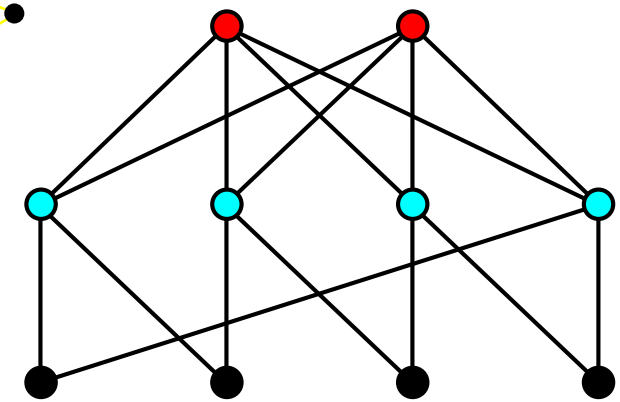
,

- *unstable manifold*

$$U(x) = \bigcap_{\mathrm{Neighborhood}\, U \,\mathrm{of}\, x} V(U)$$

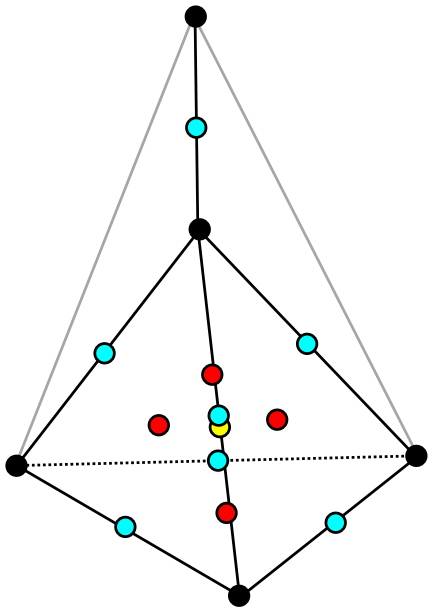
The Hasse diagram

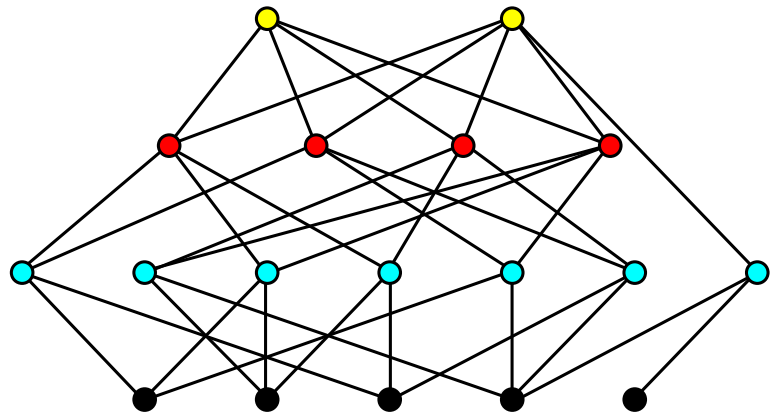




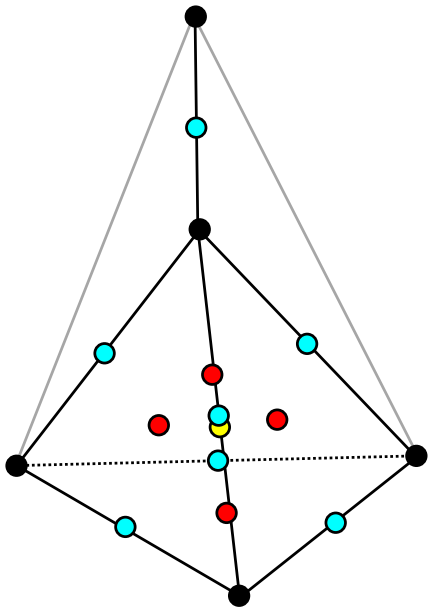
- for two critical points x and y, $S(y)$ is **incident** to $S(x)$, if $U(y) \cap S(x) \neq \emptyset$, i.e. there is a point in $U(y)$ that flows into x

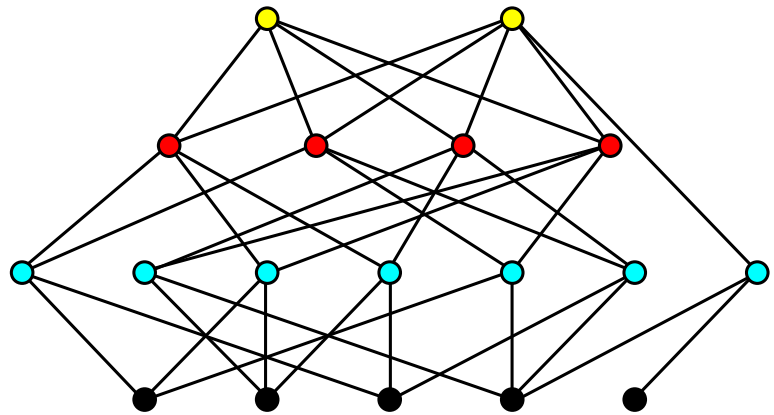
The algorithm





The algorithm





Two basic operations

1. Ascent: increasing $h(x)$ along the gradient
2. Descent: decreasing $h(x)$ along the gradient

The primitives

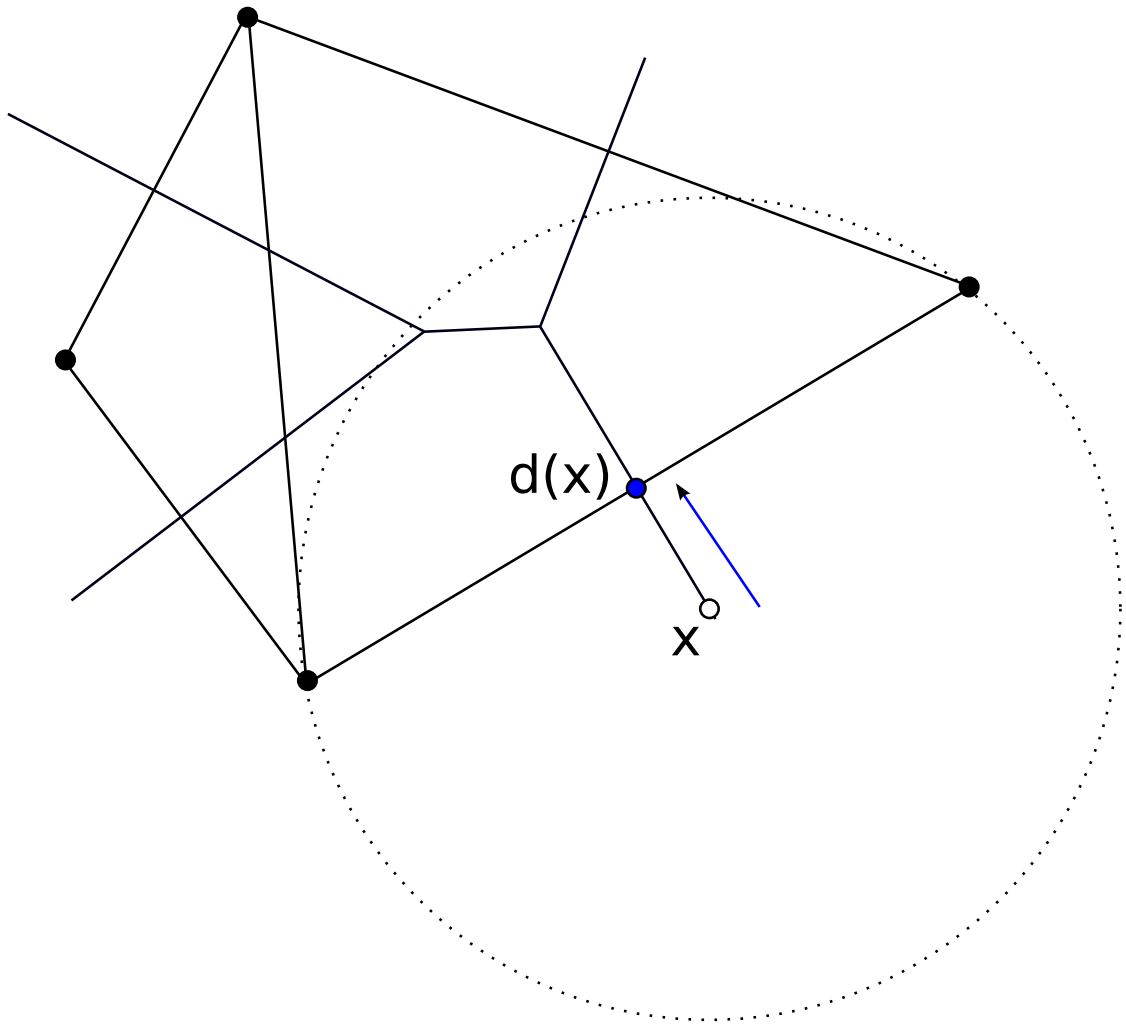
Nearest neighbor along a ray: Let v be the ray-vector, $p \in V$ and $q \in P \setminus V$. Solve $\|(x + t \cdot v) - q\|^2 = \|(x + t \cdot v) - p\|^2$, and pick the minimum $t_q > 0$, along with q

Projection onto an affine hull: In order to check for $x \in CONV(N(x))$, we compute the coefficients λ_i for which

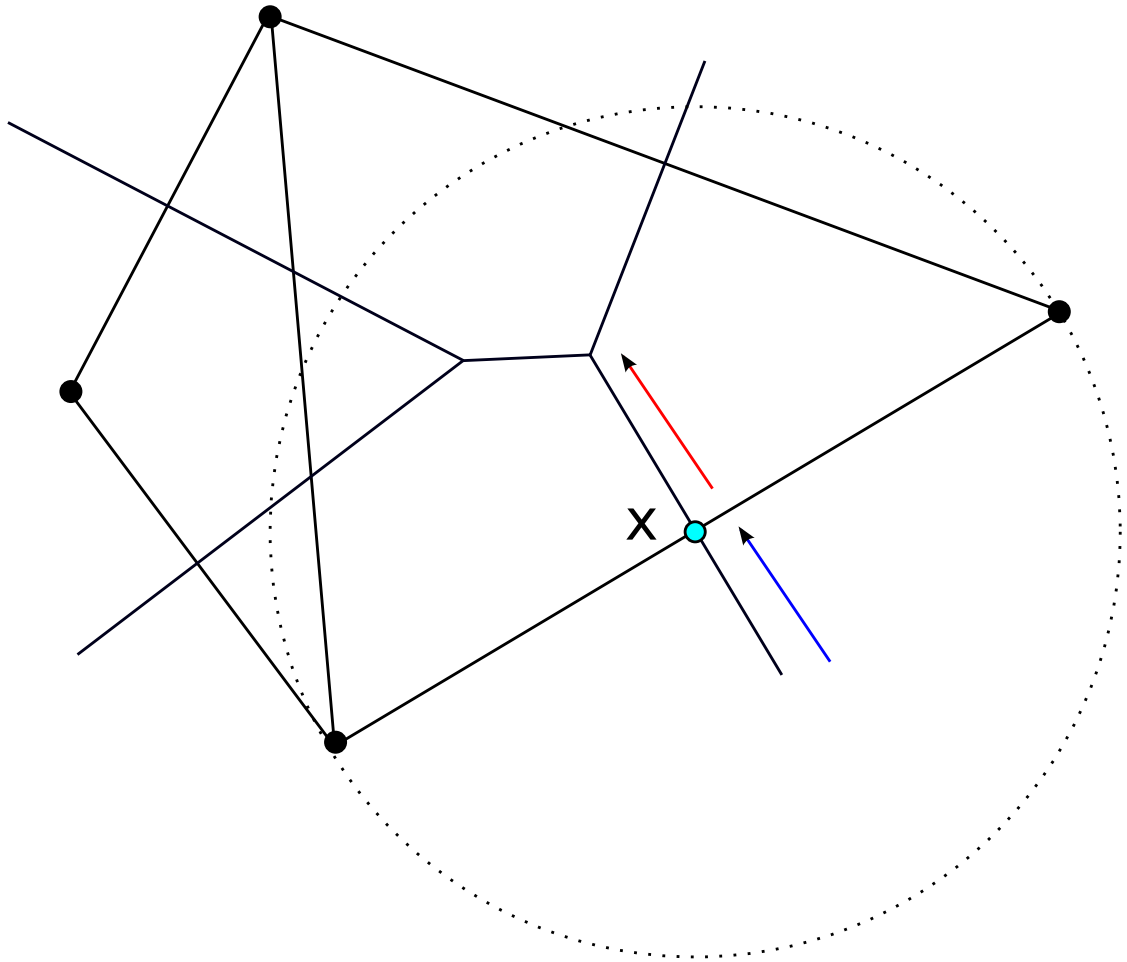
$$c = \sum_i \lambda_i \cdot p_i, \quad p_i \in V, \quad \sum \lambda_i = 1$$

Here c is the orthogonal-projection of x onto V. We use a dynamic QR-decomposition to solve the least-squares problem of computing the projection of x onto V.

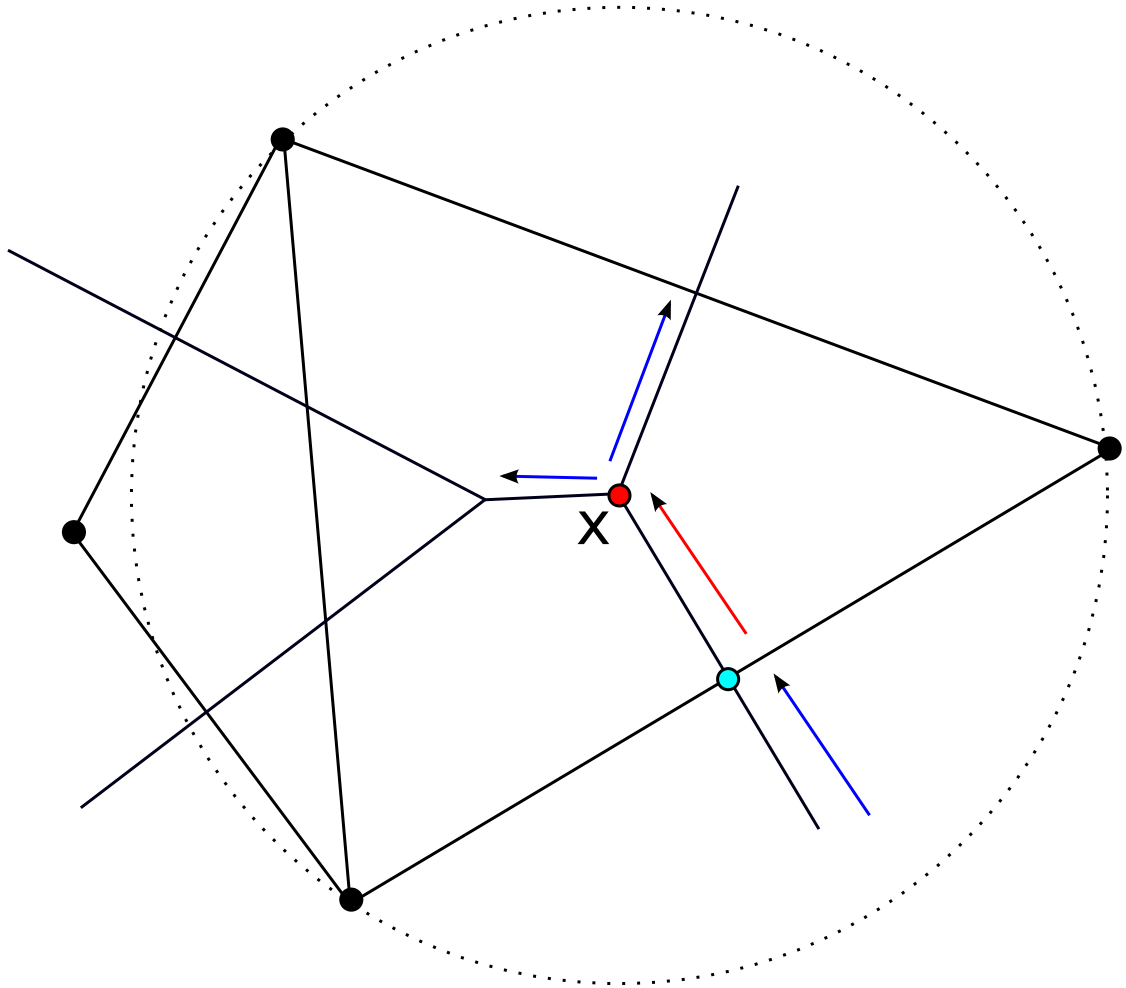
Example



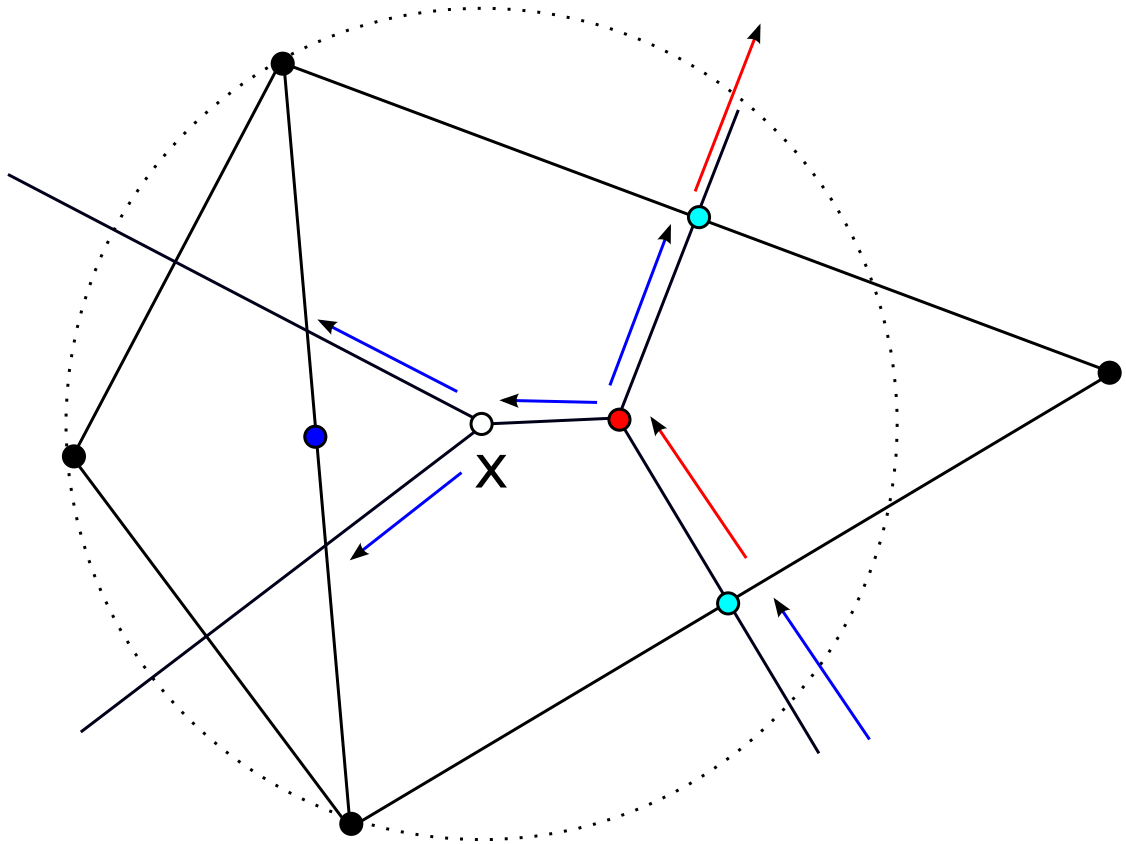
Example



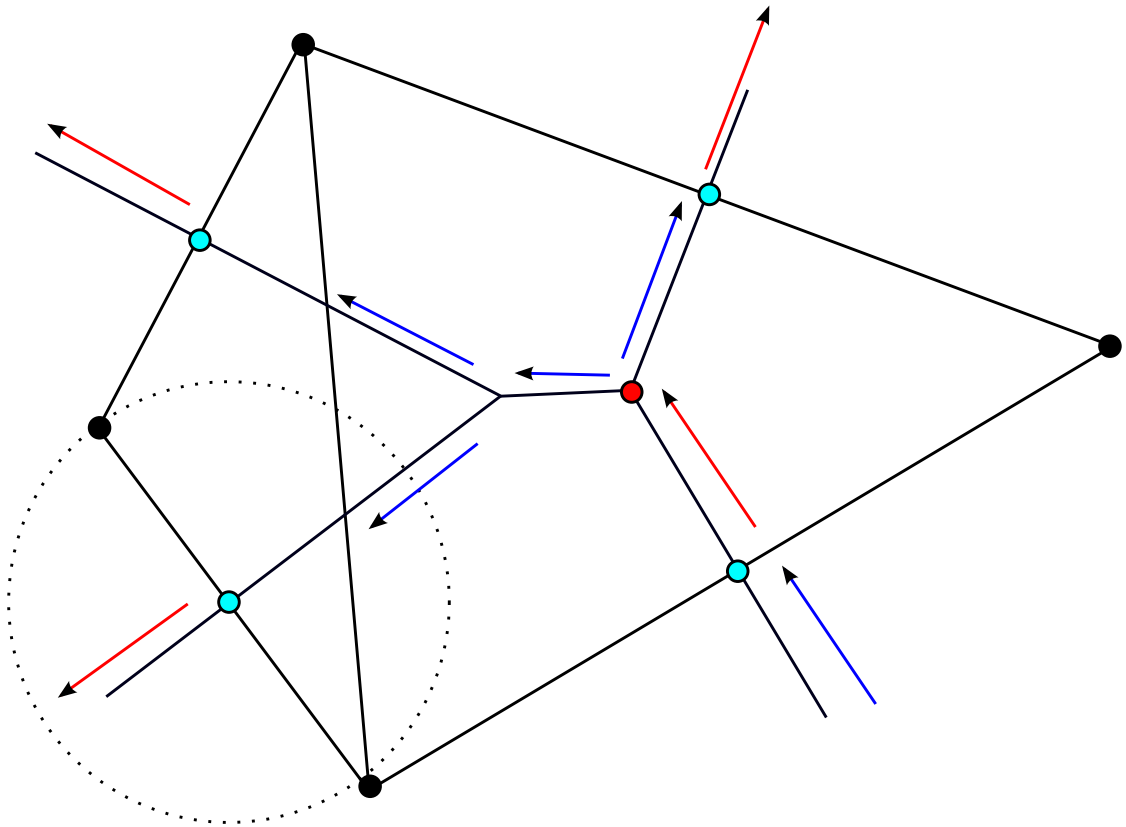
Example



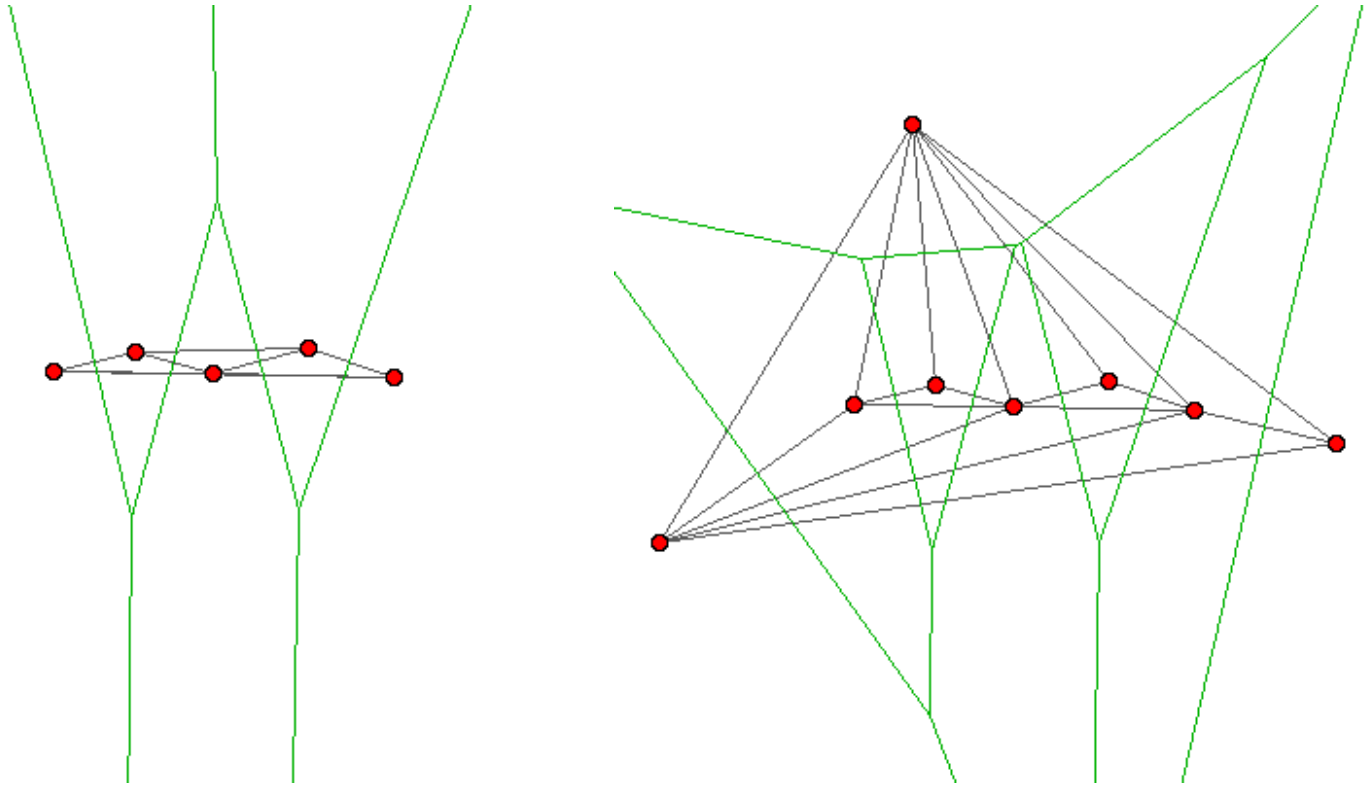
Example



Example



Bounding simplex



- low-dimensional(sub-)structures serialize the exploration
- a bounding simplex allows for a scalable parallel exploration

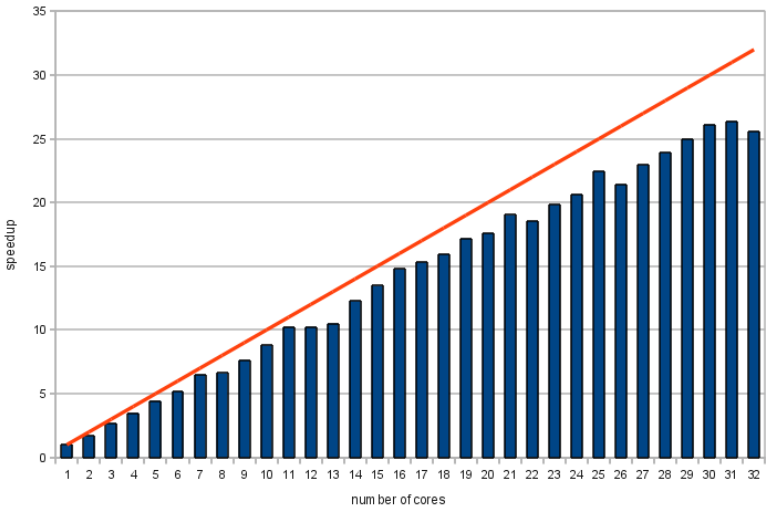
Experimental results

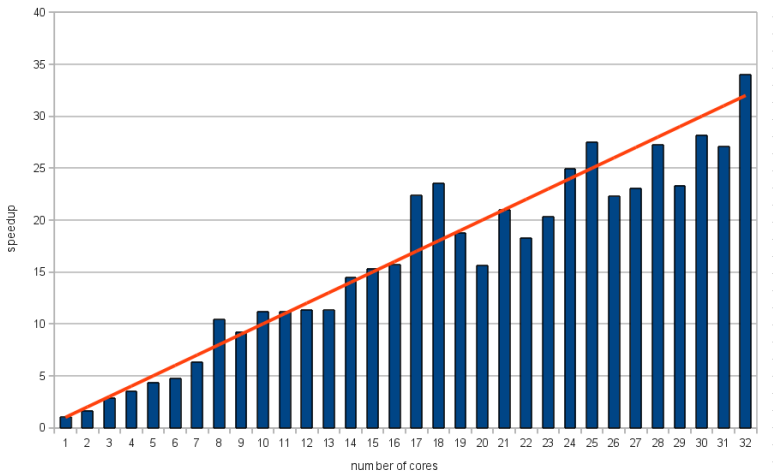
Critical point statistics

data set	dimension	distribution	time (s)
random within sphere	3	50-137-122-34	0.03
random within sphere	4	50-199-284-178-44	2.66
curve	3	50-73-56-32	0.7
spiral-3d	3	30-64-61-26	0.03
spiral-3d	4	30-41-12-0-0	0.97
spiral-3d	5	30-41-12-0-0-0	182.4
beethoven-3d-model	3	500-1076-649-72	11.81
cup-3d-model	3	300-766-599-132	0.92

Experimental results

Scalability





- On the left: 200 random points in 4 dimensions
- On the right: 100 random points in 5 dimensions

Conclusions

- a new algorithm to compute the flow complex, without explicit computation of the Delaunay triangulation
- numerical robustness
- it is inherently parallelizable, and scales well with the number of cores in a shared-memory architecture
- future work: high-level scalability by subdivision

Thank you!

Questions?