Volume computation and applications

Vissarion Fisikopoulos

Newcastle University, 3 Oct. 2017

National and Kapodistrian
UNIVERSITY OF ATHENS

Outline

Volumes, polytopes, applications

Algorithms and complexity

Polytope oracles in high dimensions

Future work and references

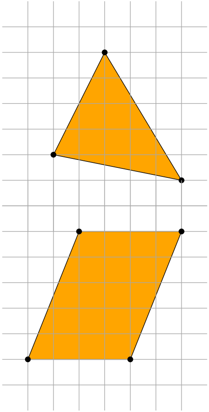
Our problem

Given P a convex polytope in $\mathbb{R}^d$ compute the volume of P.

Our problem

Given P a convex polytope in $\mathbb{R}^d$ compute the volume of P.

Some elementary polytopes (simplex, cube) have simple determinantal formulas.



$$\begin{vmatrix} 1 & 2 & 1 \\ 3 & 6 & 1 \\ 6 & 1 & 1 \end{vmatrix} /2! = 11$$

$$\begin{vmatrix} 2 & 5 \\ 4 & 0 \end{vmatrix} = 20$$

Birkhoff polytopes

- Given the complete bipartite graph $K_{n,n} = (V, E)$ a perfect matching is $M \subseteq E$ s.t. every vertex meets exactly one member of M

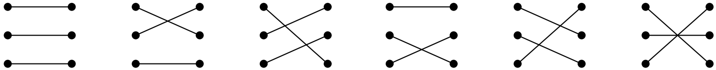
Birkhoff polytopes

- Given the complete bipartite graph $K_{n,n} = (V, E)$ a perfect matching is $M \subseteq E$ s.t. every vertex meets exactly one member of M
- $S \subseteq E,\ \chi_e^S = \{1 \text{ if } e \in S, 0 \text{ otherwise}\}$

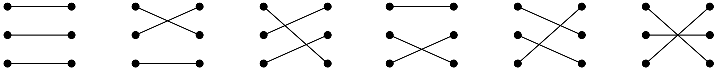
Birkhoff polytopes

- Given the complete bipartite graph $K_{n,n} = (V, E)$ a perfect matching is $M \subseteq E$ s.t. every vertex meets exactly one member of M
- $S \subseteq E,\ \chi_e^S = \{1 \text{ if } e \in S, 0 \text{ otherwise}\}$
- $B_n = conv\{\chi^M \mid M \text{ is a perfect matching of } K_{n,n}\}$

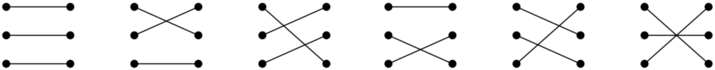
Birkhoff polytopes

- Given the complete bipartite graph $K_{n,n} = (V, E)$ a perfect matching is $M \subseteq E$ s.t. every vertex meets exactly one member of M
- $S \subseteq E,\ \chi_e^S = \{1 \text{ if } e \in S, 0 \text{ otherwise}\}$
- $B_n = conv\{\chi^M \mid M \text{ is a perfect matching of } K_{n,n}\}$
- 

Birkhoff polytopes

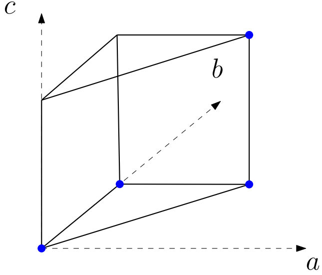
- Given the complete bipartite graph $K_{n,n} = (V, E)$ a perfect matching is $M \subseteq E$ s.t. every vertex meets exactly one member of M
- $S \subseteq E,\ \chi_e^S = \{1 \text{ if } e \in S, 0 \text{ otherwise}\}$
- $B_n = conv\{\chi^M \mid M \text{ is a perfect matching of } K_{n,n}\}$
- 
- # faces of B_3: 6, 15, 18, 9; $\mathrm{vol}(B_3) = 9/8$

Birkhoff polytopes

- Given the complete bipartite graph $K_{n,n} = (V, E)$ a perfect matching is $M \subseteq E$ s.t. every vertex meets exactly one member of M
- $S \subseteq E,\ \chi_e^S = \{1 \text{ if } e \in S, 0 \text{ otherwise}\}$
- $B_n = conv\{\chi^M \mid M \text{ is a perfect matching of } K_{n,n}\}$
- 
- # faces of B_3: 6, 15, 18, 9; $\mathrm{vol}(B_3) = 9/8$
- there exist formulas for the volume [deLoera et al '07] but values only known for $n \leq 10$ after 1yr of parallel computing [Beck et al '03]

Volumes and counting

- Given n elements & partial order; order polytope $P_O \subseteq [0,1]^n$ coordinates of points satisfies the partial order



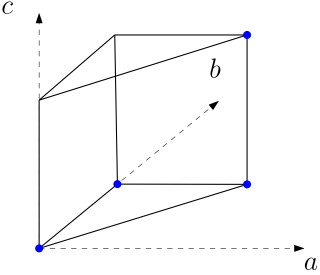
a, b, c

partial order: $a < b$

3 linear extensions: abc, acb, cab

Volumes and counting

- Given n elements & partial order; order polytope $P_O \subseteq [0,1]^n$ coordinates of points satisfies the partial order



a, b, c

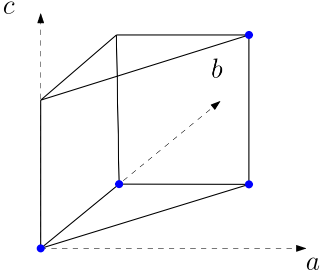
partial order: $a < b$

3 linear extensions: abc, acb, cab

- # linear extensions = volume of order polytope $\cdot\, n!$ [Stanley'86]

Volumes and counting

- Given n elements & partial order; order polytope $P_O \subseteq [0,1]^n$ coordinates of points satisfies the partial order



a, b, c

partial order: $a < b$

3 linear extensions: abc, acb, cab

- # linear extensions = volume of order polytope $\cdot\, n!$ [Stanley'86]
- Counting linear extensions is #P-hard [Brightwell'91]

Mixed volume

Let $P_1, P_2, \dots, P_d$ be polytopes in $\mathbb{R}^d$ then the mixed volume is

$$M(P_1, \dots, P_d) = \sum_{I \subseteq \{1,2,\dots,d\}} (-1)^{(d-|I|)} \cdot \mathrm{Vol}(\sum_{i \in I} P_i)$$

where the sum is the Minkowski sum.

Mixed volume

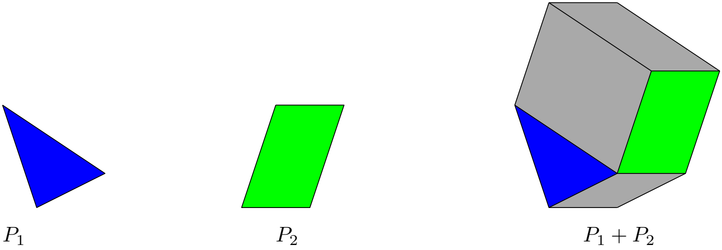
Let $P_1, P_2, \ldots, P_d$ be polytopes in $\mathbb{R}^d$ then the mixed volume is

$$M(P_1, \ldots, P_d) = \sum_{I \subseteq \{1,2,\ldots,d\}} (-1)^{(d-|I|)} \cdot \mathrm{Vol}(\sum_{i \in I} P_i)$$

where the sum is the Minkowski sum.

Example

For $d = 2$: $M(P_1, P_2) = \mathrm{Vol}(P_1 + P_2) - \mathrm{Vol}(P_1) - \mathrm{Vol}(P_2)$



Volumes and algebraic geometry

Toric geometry

If P is an integral d-dimensional polytope, then $d!$ times the volume of P is the degree of the toric variety associated to P.

BKK bound (Bernshtein, Khovanskii, Kushnirenko)

Let $f_1, \ldots, f_n$ be polynomials in $\mathbb{C}[x_1, \ldots, x_n]$. Let $N(f_j)$ denote the Newton polytope of f_j, i.e. the convex hull of it's exponent vectors. If $f_1, \ldots, f_n$ are generic, then the number of solutions of the polynomial system of equations

$$f_1 = 0, \ldots, f_n = 0$$

with no $x_i = 0$ is equal to the normalized mixed volume

$$n! M(N(f_1), \ldots, N(f_n))$$

Applications

- Volume of zonotopes is used to test methods for order reduction which is important in several areas: autonomous driving, human-robot collaboration and smart grids
- Volumes of intersections of polytopes are used in bio-geography to compute biodiversity and related measures e.g. [Barnagaud, Kissling, Tsirogiannis, Fisikopoulos, Villeger, Sekercioglu'17]

Outline

Volumes, polytopes, applications

Algorithms and complexity

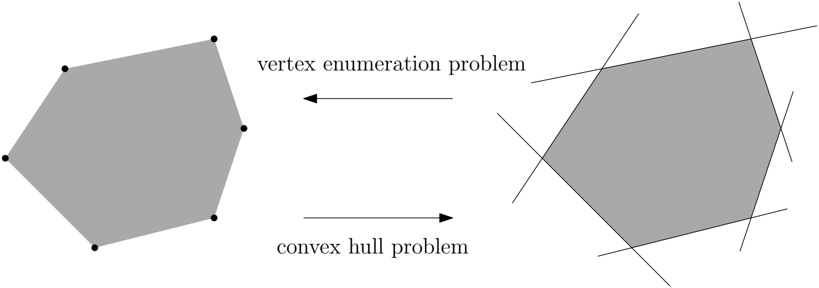
Polytope oracles in high dimensions

Future work and references

Explicit Polytope Representations

A convex polytope $P \subseteq \mathbb{R}^d$ can be represented as the

1. convex hull of a pointset $\{p_1, \ldots, p_m\}$ (V-representation)
2. intersection of halfspaces $\{h_1, \ldots, h_n\}$ (H-representation)



Faces of polytopes: vertices, edges, . . . , facets

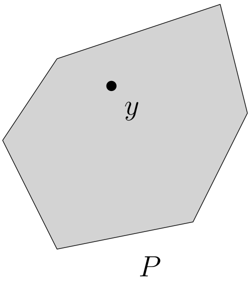
Implicit Polytope Representation (Oracles)

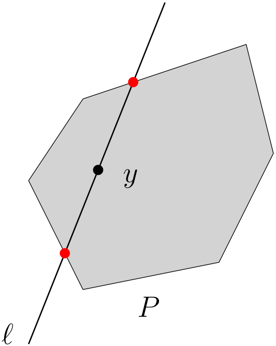
Membership oracle

Given point $y \in \mathbb{R}^d$, return yes if $y \in P$ otherwise return no.

Boundary oracle

Given point $y \in P$ and line ℓ goes through y return the points $\ell \cap \partial P$





Our setting

Input: Polytope $P := \{x \in \mathbb{R}^d \mid Ax \leq b\}$ $A \in \mathbb{R}^{m \times d}$, $b \in \mathbb{R}^m$

Output: Volume of P

Our setting

Input: Polytope $P := \{x \in \mathbb{R}^d \mid Ax \leq b\}$ $A \in \mathbb{R}^{m \times d}$, $b \in \mathbb{R}^m$

Output: Volume of P

Complexity

- #P-hard for vertex and for halfspace repres. [DyerFrieze'88]

Our setting

Input: Polytope $P := \{x \in \mathbb{R}^d \mid Ax \leq b\}$ $A \in \mathbb{R}^{m \times d}$, $b \in \mathbb{R}^m$

Output: Volume of P

Complexity

- #P-hard for vertex and for halfspace repres. [DyerFrieze'88]
- open if both vertex (V-rep) & halfspace (H-rep) representation is available

Our setting

Input: Polytope $P := \{x \in \mathbb{R}^d \mid Ax \leq b\}$ $A \in \mathbb{R}^{m \times d}$, $b \in \mathbb{R}^m$

Output: Volume of P

Complexity

- #P-hard for vertex and for halfspace repres. [DyerFrieze'88]
- open if both vertex (V-rep) & halfspace (H-rep) representation is available
- no deterministic poly-time algorithm can compute the volume with less than exponential relative error [Elekes'86]

Our setting

Input: Polytope $P := \{x \in \mathbb{R}^d \mid Ax \leq b\}$ $A \in \mathbb{R}^{m \times d}$, $b \in \mathbb{R}^m$

Output: Volume of P

Complexity

- #P-hard for vertex and for halfspace repres. [DyerFrieze'88]
- open if both vertex (V-rep) & halfspace (H-rep) representation is available
- no deterministic poly-time algorithm can compute the volume with less than exponential relative error [Elekes'86]
- randomized poly-time approximation of volume of a convex body with high probability and arbitrarily small relative error [DyerFriezeKannan'91]
  $O^*(d^{23}) \to O^*(d^4)$ [LovazVemp'04]

Implementations

Exact: VINCI [Bueler et al'00], Latte [deLoera et al], Qhull [Barber et al], LRS [Avis], Normaliz [Bruns et al]

- triangulation, sign decomposition methods
- cannot compute in high dimensions (e.g. > 15)

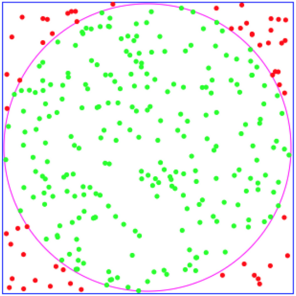
Randomized:

- [LovàszDeàk'12] cannot compute in > 10 dimensions

Goal: compute accurate estimations in few hundred dimensions

Uniform sampling

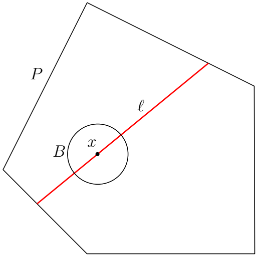
rejections techniques (sample from bounding box) fail in high dimensions



volume(unit cube) $= 1$
volume(unit ball) $\sim (c/d)^{d/2}$
–*drops exponentially with* d

Uniform sampling

for arbitrary polytopes we need *random walks*
e.g. grid walk, ball walk, hit-and-run (random directions)

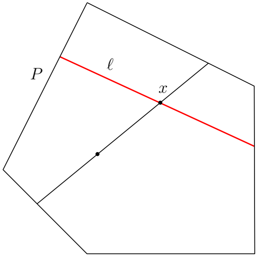


1. line ℓ through x, uniform on $B(x,1)$
2. set x to be a uniform disrtibuted point on $P \cap \ell$

Iterate this for W steps and return x.
$W = O^*(d^3)$ [LovaszVempala'06]

Uniform sampling

for arbitrary polytopes we need *random walks*
e.g. grid walk, ball walk, hit-and-run (random directions)



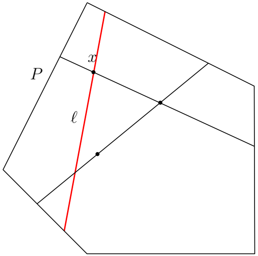
1. line ℓ through x, uniform on $B(x,1)$
2. set x to be a uniform disrtibuted point on $P \cap \ell$

Iterate this for W steps and return x.
$W = O^*(d^3)$ [LovaszVempala'06]

Uniform sampling

for arbitrary polytopes we need *random walks*

e.g. grid walk, ball walk, hit-and-run (random directions)

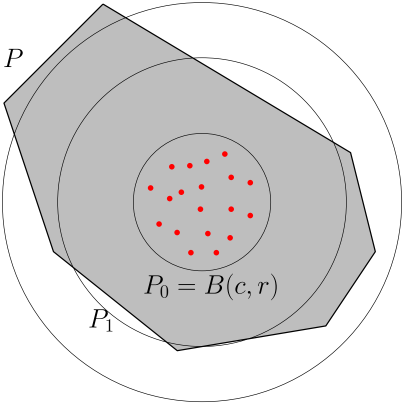


1. line ℓ through x, uniform on $B(x, 1)$
2. set x to be a uniform disrtibuted point on $P \cap \ell$

Iterate this for W steps and return x.

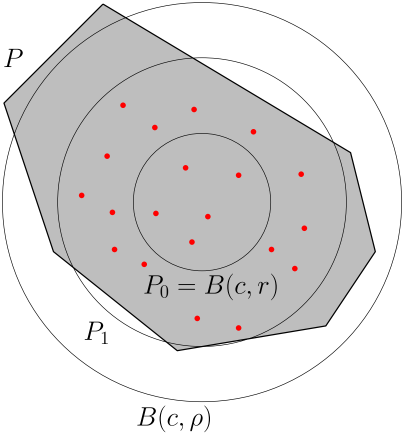
$W = O^*(d^3)$ [LovaszVempala'06]

Multiphase Monte Carlo (Generating random points)



- $B(c, 2^{i/d})$, $i = \alpha, \alpha+1, \ldots, \beta$,
  $\alpha = \lfloor d \log r \rfloor$, $\beta = \lceil d \log \rho \rceil$
- $P_i := P \cap B(c, 2^{i/d}),\ i = \alpha, \alpha+1, \ldots, \beta$
  $P_\alpha = B(c, 2^{\alpha/d}) \subseteq B(c, r)$

Multiphase Monte Carlo (Generating random points)



- $B(c, 2^{i/d}),\ i = \alpha, \alpha+1, \ldots, \beta,$
  $\alpha = \lfloor d \log r \rfloor,\ \beta = \lceil d \log \rho \rceil$
- $P_i := P \cap B(c, 2^{i/d}),\ i = \alpha, \alpha+1, \ldots, \beta$
  $P_\alpha = B(c, 2^{\alpha/d}) \subseteq B(c, r)$

1. Use rand points in P_{i-1} to generate rand points in P_i
2. Count how many rand. points in P_i fall in P_{i-1}

$$vol(P) = vol(P_\alpha) \prod_{i=\alpha+1}^{\beta} \frac{vol(P_i)}{vol(P_{i-1})}$$

Complexity [KannanLS'97]

Assuming $B(c, 1) \subseteq P \subseteq B(c, \rho)$, the volume algorithm returns an estimation of vol(P), which lies between $(1-\epsilon)$vol(P) and $(1+\epsilon)$vol(P) with probability $\geq 3/4$, making

$$O^*(d^5)$$

oracle calls, where ρ is the radius of a bounding ball for P.
Isotropic sandwitching: $\rho = O^*(\sqrt{d})$ and ball walk.

Runtime steps

- generates $d \log(\rho)$ balls
- generate $N = 400\epsilon^{-2} d \log d$ random points in each ball $\cap P$
- each point is computed after $O^*(d^3)$ random walk steps

Our contribution [Emiris-Fisikopoulos]

- $W = O(d)$ random walk steps (vs. $O^*(d^3)$ which hides constant 10^{11}) achieve $< 1\%$ error in up to 100 dim.

Our contribution [Emiris-Fisikopoulos]

- $W = O(d)$ random walk steps (vs. $O^*(d^3)$ which hides constant 10^{11}) achieve $< 1\%$ error in up to 100 dim.
- sample partial generations of $\leq N$ points in each ball $\cap P$ (starting from the largest ball)

Our contribution [Emiris-Fisikopoulos]

- $W = O(d)$ random walk steps (vs. $O^*(d^3)$ which hides constant 10^{11}) achieve $< 1\%$ error in up to 100 dim.
- sample partial generations of $\leq N$ points in each ball $\cap P$ (starting from the largest ball)
- iterative rounding using minimum ellipsoid (efficiently handle skinny polytopes)

Our contribution [Emiris-Fisikopoulos]

- $W = O(d)$ random walk steps (vs. $O^*(d^3)$ which hides constant 10^{11}) achieve $< 1\%$ error in up to 100 dim.
- sample partial generations of $\leq N$ points in each ball $\cap P$ (starting from the largest ball)
- iterative rounding using minimum ellipsoid (efficiently handle skinny polytopes)(CDHR)

Our contribution [Emiris-Fisikopoulos]

- $W = O(d)$ random walk steps (vs. $O^*(d^3)$ which hides constant 10^{11}) achieve $< 1\%$ error in up to 100 dim.
- sample partial generations of $\leq N$ points in each ball $\cap P$ (starting from the largest ball)
- iterative rounding using minimum ellipsoid (efficiently handle skinny polytopes)(CDHR)
- boundary oracles using coordinate hit-and-run(CDHR) with $O(m)$ runtime (vs. $O(md)$ using RDHR)

Our contribution [Emiris-Fisikopoulos]

- $W = O(d)$ random walk steps (vs. $O^*(d^3)$ which hides constant 10^{11}) achieve $< 1\%$ error in up to 100 dim.
- sample partial generations of $\leq N$ points in each ball $\cap P$ (starting from the largest ball)
- iterative rounding using minimum ellipsoid (efficiently handle skinny polytopes)(CDHR)
- boundary oracles using coordinate hit-and-run(CDHR) with $O(m)$ runtime (vs. $O(md)$ using RDHR)
- CDHR as accurate as RDHR

Our contribution [Emiris-Fisikopoulos]

- $W = O(d)$ random walk steps (vs. $O^*(d^3)$ which hides constant 10^{11}) achieve $< 1\%$ error in up to 100 dim.
- sample partial generations of $\leq N$ points in each ball $\cap P$ (starting from the largest ball)
- iterative rounding using minimum ellipsoid (efficiently handle skinny polytopes)(CDHR)
- boundary oracles using coordinate hit-and-run(CDHR) with $O(m)$ runtime (vs. $O(md)$ using RDHR)
- CDHR as accurate as RDHR

Software framework to test theoretical ideas and conjectures

More experimental results

- approximate the volume of a series of polytopes (cubes, random, cross, birkhoff, order) up to dimension 200 in less than 2 hours with mean approximation error at most 1%

More experimental results

- approximate the volume of a series of polytopes (cubes, random, cross, birkhoff, order) up to dimension 200 in less than 2 hours with mean approximation error at most 1%
- Running our software on order polytopes offers a method for estimating the number of linear extensions of a poset

More experimental results

- approximate the volume of a series of polytopes (cubes, random, cross, birkhoff, order) up to dimension 200 in less than 2 hours with mean approximation error at most 1%
- Running our software on order polytopes offers a method for estimating the number of linear extensions of a poset
- Compute the volume of Birkhoff polytopes $B_{11}, \ldots, B_{15}$ in few hrs whereas exact methods have only computed that of B_{10} by specialized software in $\sim$ 1 year of parallel computation

  Note: lattice points of Birkhoff are important in Bayesian statistics and volume is leading coefficient of the generating (Ehrhart) polynomial

Volumes of Birkhoff polytopes

n	d	estimation	asymptotic [CanfieldMcKay09]	$\frac{\text{estimation}}{\text{asymptotic}}$	exact	$\frac{\text{exact}}{\text{asymptotic}}$
3	4	1.12E+000	1.41E+000	0.7932847	1.13E+000	0.7973923
4	9	6.79E-002	7.61E-002	0.8919349	6.21E-002	0.8159296
5	16	1.41E-004	1.69E-004	0.8344350	1.41E-004	0.8341903
6	25	7.41E-009	8.62E-009	0.8598669	7.35E-009	0.8527922
7	36	5.67E-015	6.51E-015	0.8713891	5.64E-015	0.8665047
8	49	4.39E-023	5.03E-023	0.8729497	4.42E-023	0.8778632
9	64	2.62E-033	2.93E-033	0.8960767	2.60E-033	0.8874117
10	81	8.14E-046	9.81E-046	0.8305162	8.78E-046	0.8955491
11	100	1.40E-060	1.49E-060	0.9342584	???	???
12	121	7.85E-078	8.38E-078	0.9370513	???	???
13	144	1.33E-097	1.43E-097	0.9331517	???	???
14	169	5.96E-120	6.24E-120	0.9550089	???	???
15	196	5.70E-145	5.94E-145	0.9593786	???	???

Outline

Volumes, polytopes, applications

Algorithms and complexity

Polytope oracles in high dimensions

Future work and references

Approximate oracles

- Issue: algorithm inefficient for large number of facets (e.g. crosspolytopes have 2^d facets; volume computation inefficient for $d > 20$ in practice)

Approximate oracles

- Issue: algorithm inefficient for large number of facets (e.g. crosspolytopes have 2^d facets; volume computation inefficient for $d > 20$ in practice)
- Solution: approximate oracles (membership and boundary)

Approximate oracles

- Issue: algorithm inefficient for large number of facets (e.g. crosspolytopes have 2^d facets; volume computation inefficient for $d > 20$ in practice)
- Solution: approximate oracles (membership and boundary)
- Previous work: only for membership oracle; solutions depend exponentially in d [Bentley et al 82] [Arya et al '16]

Approximate oracles

- Issue: algorithm inefficient for large number of facets (e.g. crosspolytopes have 2^d facets; volume computation inefficient for $d > 20$ in practice)
- Solution: approximate oracles (membership and boundary)
- Previous work: only for membership oracle; solutions depend exponentially in d [Bentley et al 82] [Arya et al '16]
- Idea: reduce problem to Nearest-Neighbor (NN) and use approximate NN algorithms

Approximate oracles

- Issue: algorithm inefficient for large number of facets (e.g. crosspolytopes have 2^d facets; volume computation inefficient for $d > 20$ in practice)
- Solution: approximate oracles (membership and boundary)
- Previous work: only for membership oracle; solutions depend exponentially in d [Bentley et al 82] [Arya et al '16]
- Idea: reduce problem to Nearest-Neighbor (NN) and use approximate NN algorithms

Side note: Approximate oracle improve sampling performance so they are applicable to other algorithms that use sampling such as optimization

Approximate Nearest Neighbors

Definition

Given a pointset $X \subset \mathbb{R}^d$ with cardinality n and an approximation parameter $\epsilon \in (0, 1)$, preprocess X into a data structure, so that, given a query point q it is possible to efficiently determine a point $x^* \in X$ such that:

$$d(x^*, q) \leq (1 + \epsilon) \min_{x \in X} f(x, q),$$

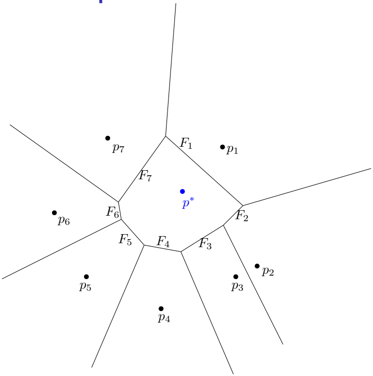
where f is a distance function.

Locality sensitive hashing (LSH) [Gionis et 'al '99]

randomized algorithm, sublinear dependence on n, polynomial dependence on d, exponential dependence on ϵ.

Approximate membership oracles

- For arbitrary point $p^* \in P$ compute symmetric points about each facet of P
- query: $q \in P$ iff p^* is the NN of q



Theorem (Anangostopoulos, Emiris, Fisikopoulos'17)

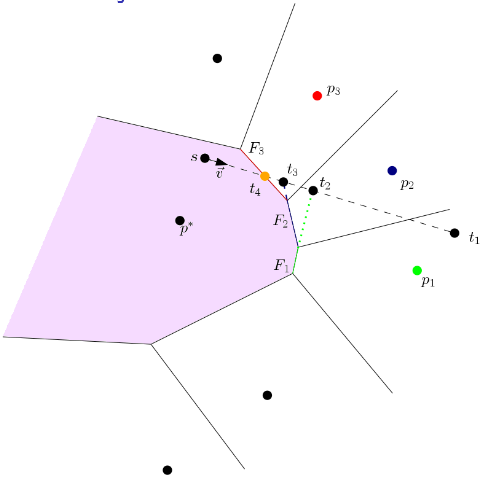
Given polytope $P \subset \mathbb{R}^d$, approximation parameter $\epsilon \in (0,1)$ and query point q, we can answer whether $q \in P$ in $O^*(dn^{\rho+o(1)})$ time with a probability of success $p = 1/(2(1+\epsilon)2-1)$, using $O^*(n^{1+\rho+o(1)} + dn)$ space; ρ is a probability related to LSH. If $f(q, \partial P) \leq \epsilon \cdot diam(P)$ the data structure can answer either way.

Approximate boundary oracles

Start point $t_1 \notin P$. At some step i :

- let p_i be the NN of t_i
- let H_i be the hyperplane supporting the facet F_i
- set $t_{i+1} = (H_i \cap r)$.

Iterate until $t_i \in P$ for some i



Note

The algorithm terminates even for the approximate case with a changed termination criterion and by doing e-steps to avoid "local optima".

Outline

Future & related work

- Volume of spectahedral shadows (joint with Konaxis, Tsigaridas)
- Volume of simple convex bodies: intersection of a simplex with two halfspaces or an ellipsoid; applications in finance (joint with Emiris, Chalkis)
- Sampling on V-polytopes or polytopes given by optimization oracles [Emiris, Fisikopoulos, Konaxis, Penaranda '13]

Open question

V-polytope sampling

Given $P \in \mathbb{R}^d$ any point $q \in P$ can be written as

$$q = \sum_{i=\{0,\ldots,m\}} \lambda_i p_i, \qquad \lambda_i \leq 0, \sum_{i=\{0,\ldots,m\}} \lambda_i = 0$$

compute distribution for λ s.t. q is uniform in P.

References

- Practical polytope volume approximation
  Emiris, Fisikopoulos
  ACM Transactions on Mathematical Software (to appear)

- Polytope oracles in high dimensions
  Anangostopoulos, Emiris, Fisikopoulos
  work in progress

Code

- C++ using `CGAL` and `Eigen` libraries
- `https://github.com/vissarion/volume_approximation`

Thank you! Questions?