Magnus Wahlström[1][a]

Computational Complexity of Star Discrepancy Computation

[1]Max-Planck-Institut für Informatik, Germany

August 19, 2010

[a]Joint work with Panos Giannopoulos, Christian Knauer, and Daniel Werner.

Outline

Star Discrepancy

Star Discrepancy

max planck institut informatik

Discrepancy (regularity) measure for sets of points

- Pointset P: n points in $[0, 1]^d$
- Anchored boxes B:

$$B = [0, b_1) \times \cdots \times [0, b_d) \subset [0, 1]^d$$

- Let $\mathrm{Vol}(B)$ =volume of B, $\mathrm{Pts}(B) = \frac{|B \cap P|}{n}$
- The *star discrepancy* of P is

$$d^*_\infty(P) = \max_B |\mathrm{Vol}(B) - \mathrm{Pts}(B)|$$

where B ranges over all anchored boxes.

Star Discrepancy

max planck institut informatik

Discrepancy (regularity) measure for sets of points

- Pointset P: n points in $[0,1]^d$
- Anchored boxes B:

$$B = [0, b_1) \times \cdots \times [0, b_d) \subset [0,1]^d$$

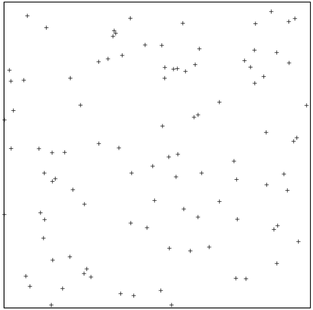
- Let $\mathrm{Vol}(B)$ =volume of B, $\mathrm{Pts}(B) = \frac{|B \cap P|}{n}$
- The *star discrepancy* of P is

$$d^*_\infty(P) = \max_B |\mathrm{Vol}(B) - \mathrm{Pts}(B)|$$

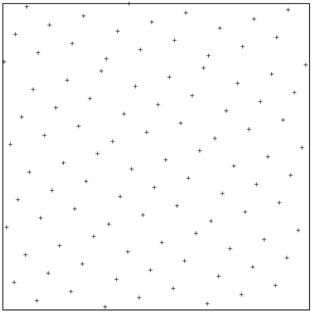
where B ranges over all anchored boxes.

100 pts, 2 dim

max planck institut informatik

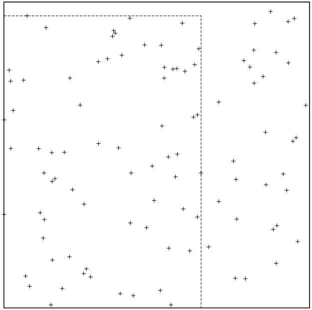


Random set

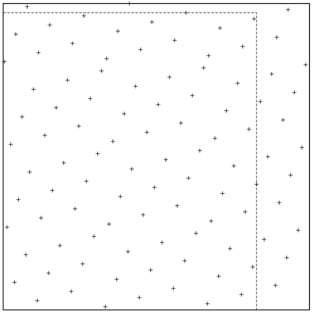


Hammersley

100 pts, 2 dim



Random set, worst box
Volume 0.616, 70 points
Discrepancy 0.084

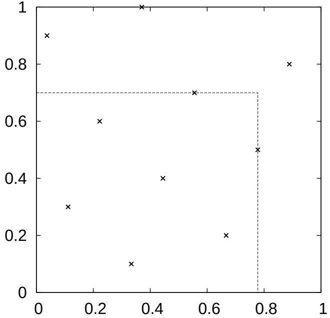


Hammersley, worst box
Volume 0.802, 83 points
Discrepancy 0.028

Star Discrepancy Computation

Discrepancy computation

max planck institut informatik



7 points, volume 0.544
Discrepancy 0.156

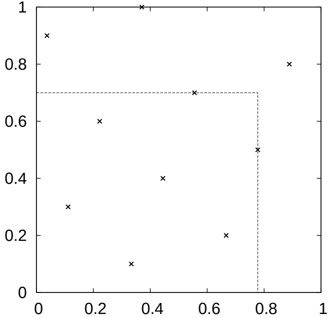
Naïve computation:

- Box defined by d points or axes
- $(n+1)^d$ boxes in d dimensions
- $O(nd)$ time per box to count

Problem is discrete, time $O(dn^{d+1})$

Discrepancy computation

max planck institut informatik



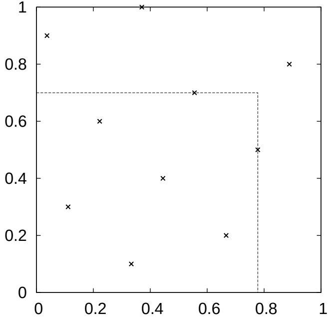
7 points, volume 0.544
Discrepancy 0.156

Naïve computation:

- Box defined by d points or axes
- $(n+1)^d$ boxes in d dimensions
- $O(nd)$ time per box to count

Problem is discrete, time $O(dn^{d+1})$

Discrepancy computation



7 points, volume 0.544
Discrepancy 0.156

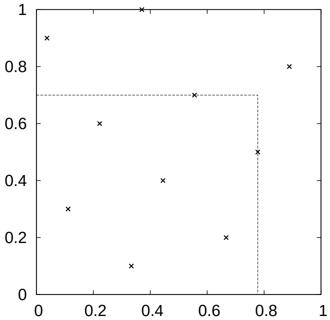
Naïve computation:

- Box defined by d points or axes
- $(n+1)^d$ boxes in d dimensions
- $O(nd)$ time per box to count

Problem is discrete, time $O(dn^{d+1})$

Discrepancy computation

max planck institut
informatik



7 points, volume 0.544
Discrepancy 0.156

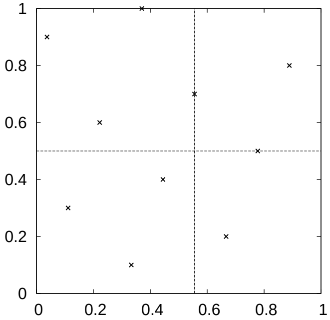
Naïve computation:

- Box defined by d points or axes
- $(n+1)^d$ boxes in d dimensions
- $O(nd)$ time per box to count

Problem is discrete, time $O(dn^{d+1})$

Improved method

max planck institut informatik

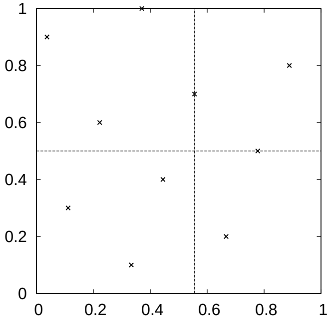


Not all boxes are meaningful

Improved method: still $O(n^d)$
(Bundschuh, Zhu: $n^d/d!$ tests)

Dobkin, Eppstein, Mitchell: $O(n^{d/2+1})$ total time (not shown)

Improved method



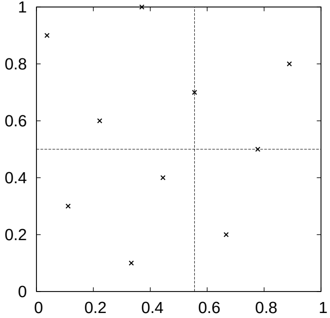
Not all boxes are meaningful

Improved method: still $O(n^d)$
(Bundschuh, Zhu: $n^d/d!$ tests)

Dobkin, Eppstein, Mitchell: $O(n^{d/2+1})$ total time (not shown)

Improved method

max planck institut informatik

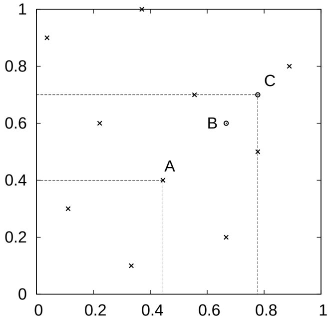


Not all boxes are meaningful

Improved method: still $O(n^d)$
(Bundschuh, Zhu: $n^d/d!$ tests)

Dobkin, Eppstein, Mitchell: $O(n^{d/2+1})$ total time (not shown)

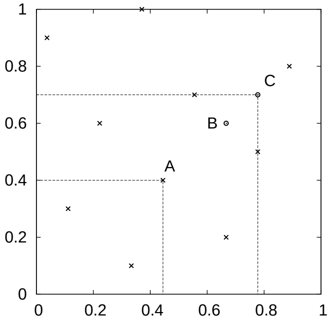
Approximation



- Assume B worst case, underfull
- $(\mathrm{Vol}(B) - \mathrm{Pts}(B)) \leq (\mathrm{Vol}(C) - \mathrm{Pts}(A))$
- $(\mathrm{Vol}(B) - \mathrm{Pts}(B)) \geq (\mathrm{Vol}(A) - \mathrm{Pts}(A))$
- $(\mathrm{Vol}(C) - \mathrm{Pts}(A))$ upper bound, error $\leq Vol(C) - Vol(A)$
- Similar if B overfull

Good set of sample points independent of instance

Approximation

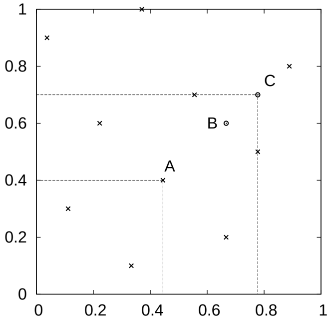


- Assume B worst case, underfull
- $(\mathrm{Vol}(B) - \mathrm{Pts}(B)) \leq (\mathrm{Vol}(C) - \mathrm{Pts}(A))$
- $(\mathrm{Vol}(B) - \mathrm{Pts}(B)) \geq (\mathrm{Vol}(A) - \mathrm{Pts}(A))$
- $(\mathrm{Vol}(C) - \mathrm{Pts}(A))$ upper bound, error $\leq Vol(C) - Vol(A)$
- Similar if B overfull

Good set of sample points independent of instance

Approximation

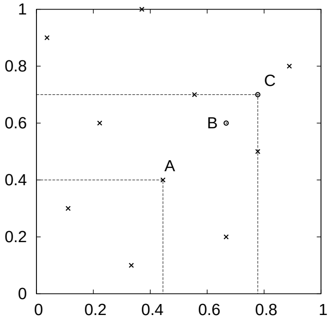
max planck institut informatik



- Assume B worst case, underfull
- $(\text{Vol}(B) - \text{Pts}(B)) \leq (\text{Vol}(C) - \text{Pts}(A))$
- $(\text{Vol}(B) - \text{Pts}(B)) \geq (\text{Vol}(A) - \text{Pts}(A))$
- $(\text{Vol}(C) - \text{Pts}(A))$ upper bound, error $\leq Vol(C) - Vol(A)$
- Similar if B overfull

Good set of sample points independent of instance

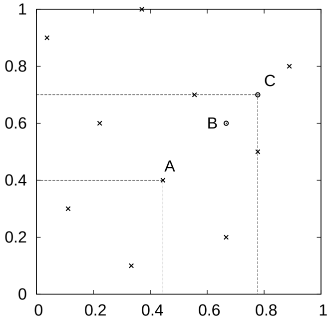
Approximation



- Assume B worst case, underfull
- $(\mathrm{Vol}(B) - \mathrm{Pts}(B)) \leq (\mathrm{Vol}(C) - \mathrm{Pts}(A))$
- $(\mathrm{Vol}(B) - \mathrm{Pts}(B)) \geq (\mathrm{Vol}(A) - \mathrm{Pts}(A))$
- $(\mathrm{Vol}(C) - \mathrm{Pts}(A))$ upper bound, error $\leq Vol(C) - Vol(A)$
- Similar if B overfull

Good set of sample points independent of instance

Approximation

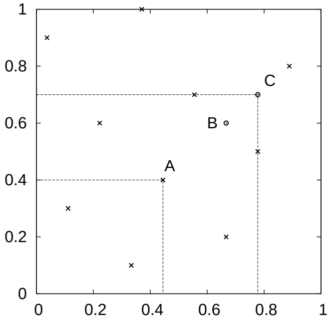


- Assume B worst case, underfull
- $(\mathrm{Vol}(B) - \mathrm{Pts}(B)) \leq (\mathrm{Vol}(C) - \mathrm{Pts}(A))$
- $(\mathrm{Vol}(B) - \mathrm{Pts}(B)) \geq (\mathrm{Vol}(A) - \mathrm{Pts}(A))$
- $(\mathrm{Vol}(C) - \mathrm{Pts}(A))$ upper bound, error $\leq Vol(C) - Vol(A)$
- Similar if B overfull

Good set of sample points independent of instance

Approximation

max planck institut informatik

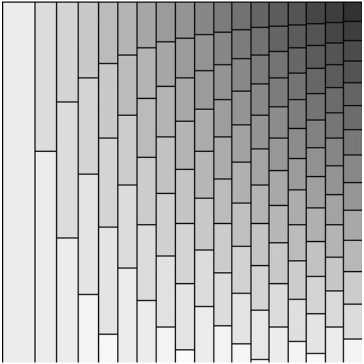


- Assume B worst case, underfull
- $(\mathrm{Vol}(B) - \mathrm{Pts}(B)) \leq (\mathrm{Vol}(C) - \mathrm{Pts}(A))$
- $(\mathrm{Vol}(B) - \mathrm{Pts}(B)) \geq (\mathrm{Vol}(A) - \mathrm{Pts}(A))$
- $(\mathrm{Vol}(C) - \mathrm{Pts}(A))$ upper bound, error $\leq Vol(C) - Vol(A)$
- Similar if B overfull

Good set of sample points independent of instance

Approximation: bracketing covers

max planck institut informatik



(Thiémard)
Shown: Vol(upper) $\leq$ Vol(lower) $+ 0.1$

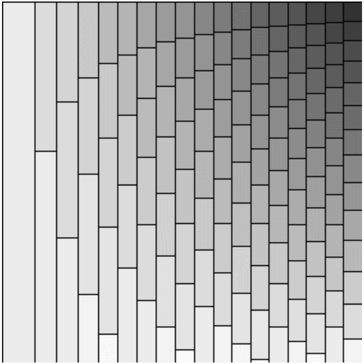
Sampling all corners gives upper bound $\leq d^*_\infty(P) + 0.1$

Error bound ϵ: $>$ $(1/\epsilon)^d$ tests (Gnewuch)

$\epsilon \approx d^*_\infty(P)$ requires $(1/\epsilon) > \sqrt{nd}$

Approximation: bracketing covers

max planck institut informatik



(Thiémard)
Shown: Vol(upper) $\leq$ Vol(lower) $+ 0.1$

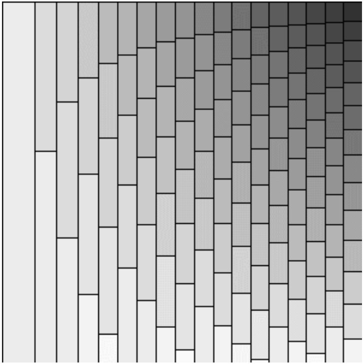
Sampling all corners gives upper bound $\leq d^*_\infty(P) + 0.1$

Error bound ϵ: $>$ $(1/\epsilon)^d$ tests (Gnewuch)

$\epsilon \approx d^*_\infty(P)$ requires $(1/\epsilon) > \sqrt{nd}$

Approximation: bracketing covers

max planck institut informatik



(Thiémard)
Shown: Vol(upper) $\leq$ Vol(lower) + 0.1

Sampling all corners gives upper bound $\leq d^*_\infty(P) + 0.1$

Error bound ϵ: $> (1/\epsilon)^d$ tests (Gnewuch)

$\epsilon \approx d^*_\infty(P)$ requires $(1/\epsilon) > \sqrt{nd}$

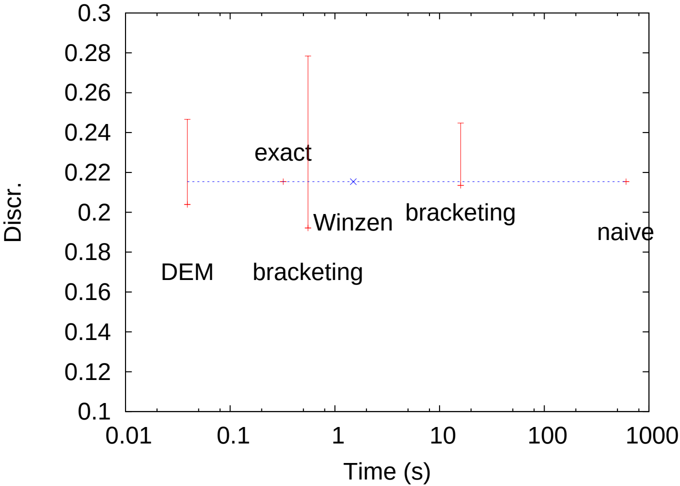
Computation methods, summary

max planck institut informatik

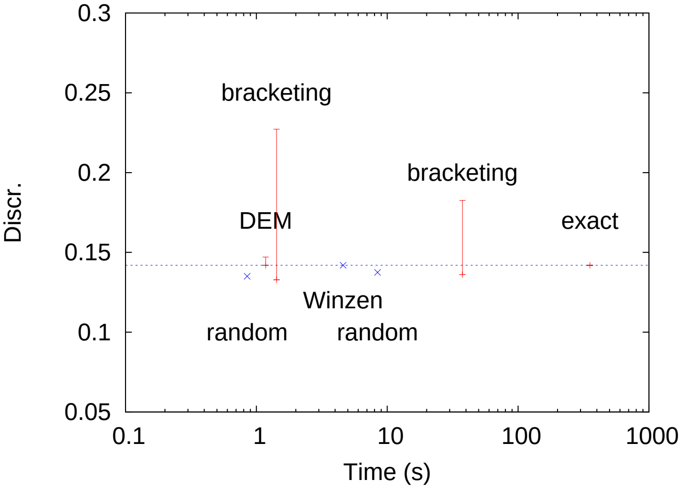
- Exact methods
  - Naïve: check all boxes. $O(n^{d+1})$ time.
  - Improved: $O(n^{d+1}/d!)$ time (Bundschuh, Zhu)
  - Dobkin, Eppstein, Mitchell: $O(n^{d/2+1})$ time.
- Inexact methods
  - Bracketing covers (Thiémard). Error ϵ in time $O(f(d,\epsilon) \cdot n)$; time $O(f(d) \cdot n^{d/2+1})$ for constant factor
  - Random experiments: simple (Monte Carlo, slightly tweaked); advanced local search (Winzen, after Winkler, Fang)

5 dimensions, 50 points

max planck institut informatik

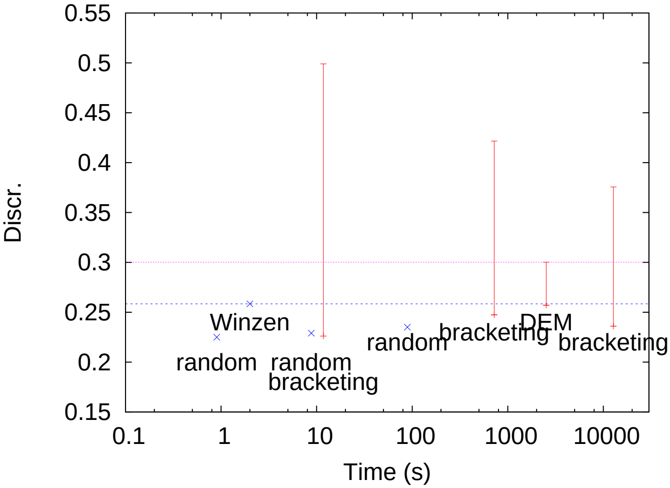


5 dimensions, 200 points



10 dimensions, 100 points

max planck institut informatik



Empirical conclusions

- Seen: All methods scale with d as $n^{O(d)}$
- No upper-bound method sufficient in practice
- Is a better algorithm possible?

Computational Complexity

Ultra-quick complexity review

max planck institut informatik

1. All problems are *decision* problems
   - Given P, x, is $d^*_\infty(P) \geq x$?
2. Running time as function of input size
   - Use nd for size, ignore encoding
3. Complexity classes:
   - P: problems decidable in time $n^{O(1)}$
   - NP: positive solutions verifiable in time $n^{O(1)}$ (deciding seems harder, e.g., $O(2^n)$ time)
4. Hardness by *reduction* from hard problem
   - NP-hard/NP-complete: reduction from every problem in NP

Ultra-quick complexity review

max planck institut informatik

1. All problems are *decision* problems
   - Given P, x, is $d^*_\infty(P) \geq x$?
2. Running time as function of input size
   - Use nd for size, ignore encoding
3. Complexity classes:
   - P: problems decidable in time $n^{O(1)}$
   - NP: positive solutions verifiable in time $n^{O(1)}$ (deciding seems harder, e.g., $O(2^n)$ time)
4. Hardness by *reduction* from hard problem
   - NP-hard/NP-complete: reduction from every problem in NP

Ultra-quick complexity review

max planck institut informatik

1. All problems are *decision* problems
   - Given P, x, is $d^*_\infty(P) \geq x$?
2. Running time as function of input size
   - Use nd for size, ignore encoding
3. Complexity classes:
   - P: problems decidable in time $n^{O(1)}$
   - NP: positive solutions verifiable in time $n^{O(1)}$ (deciding seems harder, e.g., $O(2^n)$ time)
4. Hardness by *reduction* from hard problem
   - NP-hard/NP-complete: reduction from every problem in NP

Ultra-quick complexity review

max planck institut informatik

1. All problems are *decision* problems
   - Given P, x, is $d^*_\infty(P) \geq x$?
2. Running time as function of input size
   - Use nd for size, ignore encoding
3. Complexity classes:
   - P: problems decidable in time $n^{O(1)}$
   - NP: positive solutions verifiable in time $n^{O(1)}$ (deciding seems harder, e.g., $O(2^n)$ time)
4. Hardness by *reduction* from hard problem
   - NP-hard/NP-complete: reduction from every problem in NP

Status of Star Discrepancy

- Problem: given P, x, is $d^*_\infty(P) \geq x$?
  - Fix d: in P (time $n^{O(d)}$)
  - Free d: NP-complete (Gnewuch et al.; uses $d = n$)
- But what if $d = 10$, $n = 1000$?
  - $O(n^{d/2+1})$: 10^{18} steps?
  - $O(2^d n^2)$: 10^9 steps?
- Need *parameterized* complexity to distinguish
- Will give evidence that $n^{O(d)}$ not improvable

Status of Star Discrepancy

max planck institut informatik

- Problem: given P, x, is $d^*_\infty(P) \geq x$?
  - Fix d: in P (time $n^{O(d)}$)
  - Free d: NP-complete (Gnewuch et al.; uses $d = n$)
- But what if $d = 10$, $n = 1000$?
  - $O(n^{d/2+1})$: 10^{18} steps?
  - $O(2^d n^2)$: 10^9 steps?
- Need *parameterized* complexity to distinguish
- Will give evidence that $n^{O(d)}$ not improvable

Status of Star Discrepancy

max planck institut informatik

- Problem: given P, x, is $d^*_\infty(P) \geq x$?
  - Fix d: in P (time $n^{O(d)}$)
  - Free d: NP-complete (Gnewuch et al.; uses $d = n$)
- But what if $d = 10$, $n = 1000$?
  - $O(n^{d/2+1})$: 10^{18} steps?
  - $O(2^d n^2)$: 10^9 steps?
- Need *parameterized* complexity to distinguish
- Will give evidence that $n^{O(d)}$ not improvable

Status of Star Discrepancy

- Problem: given P, x, is $d^*_\infty(P) \geq x$?
  - Fix d: in P (time $n^{O(d)}$)
  - Free d: NP-complete (Gnewuch et al.; uses $d = n$)
- But what if $d = 10$, $n = 1000$?
  - $O(n^{d/2+1})$: 10^{18} steps?
  - $O(2^d n^2)$: 10^9 steps?
- Need *parameterized* complexity to distinguish
- Will give evidence that $n^{O(d)}$ not improvable

Parameterized Complexity

max planck institut informatik

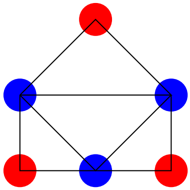
- Introduce *parameter* k (solution size, dimension, ...)
- FPT (*fixed-parameter tractable*): running time $O(f(k) \cdot n^c)$, some $f(k)$.
  - Vertex Cover: time $O(1.28^k n^c)$, solution size k
  - k-Path: find path of length $\geq k$ in time $O(2^k n^c)$
  - ...
- W[1]: Polynomial for fixed k, but degree rises (n^k-type behaviour)
  - Independent Set

Parameterized Complexity

- Introduce *parameter* k (solution size, dimension, ...)
- FPT (*fixed-parameter tractable*): running time $O(f(k) \cdot n^c)$, some $f(k)$.
  - Vertex Cover: time $O(1.28^k n^c)$, solution size k
  - k-Path: find path of length $\geq k$ in time $O(2^k n^c)$
  - ...
- W[1]: Polynomial for fixed k, but degree rises (n^k-type behaviour)
  - Independent Set

Independent Set

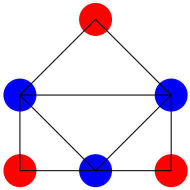
max planck institut informatik



- Problem Independent Set (IS)
  - Input: graph $G = (V, E)$, parameter k
  - Is there an independent set of k vertices?
- Defines FPT-hardness class W[1]
- All known algorithms $n^{\Theta(k)}$ (best: $O(n^{(2.376/3)k})$ time and space)
  - If time $f(k) \cdot n^{o(k)}$ is possible, then so is $f'(k) \cdot n^{O(1)}$
  - Time $f(k) \cdot n^{o(k)}$ for k-IS implies time $2^{o(n)}$ for 3-SAT, IS, ...
- Will reduce k-IS to $2k$-dim star discrepancy

Independent Set

max planck institut informatik



- Problem INDEPENDENT SET (IS)
  - Input: graph $G = (V, E)$, parameter k
  - Is there an independent set of k vertices?
- Defines FPT-hardness class W[1]
- All known algorithms $n^{\Theta(k)}$ (best: $O(n^{(2.376/3)k})$ time and space)
  - If time $f(k) \cdot n^{o(k)}$ is possible, then so is $f'(k) \cdot n^{O(1)}$
  - Time $f(k) \cdot n^{o(k)}$ for k-IS implies time $2^{o(n)}$ for 3-SAT, IS, ...
- Will reduce k-IS to $2k$-dim star discrepancy

Scaffolding

Making sense of the problem:

1. Focus on "largest empty anchored box" problem
2. *Scaffolding* forces solutions into useful discrete structure
   - Dimensions $2i-1$, $2i$ select the i:th vertex
   - Each selection has the same volume c
   - Constraint points to force compatible choices
3. Make sure worst-case box is an empty box

Scaffolding

max planck institut informatik

Making sense of the problem:

1. Focus on "largest empty anchored box" problem
2. *Scaffolding* forces solutions into useful discrete structure
   - Dimensions $2i-1$, $2i$ select the i:th vertex
   - Each selection has the same volume c
   - Constraint points to force compatible choices
3. Make sure worst-case box is an empty box

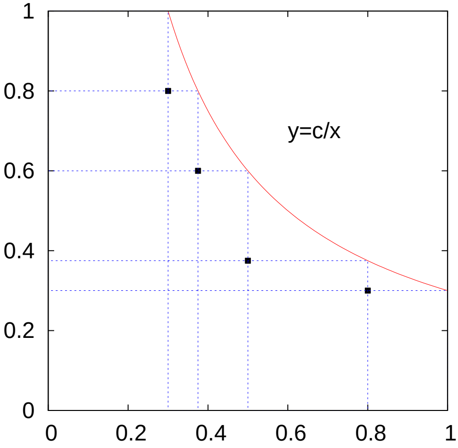
Scaffolding

max planck institut informatik

Making sense of the problem:

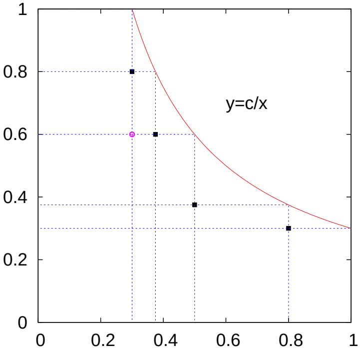
1. Focus on "largest empty anchored box" problem
2. *Scaffolding* forces solutions into useful discrete structure
   - Dimensions $2i-1$, $2i$ select the i:th vertex
   - Each selection has the same volume c
   - Constraint points to force compatible choices
3. Make sure worst-case box is an empty box

Scaffolding points



Constraint points

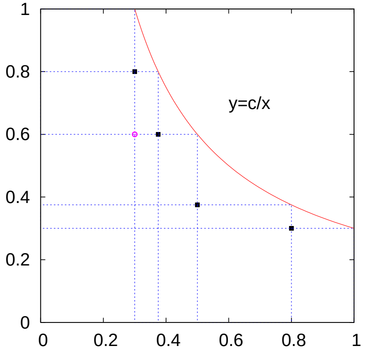
max planck institut
informatik

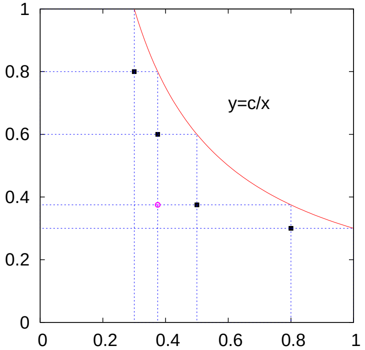


Selection 1: dim 1,2 Selection 2: dim 3,4

Pockets=vertex selection. Purple point: not both pockets.

Constraint points





Selection 1: dim 1,2 Selection 2: dim 3,4

Pockets=vertex selection. Purple point: not both pockets.

Construction, repeat

max planck institut informatik

- Pair $2k$ dimensions into k selections
- Place points in dimension pair $2i-1, 2i$ (other components zero) to create pockets for selection i
- Add constraint points to forbid repeated vertices
- Add constraint points to forbid combination i, j when $(u_i, u_j) \in E$
- "Pockets" give volume c, points outside of pockets volume $c' < c$

Construction, repeat

max planck institut informatik

- Pair $2k$ dimensions into k selections
- Place points in dimension pair $2i-1, 2i$ (other components zero) to create pockets for selection i
- Add constraint points to forbid repeated vertices
- Add constraint points to forbid combination i, j when $(u_i, u_j) \in E$
- "Pockets" give volume c, points outside of pockets volume $c' < c$

Finalizing

- If G has k-IS, we can find an empty box of volume c^k (k pockets selected)
- If not, largest volume is $c'c^{k-1}$, $c' < c$ (one dimension-pair outside of pocket)
- If $c^k > \frac{n-1}{n}$, then worst-case box is empty
- Conclusion: Exact computation of star discrepancy for d-dimensional point set as hard as k-IS with $k = d/2$

Finalizing

max planck institut informatik

- If G has k-IS, we can find an empty box of volume c^k (k pockets selected)
- If not, largest volume is $c'c^{k-1}$, $c' < c$ (one dimension-pair outside of pocket)
- If $c^k > \frac{n-1}{n}$, then worst-case box is empty
- Conclusion: Exact computation of star discrepancy for d-dimensional point set as hard as k-IS with $k = d/2$

Finalizing

max planck institut informatik

- If G has k-IS, we can find an empty box of volume c^k (k pockets selected)
- If not, largest volume is $c'c^{k-1}$, $c' < c$ (one dimension-pair outside of pocket)
- If $c^k > \frac{n-1}{n}$, then worst-case box is empty
- Conclusion: Exact computation of star discrepancy for d-dimensional point set as hard as k-IS with $k = d/2$

Conclusions

- Bounding the star discrepancy of a d-dimensional point set is empirically computationally difficult already for moderate dimension
- Even approximations require time $n^{d/2}$ or more (with known tools)
- Shown: exact computation requires $n^{O(d)}$ time (unless FPT=W[1]); even a dependency of $n^{d/3}$ would mean a breakthrough for Independent Set
- Open: Approximation hardness?

Conclusions

max planck institut informatik

- Bounding the star discrepancy of a d-dimensional point set is empirically computationally difficult already for moderate dimension
- Even approximations require time $n^{d/2}$ or more (with known tools)
- Shown: exact computation requires $n^{O(d)}$ time (unless FPT=W[1]); even a dependency of $n^{d/3}$ would mean a breakthrough for Independent Set
- Open: Approximation hardness?

Conclusions

max planck institut
informatik

- Bounding the star discrepancy of a d-dimensional point set is empirically computationally difficult already for moderate dimension
- Even approximations require time $n^{d/2}$ or more (with known tools)
- Shown: exact computation requires $n^{O(d)}$ time (unless FPT=W[1]); even a dependency of $n^{d/3}$ would mean a breakthrough for Independent Set
- Open: Approximation hardness?

Conclusions

max planck institut
informatik

- Bounding the star discrepancy of a d-dimensional point set is empirically computationally difficult already for moderate dimension
- Even approximations require time $n^{d/2}$ or more (with known tools)
- Shown: exact computation requires $n^{O(d)}$ time (unless FPT=W[1]); even a dependency of $n^{d/3}$ would mean a breakthrough for Independent Set
- Open: Approximation hardness?