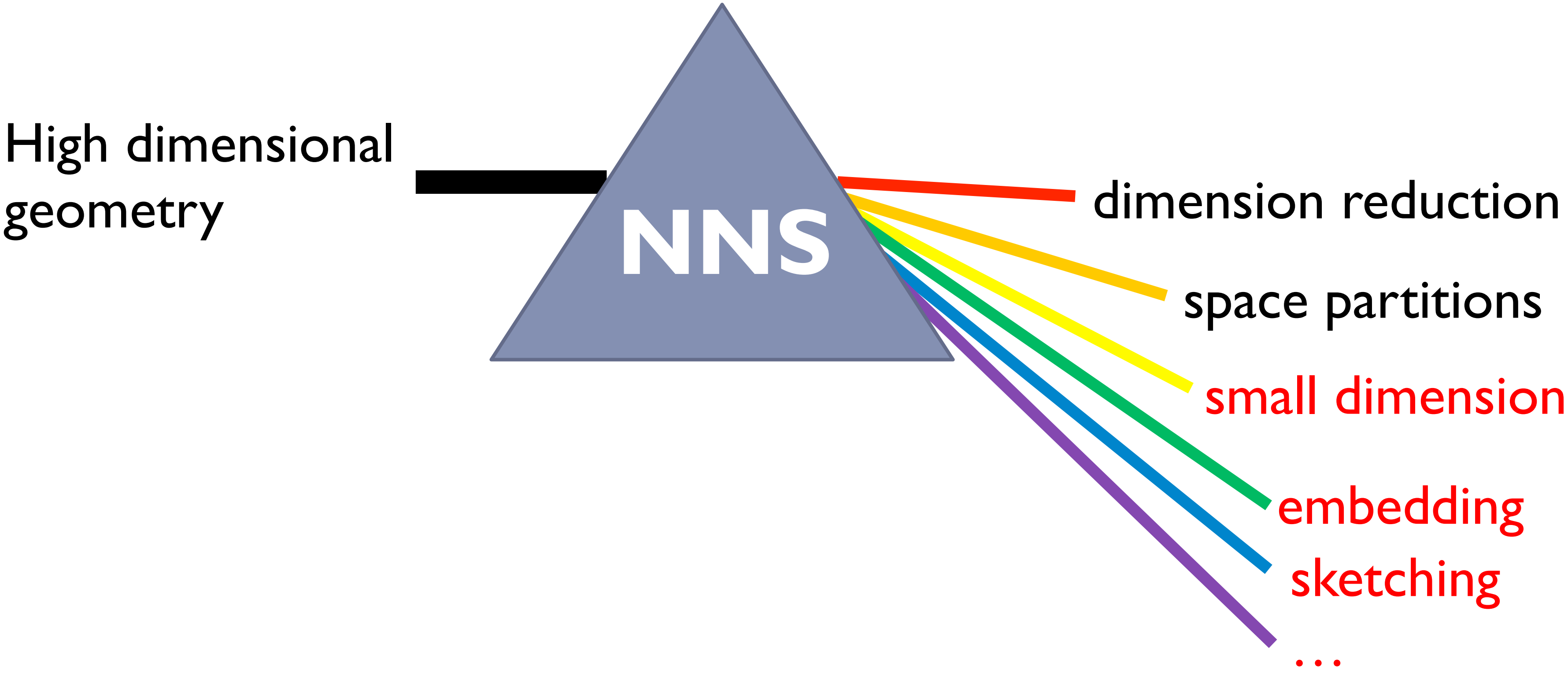
Algorithmic High-Dimensional Geometry 2

Alex Andoni

(Microsoft Research SVC)

The NNS prism



High dimensional geometry

NNS

dimension reduction

space partitions

small dimension

embedding

sketching

…

Small Dimension

What if d is small?

- Can solve $1+\epsilon$ approximate NNS with
  - $O(nd)$ space
  - $(O(d)/\epsilon)\uparrow d \log n$ query time
  - [AMNSW'98,…]
- OK, if say $d=5$!

- Usually, d is not small…

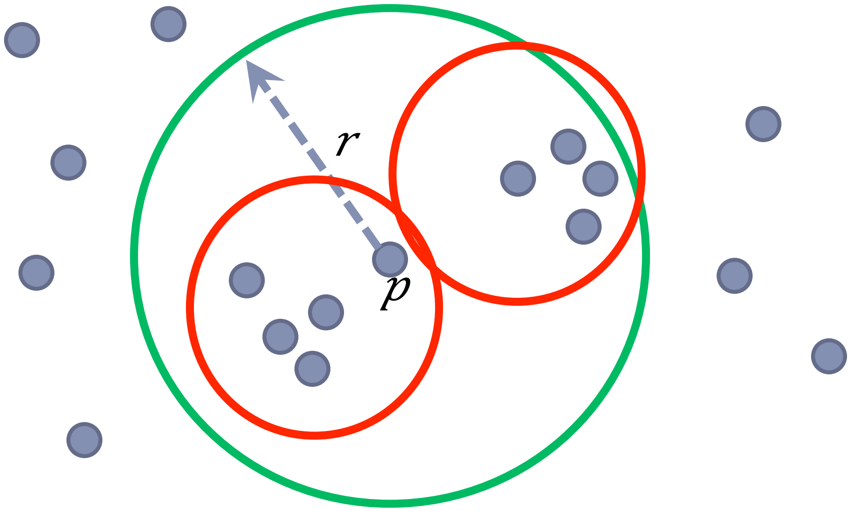
What if d is “effectively” small?

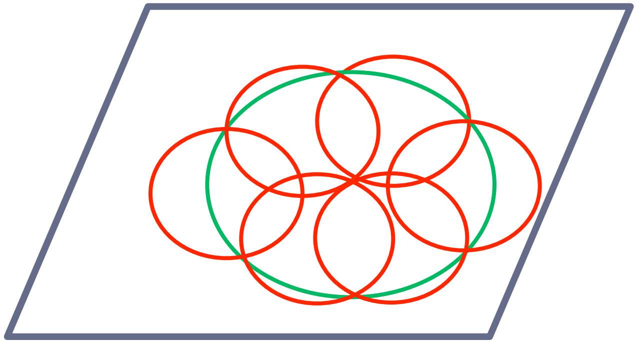
- Eg:
  - k-dimensional subspace of $\Re\uparrow d$, with $k \ll d$
  - Obviously, extract subspace and solve NNS there!
  - Not a robust definition…

- More robust definitions:
  - KR-dimension [KR’02]
  - Doubling dimension [Assouad’83, Cla’99, GKL’03, KL’04]
  - Smooth manifold [BW’06, Cla’08]
  - other [CNBYM’01, FK97, IN’07, Cla’06…]

Doubling dimension

- Definition: pointset S has *doubling dimension* λ if:
  - for any point $p \in S$, radius r, consider ball $B(p,r)$ of points within distance r of p
  - can cover $B(p,r)$ by 2^{λ} balls $B(y_1, r/2), B(y_2, r/2), \ldots$
- Sanity check:
  - k-dimensional subspace has $\lambda = O(k)$
  - n points always have dimension at most $O(\log n)$
- Can be defined for any metric space!





$2^{O(k)}$ balls to cover

NNS for small doubling dimension

- Euclidean space [Indyk-Naor'07]
  - JL into dimension $k=O(\lambda)$ "works" !
  - Contraction of *any pair* happens with very small probability
  - Expansion of *some pair* happens with constant probability
  - Good enough for NNS!

- Arbitrary metric
  - Navigating nets/cover trees [Krauthgamer-Lee'04, Har-Peled-Mendel'05, Beygelzimer-Kakade-Langford'06,…]
  - Algorithm:
    - A data-dependent tree: recursive space partition using balls $B(p,r)$
    - At query q, follow all paths that intersect with the ball $B(q,r)$

Embeddings

General Theory: embeddings

- General motivation: given distance (metric) M, solve a computational problem P under M

Hamming distance

Euclidean distance (ℓ_2)

Edit distance between two strings

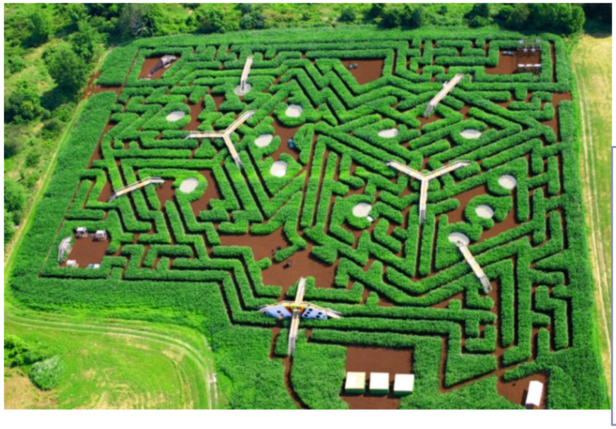
Earth-Mover (transportation) Distance

Compute distance between two points

Nearest Neighbor Search

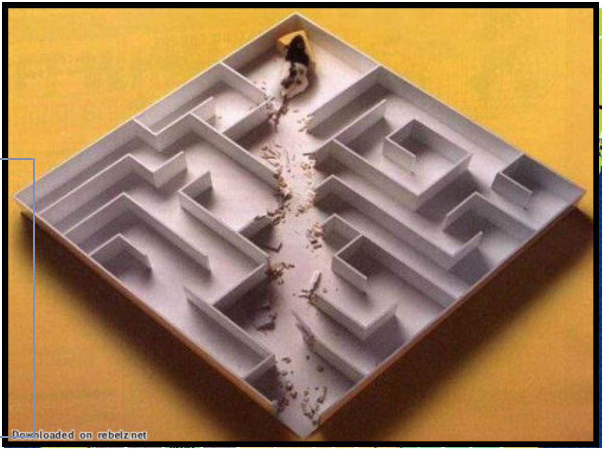
Diameter/Close-pair of set S

Clustering, MST, etc



$\xrightarrow{f}$

Reduce problem
<P under hard metric>
to
<P under simpler metric>



Embeddings: landscape

- Definition: an embedding is a map $f: M \to H$ of a metric (M, d_M) into a host metric (H, ρ_H) such that for any $x, y \in M$:

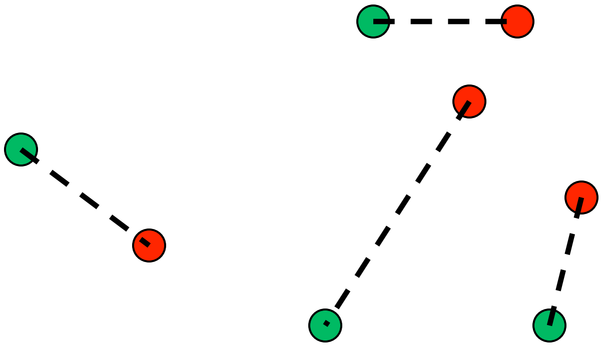
  $$d_M(x,y) \le \rho_H(f(x), f(y)) \le D \cdot d_M(x,y)$$

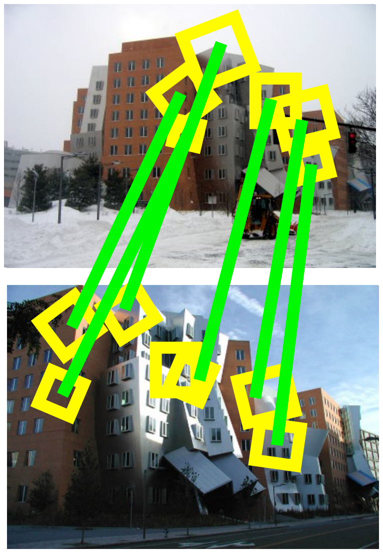
  where D is the distortion (approximation) of the embedding f.

- Embeddings come in all shapes and colors:
  - Source/host spaces M, H
  - Distortion D
  - Can be randomized: $\rho_H(f(x), f(y)) \approx d_M(x,y)$ with $1-\varepsilon$ probability
  - Time to compute $f(x)$
- Types of embeddings:
  - From norm to the same norm but of *lower dimension* (dimension reduction)
  - From one norm (ℓ_2) into another norm (ℓ_1)
  - From non-norms (edit distance, Earth-Mover Distance) into a norm (ℓ_1)
  - From given finite metric (shortest path on a planar graph) into a norm (ℓ_1)
  - H not a metric but a computational procedure ← sketches

Earth-Mover Distance

- Definition:
  - Given two sets *A*, *B* of points in a metric space
  - *EMD*(*A*,*B*) = min cost bipartite matching between *A* and *B*
- Which metric space?
  - Can be plane, ℓ_2 , ℓ_1 …
- Applications in image vision





Images courtesy of Kristen Grauman

Embedding EMD into ℓ_1

- At least as hard as ℓ_1
- Theorem [Cha02, IT03]: Can embed EMD over $[\Delta]^2$ into ℓ_1 with distortion $O(\log\Delta)$. Time to embed a set of s points: $O(s\log\Delta)$.

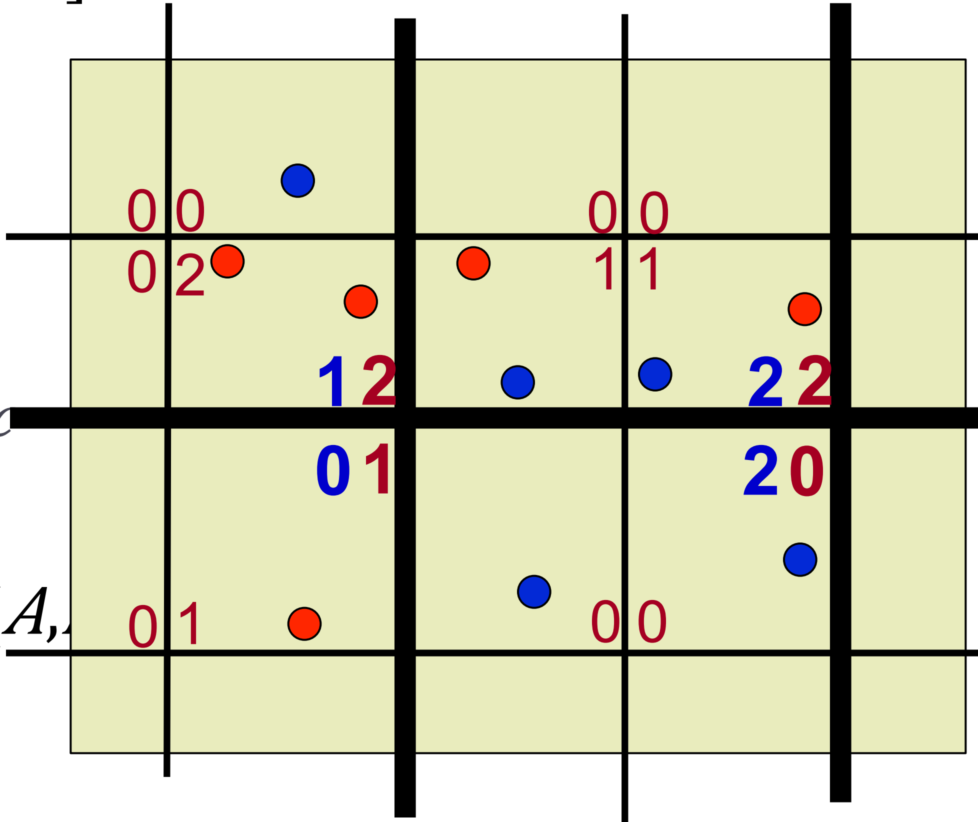
- Consequences:
  - Nearest Neighbor Search: $O(c\log\Delta)$ approximation with $O(sn^{1+1/c})$ space, and $O(n^{1/c}\cdot s\log\Delta)$ query time.
  - Computation: $O(\log\Delta)$ approximation in $O(s\log\Delta)$ time
    - Best known: $1+\epsilon$ approximation in $\tilde{O}(s)$ time [SA'12]
    - The higher-dimensional variant is still fastest via embedding [AIK'08]

High level embedding

- Sets of size s in $[1\ldots\Delta]\times[1\ldots\Delta]$ box
- Embedding of set A:
  - take a quad-tree
  - randomly shift it
  - Each cell gives a coordinate:

    $f(A)_c$=#points in the cell c
- Need to prove

$E[||f(A)-f(B)||{\downarrow}1\ ]\approx EMD(A,$

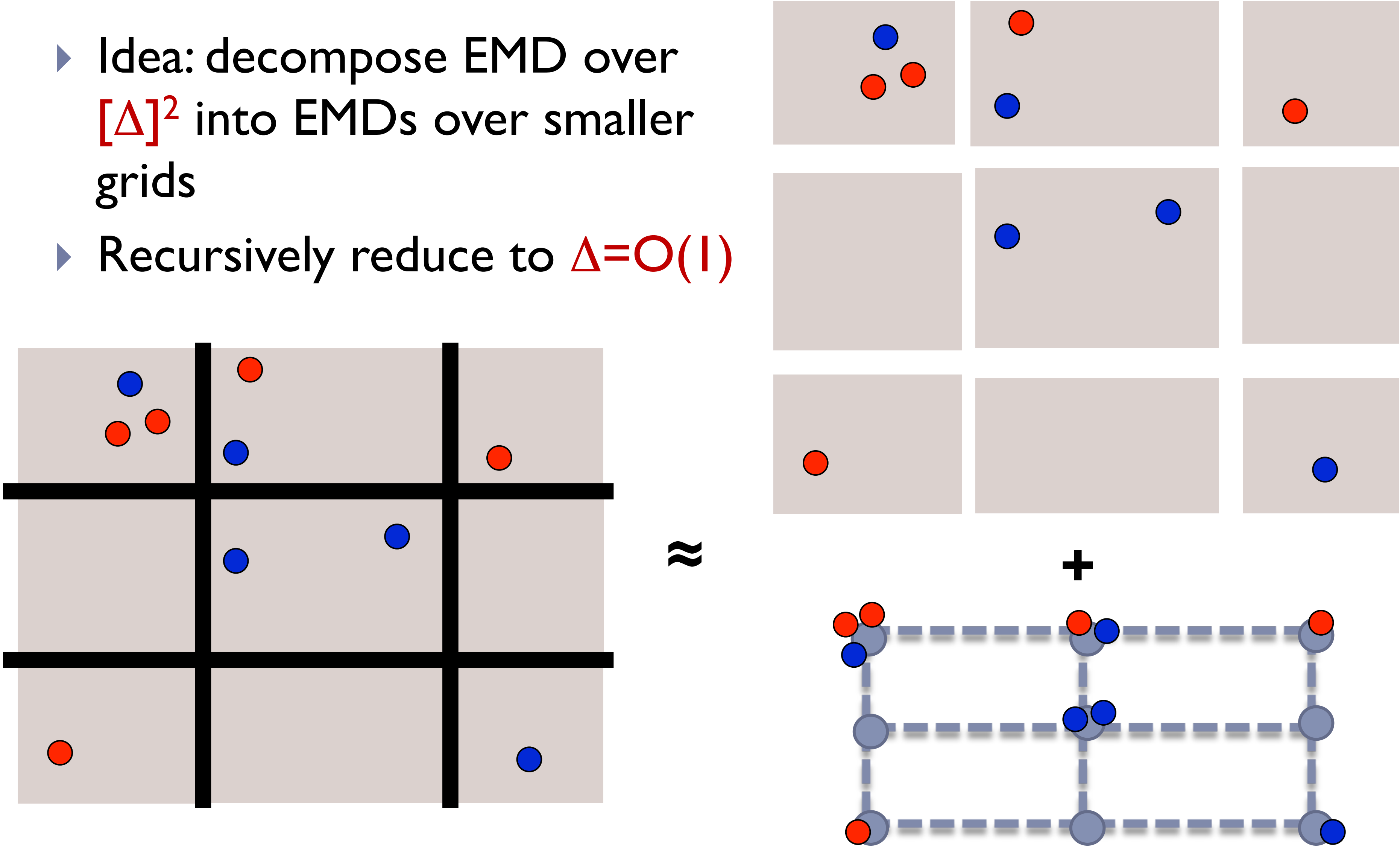


$f(\boldsymbol{A})$= …2210… 0002…0011…0100…0000…

$f(\boldsymbol{B})$= …1202… 0100…0011…0000…1100…

Main Approach

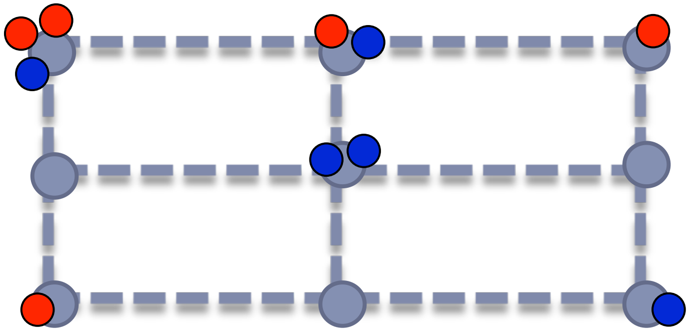
- Idea: decompose EMD over $[\Delta]^2$ into EMDs over smaller grids
- Recursively reduce to $\Delta=O(1)$



EMD over small grid

- Suppose $\Delta=3$

- f(A) has nine coordinates, counting # points in each joint
  - f(A)=(2,1,1,0,0,0,1,0,0)
  - f(B)=(1,1,0,0,2,0,0,0,1)
- Gives O(1) distortion

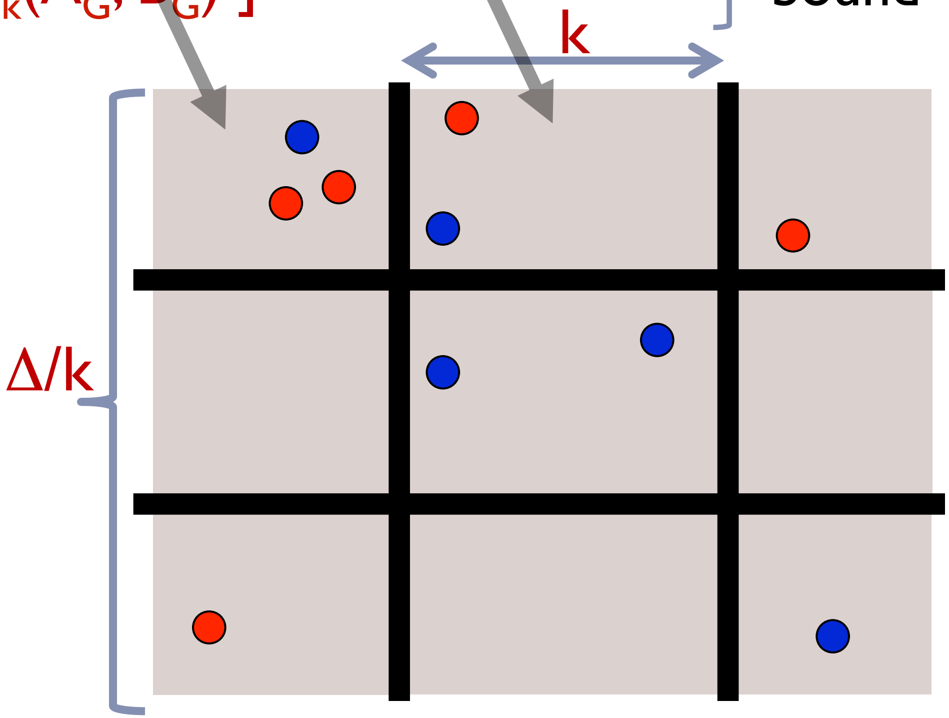


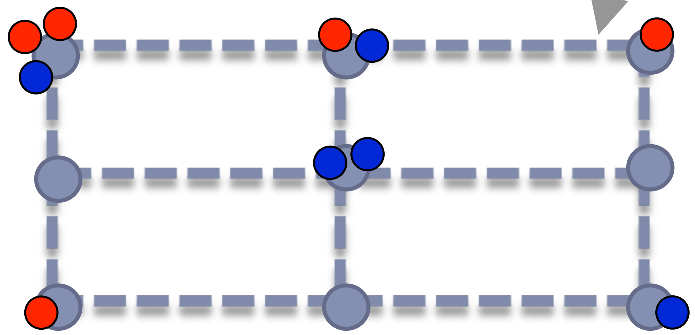
Decomposition Lemma [I07]

- For randomly-shifted cut-grid G of side length k, we have:
  - $\mathrm{EEMD}_{\Delta}(A,B) \le \boxed{\mathrm{EEMD}_{k}(A_1, B_1)} + \boxed{\mathrm{EEMD}_{k}(A_2,B_2)+}\dots + k{*}\boxed{\mathrm{EEMD}_{\Delta/k}(A_G, B_G)}$

    lower bound on cost
  - $\mathrm{EEMD}_{\Delta}(A,B) \ge 1/3\ \mathrm{E}[\ \mathrm{EEMD}_{k}(A_1, B_1) + \mathrm{EEMD}_{k}(A_2,B_2)+\dots\ ]$
  - $\mathrm{EEMD}_{\Delta}(A,B) \ge \mathrm{E}[\ k{*}\mathrm{EEMD}_{\Delta/k}(A_G, B_G)\ ]$

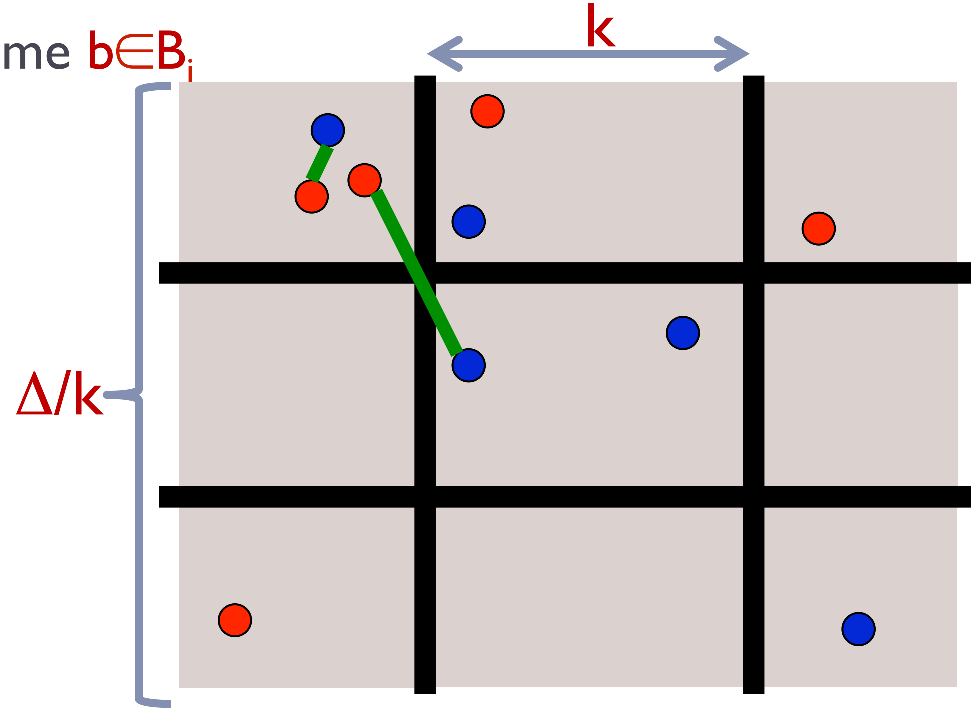
    upper bound
- The distortion will follow by applying the lemma recursively to (A_G,B_G)





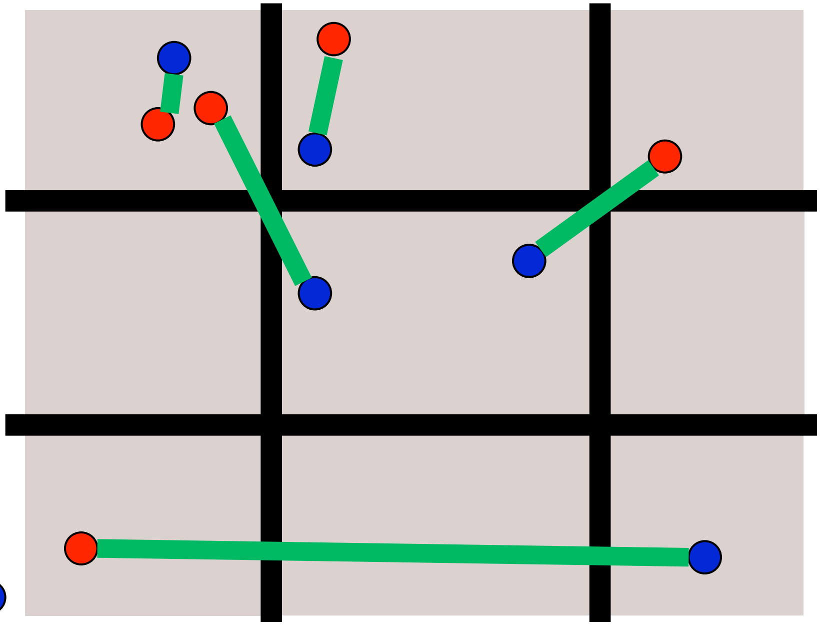
Part 1: lower bound

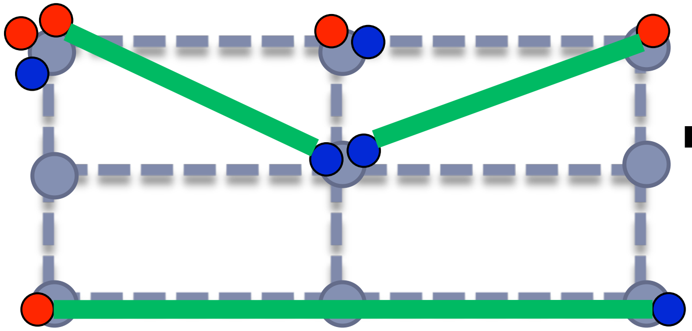
- For a randomly-shifted cut-grid G of side length k, we have:
  - $\mathrm{EEMD}_{\Delta}(A,B) \leq \mathrm{EEMD}_k(A_1, B_1) + \mathrm{EEMD}_k(A_2,B_2)+\ldots$
    $+ k*\mathrm{EEMD}_{\Delta/k}(A_G, B_G)$
- Extract a matching π from the matchings on right-hand side
- For each $a\in A$, with $a\in A_i$, it is either:
  - matched in $\mathrm{EEMD}(A_i,B_i)$ to some $b\in B_i$
  - or $a\in A_i\backslash B_i$, and it is matched in $\mathrm{EEMD}(A_G,B_G)$ to some $b\in B_j$
- Match cost in 2nd case:
  - Move a to center (Δ)
    - paid by $\mathrm{EEMD}(A_i,B_i)$
  - Move from cell i to cell j
    - paid by $\mathrm{EEMD}(A_G,B_G)$



Parts 2 & 3: upper bound

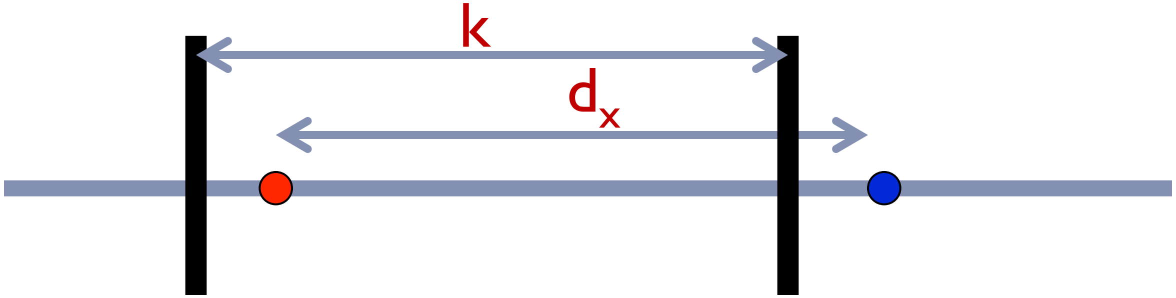
- For a randomly-shifted cut-grid G of side length k, we have:
  - $\mathrm{EEMD}_{\Delta}(A,B) \geq 1/3\ \mathrm{E}[\ \mathrm{EEMD}_{k}(A_1, B_1) + \mathrm{EEMD}_{k}(A_2,B_2)+\dots\ ]$
  - $\mathrm{EEMD}_{\Delta}(A,B) \geq \mathrm{E}[\ k^{*}\mathrm{EEMD}_{\Delta/k}(A_G, B_G)\ ]$
- Fix a matching π minimizing $\mathrm{EEMD}_{\Delta}(A,B)$
  - Will construct matchings for each EEMD on RHS
- Uncut pairs (a,b) are matched in respective (A_i,B_i)
- Cut pairs (a,b) are matched
  - in (A_G,B_G)
  - and remain unmatched in their mini-grids





Part 2: Cost?

- $\mathrm{EEMD}_{\Delta}(A,B) \geq 1/3\ \mathrm{E}[\ \sum_i \mathrm{EEMD}_k(A_i, B_i)]$
- Uncut pairs (a,b) are matched in respective (A_i,B_i)
  - Contribute a total ≤ $\mathrm{EEMD}\ _{\Delta}(A,B)$
- Consider a cut pair (a,b) at distance $a-b=(d\downarrow x\ ,d\downarrow y\ )$
  - Contribute ≤ 2k to $\sum_i \mathrm{EEMD}_k(A_i, B_i)$
  - Pr[(a,b) cut] = $1-(1-d\downarrow x\ /k)(1-d\downarrow y\ /k)\leq ||a-b||\downarrow 1\ /k$
  - Expected contribution ≤ Pr[(a,b) cut] $\cdot 2k \leq 2||a-b||1$
  - In total, contribute $2\cdot\mathrm{EEMD}\ _{\Delta}(A,B)$



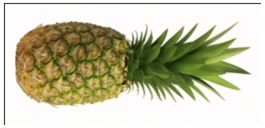
Wrap-up of EMD Embedding

- In the end, obtain that
  - EMD(A,B) ≈ sum of EMDs of smaller grids in expectation
  - Repeat $O(\log\Delta)$ times to get to 1×1 grid
  - $O(\log\Delta)$ approximation it total!

Embeddings of various metrics into ℓ↓1

Metric	Upper bound
Earth-mover distance (-sized sets in 2D plane)	[Cha02, IT03]
Earth-mover distance (-sized sets in)	[AIK08]
Edit distance over (#indels to transform x->y)	[OR05]
Ulam (edit distance between permutations)	[CK06]
Block edit distance	[MS00, CM07]

edit(,

) = 2

edit(1234567,
7123456) = 2

Non-embeddability into ℓ↓1

Metric	Upper bound	Lower bounds
Earth-mover distance (-sized sets in 2D plane)	[Cha02, IT03]	[NS07]
Earth-mover distance (-sized sets in)	[AIK08]	[KN05]
Edit distance over (#indels to transform x->y)	[OR05]	[KN05,KR06]
Ulam (edit distance between permutations)	[CK06]	[AK07]
Block edit distance	[MS00, CM07]	4/3 [Cor03]

Non-embeddability proofs

- Via *Poincaré-type inequalities*…
- [Enflo'69]: embedding $\{0,1\}^d$ into ℓ_2 (any dimension) must incur $\Omega(\sqrt{d})$ distortion
- Proof [Khot-Naor'05]
  - Suppose f is the embedding of $\{0,1\}^d$ into ℓ_2
  - Two distributions over pairs of points $x,y \in \{0,1\}^d$:
    - C: $x=y+e_i$ for random y and index i
    - F: x,y are random
  - Two steps:
    - $E_C [||x-y||_1] \leq O(1/d) \cdot E_F [||x-y||_1]$
    - $E_C [||f(x)-f(y)||_2^2] \geq \Omega(1/d) \cdot E_F [||f(x)-f(y)||_2^2]$ (short diagonals)
  - Implies $\Omega(\sqrt{d})$ lower bound!

Other good host spaces?

- What is “good”:
  - is algorithmically tractable
  - is rich (can embed into it)

	sq- ,etc
✓	✓
✗	✗

???

sq-ℓ_2=real space with distance: $||x-y||_2^2$

Metric	Lower bound into
Edit distance over	[KN05, KR06]
Ulam (edit distance between permutations)	[AK07]
Earth-mover distance (-sized sets in)	[KN05]

sq-ℓ_2, hosts with very good LSH (lower bounds via communication complexity)

[AK’07]

[AK’07]

[AIK’08]

The ℓ_∞ story

- [Mat96]: Can embed any metric on n points into ℓ_∞^n

- Theorem [I'98]: NNS for ℓ_∞^d with
  - $O_\delta(\log\log d)$ approximation
  - $n^{1+\delta}$ space, $\delta>0$
  - $O(d\log n)$ query time

- Dimension n is an issue though…
- Smaller dimension?
  - Possible for some: Hausdorff,… [FCI99]
- But, not possible even for $\{0,1\}^d$ [JL01]

Other good host spaces?

- What is "good":
  - algorithmically tractable
  - rich (can embed into it)

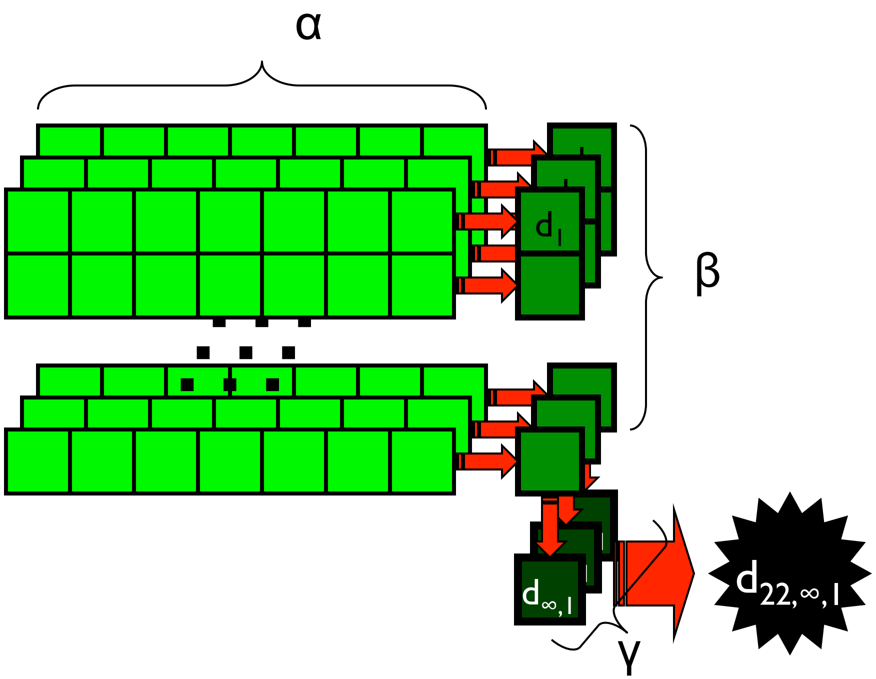
	sq- ,etc		
✓	✓	✓	✗
✗	✗	✗	✓

- But: combination sometimes works!

Meet our new host

[A-Indyk-Krauthgamer'09]

- Iterated product space



$$\text{sq-}\ell_2^{\gamma}\left(\ell_\infty^{\beta}\left(\ell_1^{\alpha}\right)\right)$$

$$x=(x_1,\dots,x_\alpha)\in R^\alpha$$
$$d_1(x,y)=\sum_{i=1}^{\alpha}|x_i-y_i|$$

$$x=(x_1,\dots,x_\beta)\in\ell_1^\alpha\times\ell_1^\alpha\times\dots\ell_1^\alpha$$
$$d_{\infty,1}(x,y)=\max_{i=1..\beta} d_1(x_i,y_i)$$

$$x=(x_1,\dots,x_\gamma)\in\ell_\infty^\beta(\ell_1^\alpha)\times\dots\times\ell_\infty^\beta(\ell_1^\alpha)$$
$$d_{22,\infty,1}(x,y)=\sum_{i=1}^{\gamma}\left(d_{\infty,1}(x_i,y_i)\right)^2$$

Why $sq-\ell_2^\gamma(\ell_\infty^\beta(\ell_1^\alpha))$?

[Indyk'02, A-Indyk-Krauthgamer'09]

edit distance between permutations
ED(1234567,
7123456) = 2

- Because we can…

Rich

- Embedding: …embed Ulam into $sq-\ell_2^\gamma(\ell_\infty^\beta(\ell_1^\alpha))$ with *constant* distortion
  - dimensions = length of the string

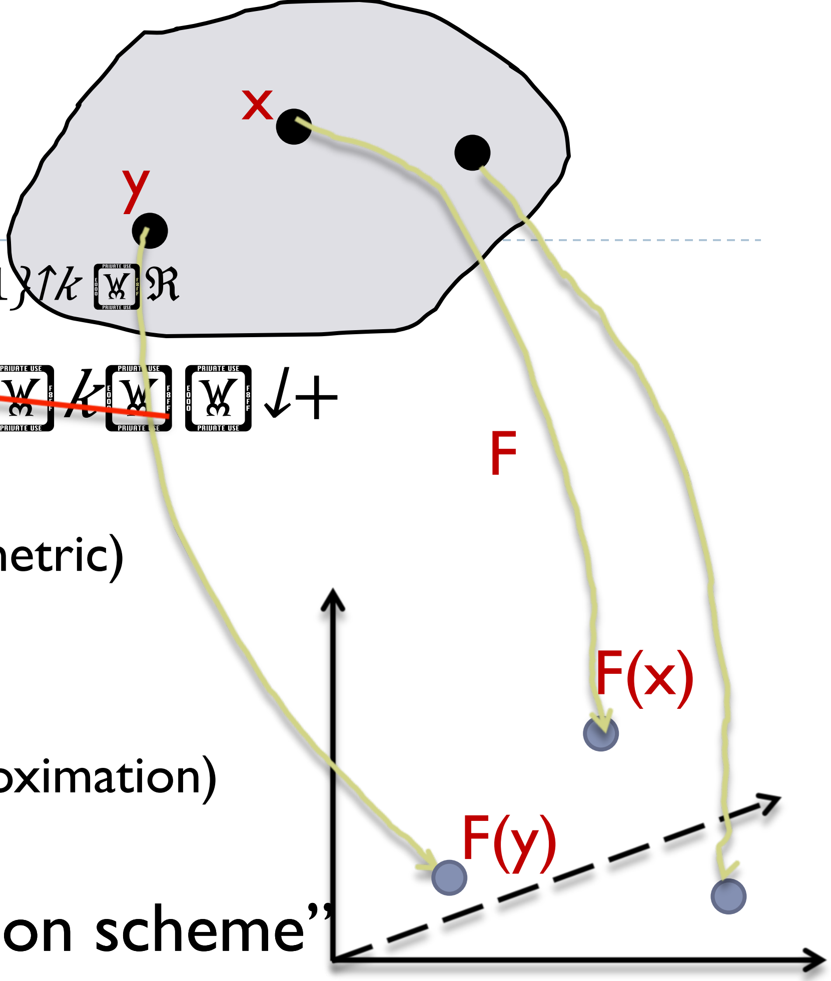
Algorithmically tractable

- NNS: Any t-iterated product space has NNS on n points with
  - $(\log\log n)^{O(t)}$ approximation
  - near-linear space and sublinear time
- Corollary: NNS for Ulam with $O(\log\log n)^2$ approx.
  - Better than via each ℓ_p component separately!

Sketching

Computational view

- F:M □ ~~ℜ↑k~~ {0,1}↑k {0,1}↑k × {0,1}↑k □ ℜ
- Arbitrary computation C: ~~□ $k x$ □ k □~~ □ ↓+
  - Cons:
    - No/little structure (e.g., (F,C) not metric)
  - Pros:
    - More expressability:
    - may achieve better distortion (approximation)
    - smaller "dimension" k
- Sketch F :"functional compression scheme"
  - for estimating distances
  - almost all lossy (1+□ distortion or more) and randomized



d↓M (x,y)≈√∑

$C(F(x), F(y))$

Why?

- 1) Beyond embeddings:
  - can more do if "embed" into computational space
- 2) A waypoint to get embeddings:
  - computational perspective can give actual embeddings
- 3) Connection to informational/computational notions
  - communication complexity

Beyond Embeddings:

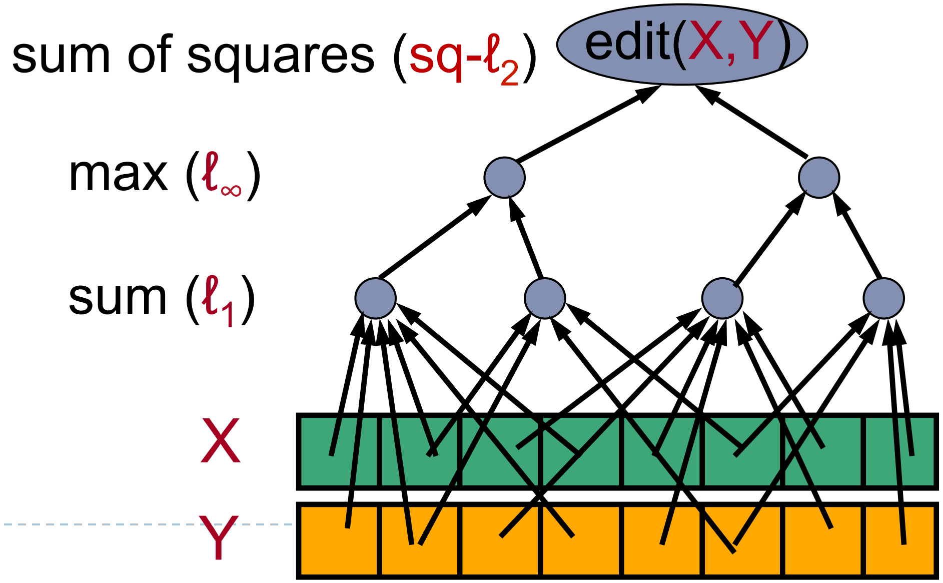
- “Dimension reduction” in ℓ_1 !
- Lemma [I00]: exists linear $F:\ell_1 \to \Re^k$, and C
  - where $k=O(\epsilon^{-2}\cdot\log n)$
  - achieves: for any $x,y \in \ell_1$, with probability $1-1/n^2$:
    - $C(F(x),F(y)) = (1\pm\epsilon)\cdot \|x-y\|_1$
- $F(x) = (s_1\cdot x, s_2\cdot x, \ldots s_k\cdot x)/k = 1/k\cdot Sx$
  - Where $s_i=(s_{i1},s_{i2},\ldots s_{id})$ with each s_{ij} distributed from Cauchy distribution (1-stable distribution)
  - $C(F(x),F(y)) = median(|F_1(x)-F_1(y)|,$ $|F_2(x)-F_2(y)|,$ … $|F_k(x)-F_k(y)|)$

$pdf(s)=1/\pi(s^2+1)$

- While $|s\cdot x|$ does not have expectation, it has median!

Waypoint to get embeddings

- Embedding of Ulam metric into $sq-\ell_2^{\gamma}(\ell_\infty^{\beta}(\ell_1^{\alpha}))$ was obtained via "geometrization" of an algorithm/characterization:
  - *sublinear (local) algorithms:* property testing & streaming [EKKRV98, ACCL04, GJKK07, GG07, EJ08]



sum of squares (sq-ℓ_2)

edit(X,Y)

max (ℓ_∞)

sum (ℓ_1)

X

Y

Ulam: algorithmic characterization

[Ailon-Chazelle-Commandur-Lu'04, Gopalan-Jayram-Krauthgamer-Kumar'07, A-Indyk-Krauthgamer'09]

- **Lemma:** Ulam(x,y) approximately equals the number of "faulty" characters a satisfying:
  - there exists K≥1 (prefix-length) s.t.
  - the set of K characters preceding a in x differs much from

    the set of K characters preceding a in y

E.g., a=5; K=4

$$x = \boxed{\overbrace{1234}^{X[5;4]}56789}$$

$$y = \boxed{1234\underbrace{6789}_{Y[5;4]}5}$$

$$|X[a;K] \Delta Y[a;K]| > K$$

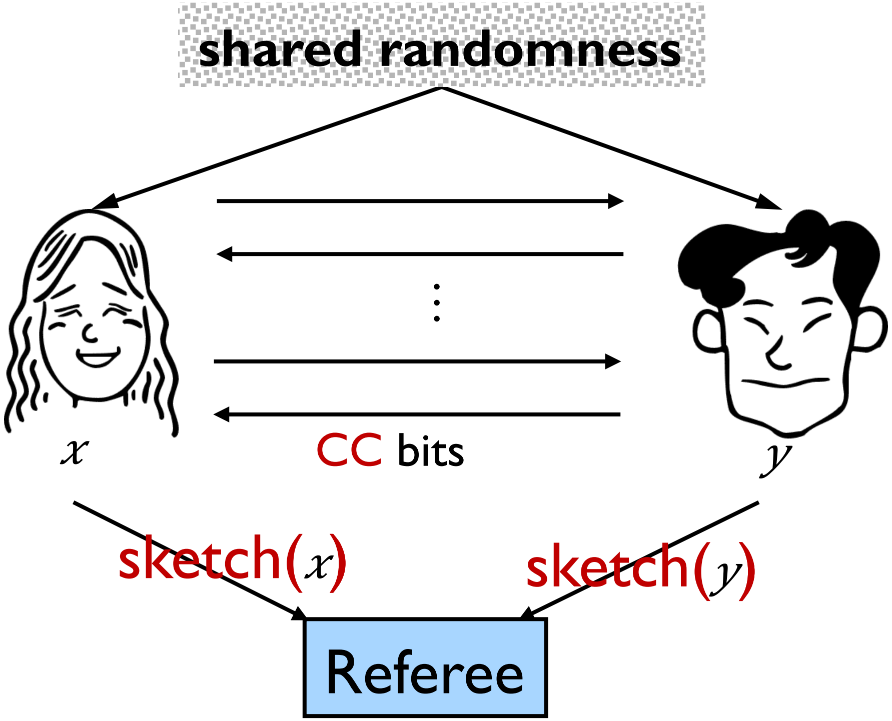
$$\Updownarrow$$

$$\left\| 1_{X[a;K]} - 1_{Y[a;K]} \right\|_1 > K$$

E.g. $1_{X[5;4]} = (1,1,1,1,0,0,0,0,0)$

Connection to communication complexity

- Enter the world of Alice and Bob…



shared randomness

x

CC bits

y

sketch(x)

sketch(y)

Referee

decide whether:

$d(x,y) \le R$ or $d(x,y) > cR$

Communication complexity model:

- Two-party protocol
- Shared randomness
- Promise (gap) version
- c = approximation ratio
- CC = min. # bits to decide (for 90% success)

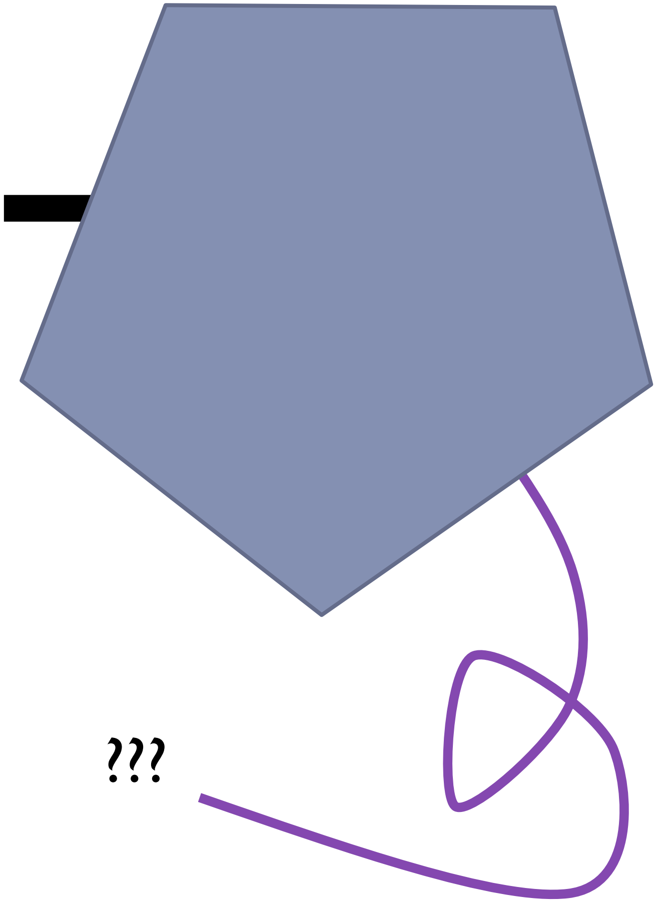
Sketching model:

- Referee decides based on sketch(x), sketch(y)
- SK = min. sketch size to decide

Fact: SK ≥ CC

Communication Complexity

- VERY rich theory [Yao'79, KN'97,…]
- Some notable examples:
  - ℓ_1, ℓ_2 are sketchable with $O(1/\epsilon^2)$ bits! [AMS'96,KOR'98]
  - hence also everything than embeds into it!
  - $\Omega(1/\epsilon^2)$ is tight [IW'03,W'04, BJKS'08,CR'12]
  - ℓ_∞^d requires $\Omega(d/c^2)$ bits [BJKS'02]
  - Coresets: sketches of *sets* of points for geometric problems [AHV04…]
- Connection to NNS:
  - [KOR'98]: if sketch size is s, then NNS with $n^{O(s)}$ space and one memory lookup!
  - From the perspective of NNS lower bounds, communication complexity closer to ground truth
- Question: do non-embeddability result say something about non-sketchability?
  - also Poincaré-type inequalities… [AK07,AJP'10]
- Connections to *streaming*: see Graham Cormode's lecture

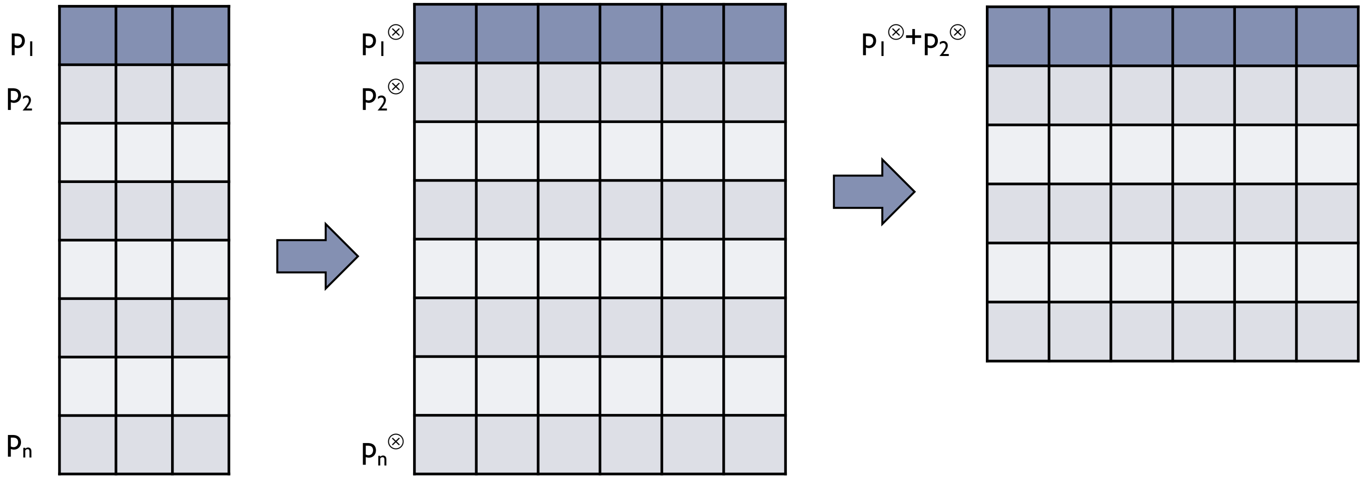


High dimensional geometry

???

Closest Pair

- Problem: n points in d-dimensional Hamming space, which are *random* except a planted pair at distance ½-ε
- Solution 1: build NNS and query n times
  - LSH-type algo would give $\sim dn\uparrow 2-\Theta(\epsilon)$ [PRR89,IM98,D08]
- Theorem [Valiant'12]: $O(dn\uparrow 1.8\ / poly($[illegible]$))$ time



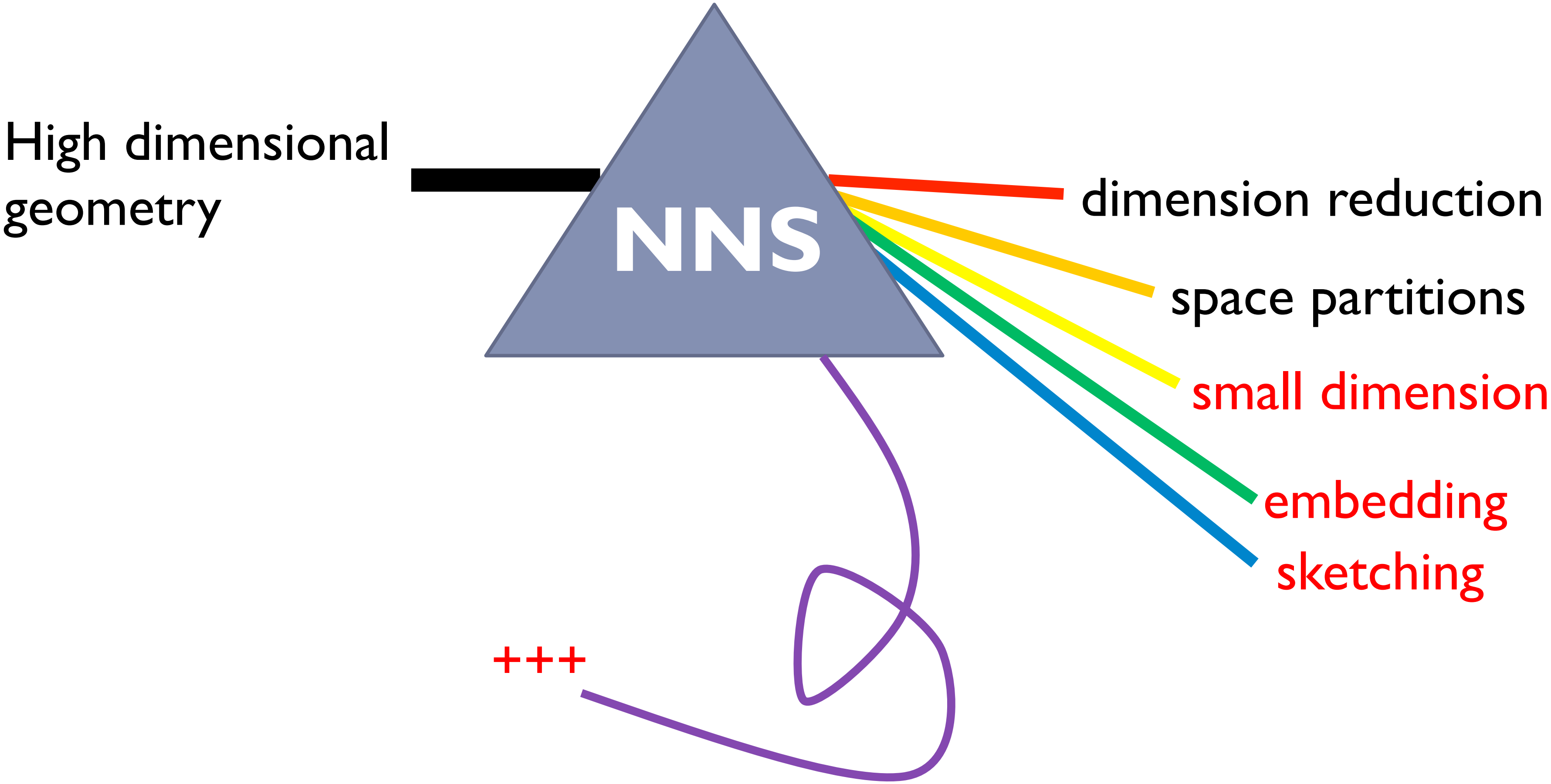
= M

Find max entry of MM^t using subcubic MM algorithms

What I didn't talk about:

- Too many things to mention
  - Includes embedding of fixed finite metric into simpler/more-structured spaces like ℓ_1
- Tiny sample among them:
  - [LLR94]: introduced metric embeddings to TCS. E.g. showed can use [Bou85] to solve sparsest cut problem with $O(\log n)$ approximation
  - [Bou85]: Arbitrary metric on n points into ℓ_1, with $O(\log n)$ distortion
  - [Rao99]: embedding planar graphs into ℓ_1, with $O(\sqrt{\log n})$ distortion
  - [ARV04,ALN05]: sparsest cut problem with $O(\sqrt{\log n})$ approximation
  - [KMS98,…]: space partition for rounding SDPs for coloring
  - Lots others…
- A list of open questions in embedding theory
  - Edited by Jiří Matoušek + Assaf Naor:
    - http://kam.mff.cuni.cz/~matousek/metrop.ps

High dimensional geometry via NNS prism



High dimensional geometry

NNS

dimension reduction

space partitions

small dimension

embedding

sketching

+++