Transitive-Closure Spanners

Sofya Raskhodnikova

Penn State University

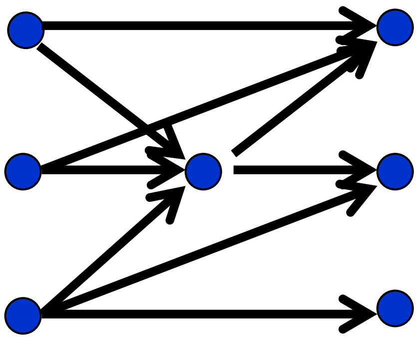
Based on joint work with

Arnab Bhattacharyya MIT
Elena Grigorescu MIT
Kyomin Jung MIT
David Woodruff IBM *Almaden*

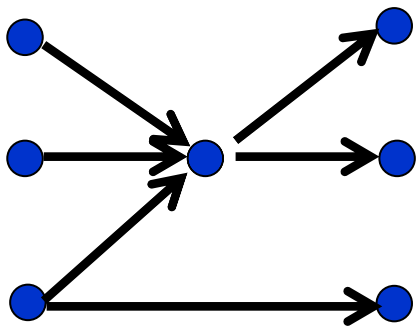
and with Madhav Jha Penn State

Graph Spanners [Awerbuch85,Peleg Schäffer89]

A subgraph *H* of *G* is a *k*-spanner
if for all pairs of vertices *u*, *v* in *G*,
$distance_H(u,v) \le k\ distance_G(u,v)$



dense graph G



sparse subgraph H

Goal: to find a sparsest *k*-spanner

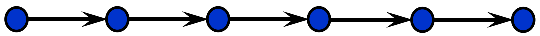
Applications:

- efficient routing
- protocols in unsynchronized networks
- parallel /distributed algorithms for approximate shortest paths

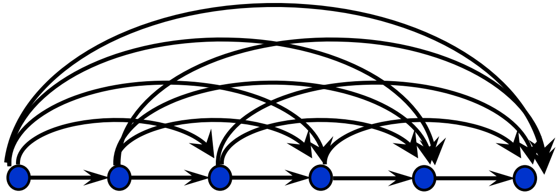
Transitive-Closure Spanners

Transitive closure TC(*G*) has an edge from *u* to *v* iff

G has a path from *u* to *v*



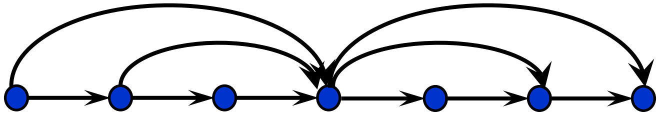
G



TC(*G*)

k-TC-spanner *H* of *G* has $distance_H(u,v) \leq k$ iff

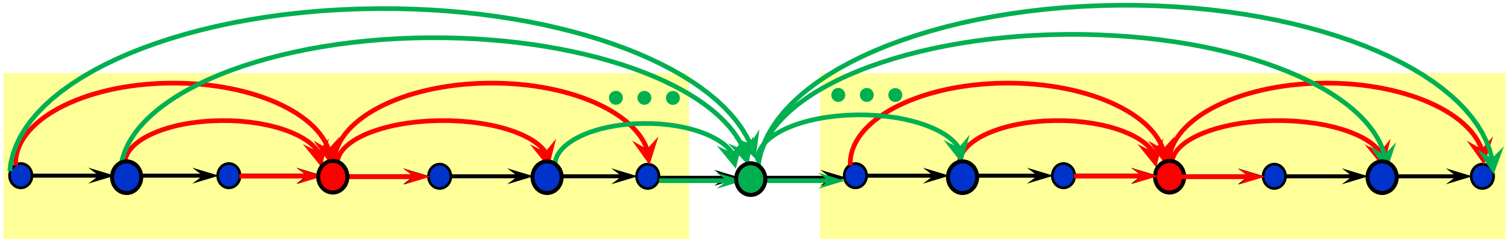
G has a path from *u* to *v*



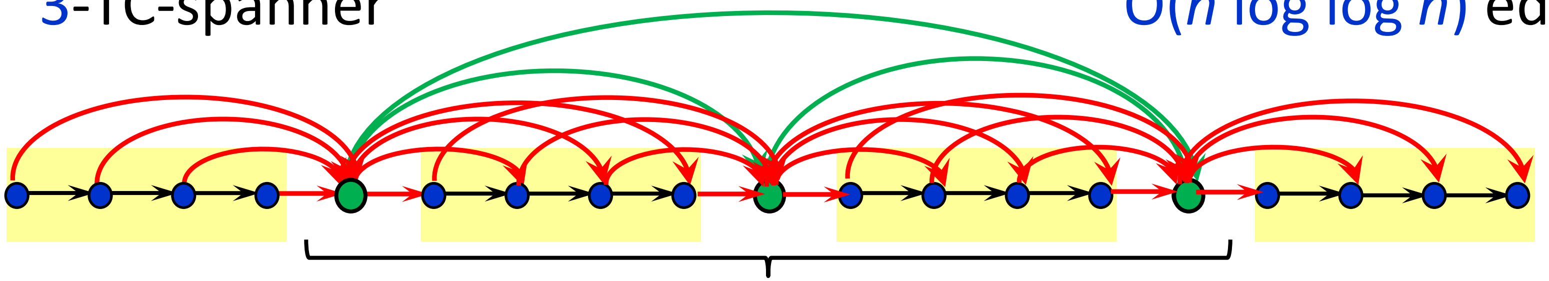
Alternatively: *k*-TC-spanner of *G* is a *k*-spanner of TC(*G*)

Example: Directed Line on n Vertices

- 2-TC-spanner — $\leq n \log n$ edges



- 3-TC-spanner — $O(n \log \log n)$ edges



$n^{1/2}$ hubs

- 4-TC-spanner — $O(n \log^* n)$ edges
- k-TC-spanner — $O(n\, \lambda(k,n))$ edges
  - Add a (k-2)-TC-spanner on hubs
  - Connect each node to the hubs before and after
  - Recurse on the fragments between hubs

Previous work

Structural results on TC-spanners

(what is a sparsest k-TC-spanner for a given graph family?)

- Shortcut graphs (special case when $|E(H)| \leq 2\,|E(G)|$)
  [Thorup 92, 95, Hesse 03]
- For directed line/trees [Thorup 97]
  implicit in
  - data structures [Yao 82, Alon Schieber 87, Chazelle 87, Bodlaender Tel Santoro 94]
  - property testing
    [Dodis Goldreich Lehman Raskhodnikova Ron Samorodnitsky 99]
  - access control [Attalah Frikken Blanton 05]
- *new* For low-dimensional posets
  - access control [Attalah Blanton Fazio Frikken 09]

Computational results on directed spanners

(given a graph, compute a sparsest k-spanner)

- $O(\log n)$-approximation algorithm for $k=2$ [Kortsarz Peleg 94]
- $O(n^{2/3}\,\mathrm{polylog}\, n)$-approximation for $k=3$ [Elkin Peleg 99]

Our Contributions

- Common abstraction for several applications [BGJRW09]
  - property testing
    - testing monotonicity of functions
    - testing if a function f:{1,…,n} $\rightarrow \mathbb{R}$ is Lipschitz [Jha R]
  - access control
  - data structures
- Structural results on TC-spanners
  - path-separable graphs [BGJRW09]
  - directed hypercube/hypergrid [BGJRW]
  - low-dimensional posets [BGJRW]
- Computational results on directed spanners

  k-TC-SPANNER: Given a graph, compute a sparsest k-TC-spanner
  - new algorithms, inapproximability results [BGJRW09]

Application 1: Testing if a List is Sorted

- Is a list of n numbers sorted?

  Requires reading entire list.

- Is a list of n numbers sorted or ϵ-far from sorted?

  (An ϵ fraction of list entries have to be changed to make it sorted.)

  [Ergün Kannan Kumar Rubinfeld Viswanathan 98]: $O((\log n)/\epsilon)$ time

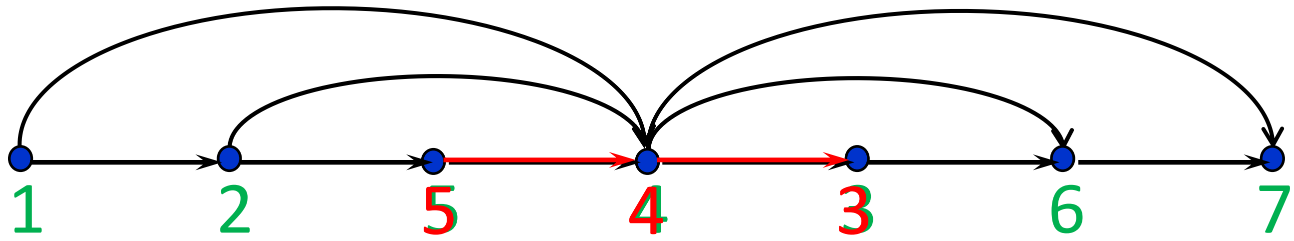
  [Fischer 01]: $\Omega(\log n)$ queries

- Special case of testing monotonicity of functions defined in [Goldreich Goldwasser Lehman Ron 98]

Is a list sorted or ϵ-far from sorted?

[Dodis Goldreich Lehman Raskhodnikova Ron Samorodnitsky 99]

Test can be viewed as: Pick a random edge from sparsest 2-TC-spanner for the line and compare its endpoints. Reject if they are out of order.



Claim 1. There are $\leq n \log n$ edges in the 2-TC-spanner.

Claim 2. Green numbers are sorted.

Proof: Any two green numbers are connected by a length-2 path of black edges

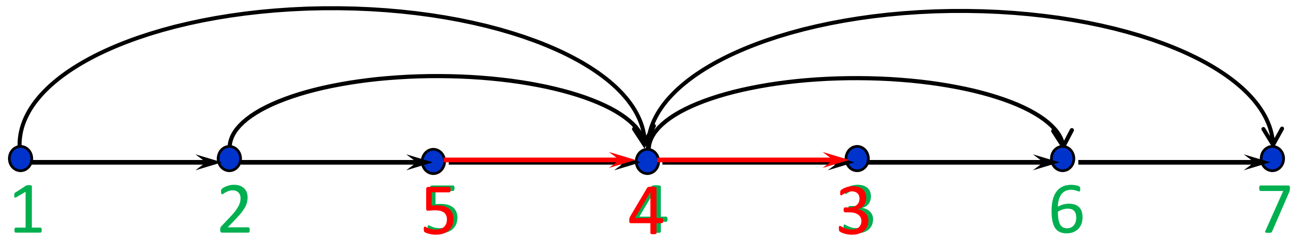
Analysis of the test:

- All sorted lists are accepted.
- If a list is ϵ-far from sorted, it has $\geq \epsilon n$ red numbers, $\Rightarrow$ $\geq \epsilon n/2$ red edges
  - If $\Theta((\log n)/\epsilon)$ edges are checked, a red edge will be discovered w.p. $\geq 2/3$

Is a list sorted or ϵ-far from sorted?

[Dodis Goldreich Lehman Raskhodnikova Ron Samorodnitsky 99]

Test can be viewed as: Pick a random edge from sparsest 2-TC-spanner for the line and compare its endpoints. Reject if they are out of order.



Claim 1. There are $\leq n \log n$ edges in the 2-TC-spanner.

Claim 2. Green numbers are sorted.

Conclusion: It suffices to check $O((\log n)/\epsilon)$ random edges from 2-TC-spanner.

Observation:

The same test/analysis apply to any property of a list of numbers if

- it can be expressed in terms of pairs of numbers
- it is transitive: (x,y) and (y,z) are *good* $\Rightarrow$ (x,z) is *good*

Generalization of Sortedness to Matrices

3	4	5	6
3	4	5	6
2	3	4	5
1	2	3	4

- Is a matrix sorted along all rows and columns or far from sorted (many numbers have to be changed to make it sorted)?
- [Goldreich Goldwasser Lehman Ron Samorodnitsky 00, Batu Rubinfeld White 99, Dodis Goldreich Lehman Raskhodnikova Ron Samorodnitsky 99, Halevy Kushilevitz 04] considered this problem for d-dimensional matrices.
- Is a function $f : \{1,\ldots,m\}^d \to R$ monotone or ϵ-far from monotone?

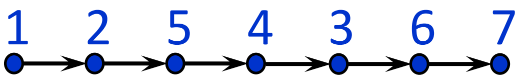
  [DGLRRS99]: $O(d \cdot (\log m) \cdot (\log |R|) / \epsilon)$ time

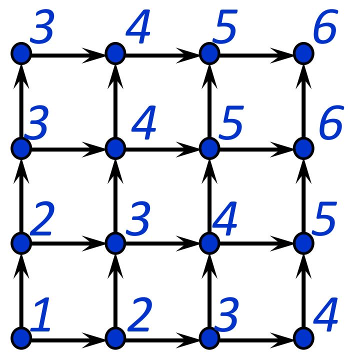
  [Fischer Lehman Newman Raskhodnikova Rubinfeld Samorodnitsky 02]:

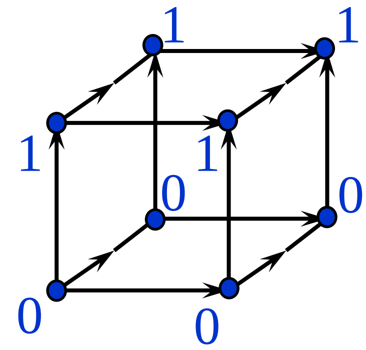
  for $m = 2$ and $R = \{0,1\}$, need $\Omega(\log \log d)$ queries

Monotonicity of Functions Over PO domains

[FLNRRS 02]:







Graph = partially ordered domain; node labels = values of the function

- A function is monotone if there are no violated edges (along which labels decrease):

- A function is ϵ-far from monotone if $\geq \epsilon$ fraction of labels need to be changed to make it monotone.

- Testing monotonicity is equivalent to several other testing problems.

Monotonicity Testers via Sparse 2-TC-spanners

Lemma. G has a 2-TC-spanner with $s(n)$ edges

$$\Downarrow$$

monotonicity of functions on G can be tested in time $O(s(n)/(\epsilon\, n))$

Implications of structural bounds on the size of 2-TC-spanners:

- Lower bounds for a hypercube and (hyper)grid [BGJRW]:

TC-spanner method does not improve sortedness testers for matrices

- Upper bounds for other graph families [BGJRW09]:

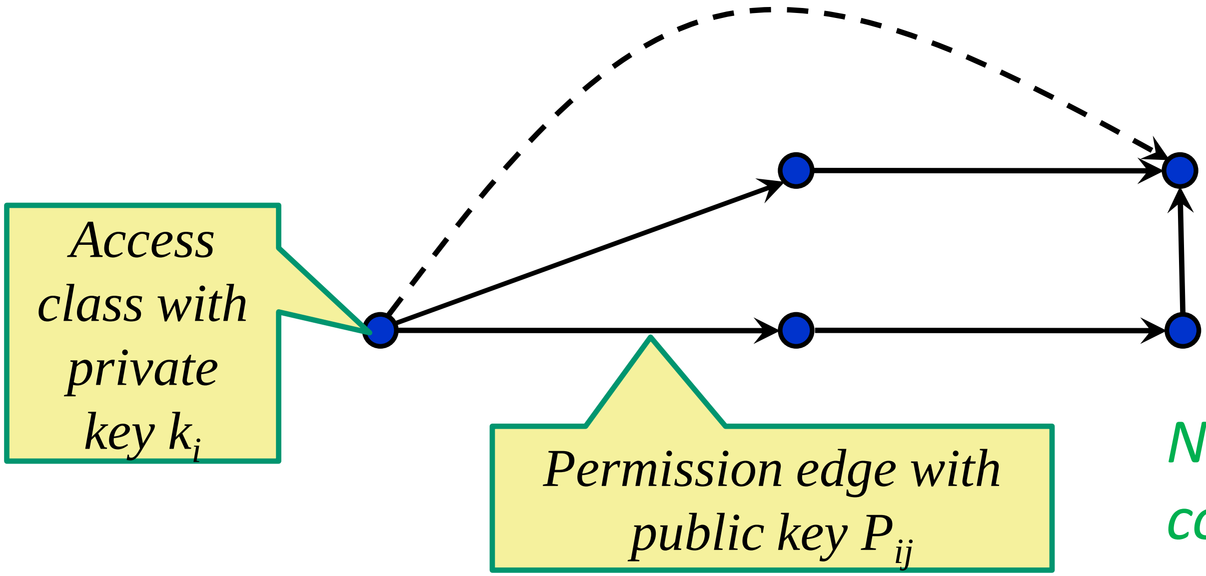
better monotonicity testers for those graph families

e.g., for planar graphs run time improved from $O((n^{1/2} \log n)/\epsilon)$ [FLNRRS02] to $O((\log^2 n)/\epsilon)$

Application 2: Access Control

Efficient key management in access hierarchies [Attalah Frikken Blanton 05, Attalah Blanton Frikken 06, Santis Ferrara Massuci 07, Attalah Blanton Fazio Frikken 09]

Used in content distribution, operating systems and project development



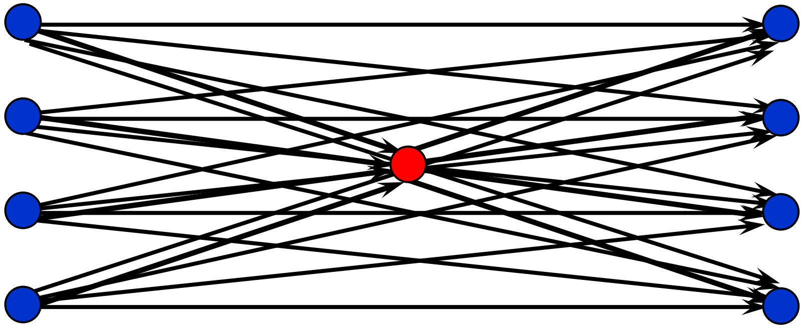
Need k_i to efficiently compute k_j from P_{ij}

To speed up key derivation time, add shortcut edges consistent with permission edges

[Attalah Blanton Fazio Frikken 09]: access hierarchies are often low-dim posets (can be embedded into low-dim grids via order-preserving embeddings).

Steiner TC-spanners

In the access control application, one can add new nodes to the TC-spanner



H is a Steiner k-TC-spanner of G if

- vertices(G) $\subseteq$ vertices(H)
- $distance_H(u,v) \leq k$ if G has a path from u to v

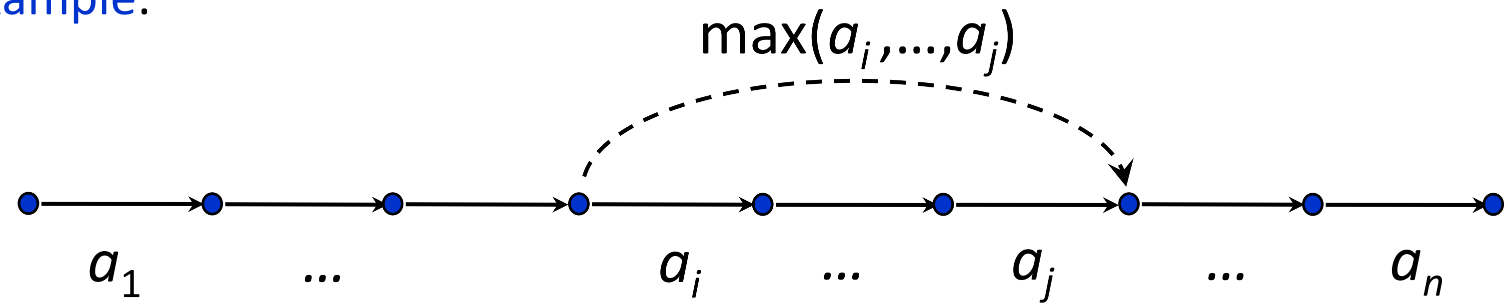
  ∞ otherwise

Observation: Steiner vertices do not decrease the number of edges in sparsest TC-spanners of grids.

Application 3: Data Structures

Computing partial products in a semigroup [Yao 82, Alon Schieber 87, Chazelle 87, Bodlaender Tel Santoro 94, Thorup 97]

Example:



Goal: quickly answer queries max(a_i,…,a_j) for all $i \leq j$.

- Question: How many values should we store if we want to compute max of at most k numbers per query?

- Answer: storage = size of sparsest k-TC-spanner for the directed line.

This example generalizes to other partial products and to directed trees.

Our Contributions

- Common abstraction for several applications [BGJRW09]
  - property testing
    - testing monotonicity of functions
    - testing if a function f:{1,…,n} $\rightarrow$ R is Lipschitz [Jha R]
  - access control
  - data structures
- Structural results on TC-spanners
  - path-separable graphs [BGJRW09]
  - directed hypercube/hypergrid [BGJRW]
  - low-dimensional posets [BGJRW]
- Computational results on directed spanners

  k-TC-SPANNER: Given a graph, compute a sparsest k-TC-spanner
  - new algorithms, inapproximability results [BGJRW09]

Structural Results on TC-spanners

Let $S_k(G)$= size of sparsest k-TC-spanner of G

- Path-separable graphs [BGJRW09]

  (e.g., H-minor-free graphs: planar, bounded genus, and bounded tree-width)
  - Path separators were defined in [Abraham Gavoille 06]

  For $O(1)$-path-separable graphs G with n nodes, $S_k(G)= O(n \log n \cdot \lambda(k,n))$
  - E.g., improves run time of monotonicity testers on planar graphs

  from $O((n^{1/2} \log n)/\epsilon)$ [FLNRRS02] to $O((\log^2 n)/\epsilon)$.

- Directed hypergrid $[m]^d$ [BGJRW]
  - For small m:

    $$\log_2(S_2([m]^d)) = c_m d \text{ where } c_2 \approx 1.16,\ c_3 \approx 2.03,\ c_4 \approx 2.82$$

    For reference: $[2]^d$ has $\Theta(2^d\, d)$ edges; $TC([2]^d)$ has $3^d = 2^{cd}$ edges where $c \approx 1.59$
  - For $m \geq 3$:

    $$S_2([m]^d) \leq m^d \log^d m, \text{ and this is tight up to } O(2d\,(\log\log m)^{d-1}) \text{ factor.}$$
  - Monotonicity testers for hypercube/hypergrid cannot be improved using the TC-spanner method.

Structural Results on Steiner TC-spanners

Let $S_k(G)$= size of sparsest Steiner k-TC-spanner of G

- Low-dimensional posets

(posets embeddable into low-dim grids via order-preserving embeddings).

Previous work [Attalah Blanton Fazio Frikken 09, DeSantis Ferrara Masucci 07]

Setting of k	Bounds on $S_k(G)$	Authors
$k = 2d-2+i$ for $i \geq 2$	$O(n (\log^{d-1} n) \lambda(k,n))$	[ABFF]
$k = 2d + \log^* n$	$O(n \log^{d-1} n)$	[ABFF]
$k = 3$ for fixed d	$O(n \log^{d-1} n \log \log n)$	[DFM]

[BGJRW]

Setting of k	Our Bounds on $S_k(G)$	
$k = 2$	$O(n \log^d n)$ for all d	$\Omega(n (\log n / \log \log n)^d)$ for fixed d
$k \geq 3$		$n \log^{\Omega(d)} n$ for fixed d

Our Contributions

- Common abstraction for several applications [BGJRW09]
  - property testing
    - testing monotonicity of functions
    - testing if a function $f:\{1,\ldots,n\} \to R$ is Lipschitz [Jha R]
  - access control
  - data structures
- Structural results on TC-spanners
  - path-separable graphs [BGJRW09]
  - directed hypercube/hypergrid [BGJRW]
  - low-dimensional posets [BGJRW]
- Computational results on directed spanners

  k-TC-SPANNER: Given a graph, compute a sparsest k-TC-spanner
  - new algorithms, inapproximability results [BGJRW09]

2-TC-Spanner of an $m\times m$ Grid

Lemma. *A sparsest 2-TC-spanner of an $m \times m$ grid has*

$\leq m^2 \log^2 m$ and $\Omega(m^2 \log^2 m / \log \log m)$ edges.

Proof:

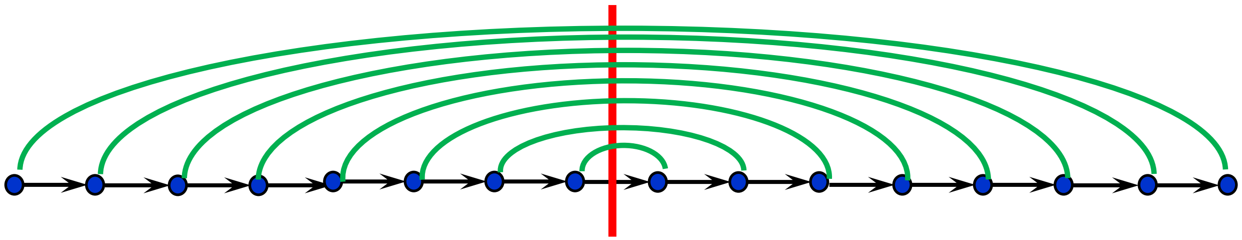
Upper bound: graph product of two 2-TC-spanners of the line.

Lower bound: a tradeoff argument, balancing number of edges of different types.

Lower Bound for an $m\times m$ Grid: Starting Point

Lemma. *A sparsest 2-TC-spanner of a line with m nodes has $\Omega(m \log m)$ edges.*

Proof:

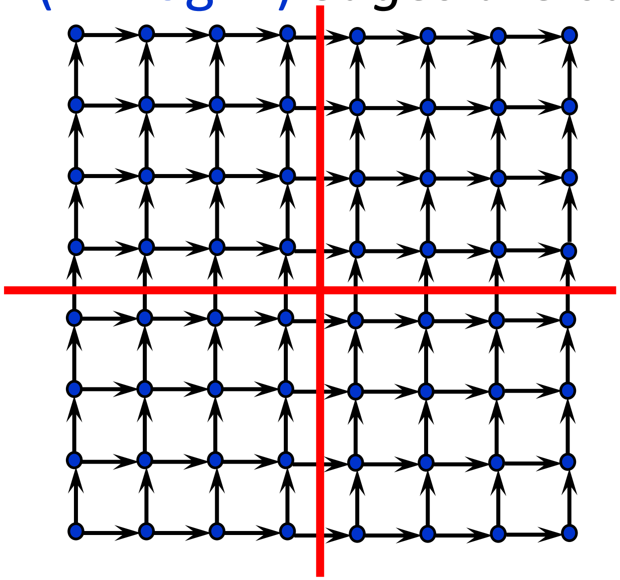


- Each pair of nodes connected by a green arc contributes an edge crossing the midline.
- $\geq m/2$ edges cross the midline.
- Continue recursively to obtain $\Omega(m \log m)$ bound.

■

Lower Bound for an m×m Grid: First Attempt

Approach: Recursively halve the grid in both dimensions hoping that each time $\Omega(m^2 \log m)$ *edges are cut.*

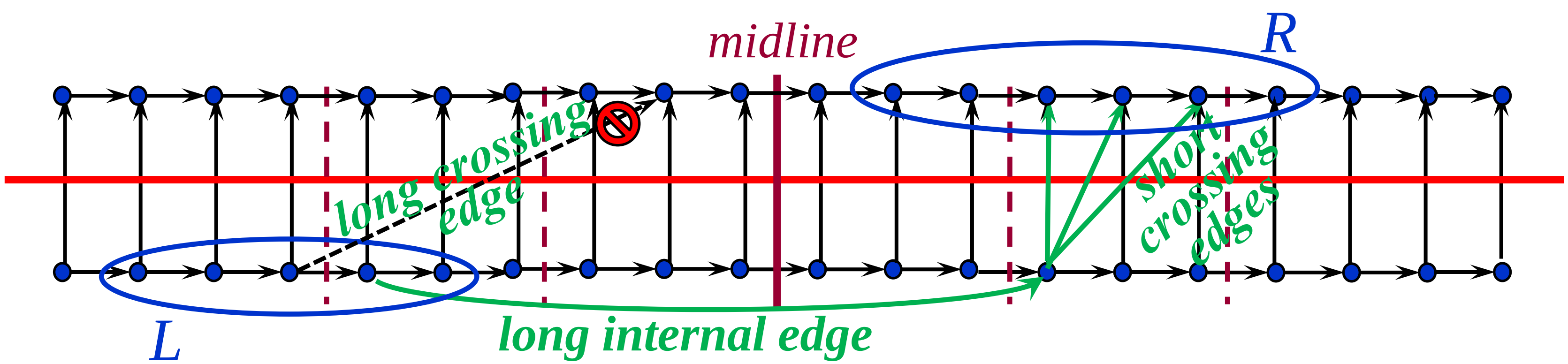


Problem: A 2-TC-spanner could contain the transitive closure for each quarter, and only $3m^2$ edges crossing the cut.

There is a tradeoff between the number of internal edges and the number of edges crossing the cut.

Two-Line Tradeoff Lemma

Lemma. *Consider a 2-TC-spanner of an* $m \times 2$ *grid. Cut it horizontally. If it has* $\le m \log^2 m / 32$ *internal edges, it has* $\ge m \log^2 m / (16 \log\log m)$ *crossing edges.*

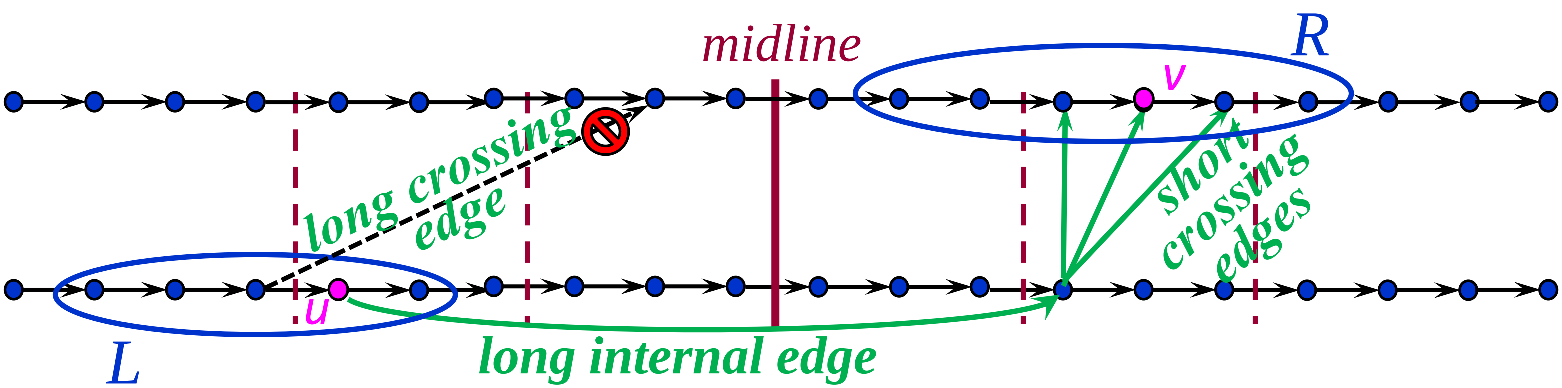


- $\log m / (2 \log\log m)$ stages, each contributing $m/8$ crossing edges.
- In the first stage, divide the graph into $\log^2 m$ blocks of the same length.
- Call an edge ***long*** if it starts and ends in different blocks (***short*** otherwise).
- L = set of low left nodes, not incident to long crossing edges.

  R = set of high right nodes, not incident to long crossing edges.

Two-Line Tradeoff Lemma: Stage 1 in the Proof

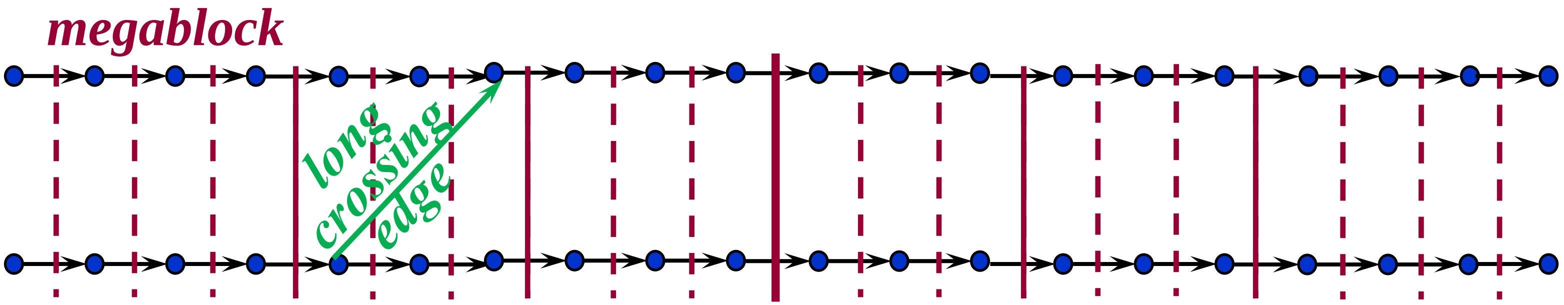
Lemma. Consider a 2-TC-spanner of an $m \times 2$ grid. Cut it horizontally. If it has $\leq m \log^2 m / 32$ internal edges, it has $\geq m \log m / (16 \log \log m)$ crossing edges.



- $\log m / (2 \log \log m)$ stages, each contributing $m/8$ crossing edges.
- In the first stage, divide the graph into $\log^2 m$ blocks of the same length.
- $u \in L$ can have a length-2 path to $v \in R$ only via a long internal edge.
- Each long low internal edge can be used by $\leq m/\log^2 m$ such pairs (u,v).
- If there are $< m/8$ long crossing edges, there are $> m^2/16$ pairs in $L \times R$.
- Then there are $> (m^2/16) / (m/\log^2 m) = m \log^2 m / 16$ (long) internal edges.
- Contradiction. That is, stage 1 indeed contributes $m/8$ long crossing edges.

Two-Line Tradeoff Lemma: Subsequent Stages

Lemma. Consider a 2-TC-spanner of an $m \times 2$ grid. Cut it horizontally. If it has $\le m \log^2 m / 32$ internal edges, it has $\ge m \log^2 m / (16 \log \log m)$ crossing edges.



- $\log m / (2 \log \log m)$ stages, each contributing $m/8$ crossing edges.
- In each subsequent stage, call blocks from the previous stage ***megablocks***.
- Divide each megablock into $\log^2 m$ blocks of the same length.
- Call an edge ***long*** if it starts and ends in different blocks, but stays within the same megablock.
- Show that there are $m/8$ long crossing edges (contribution of this stage).

■

2-TC-Spanner of the Grid:

Lemma. *A sparsest 2-TC-spanner of an* $m \times m$ *grid has*

$\leq m^2 \log^2 m$ *and* $\Omega(m^2 \log^2 m / \log\log m)$ *edges.*

Lemma. *A sparsest 2-TC-spanner of the directed grid* $[m]^d$ *has*

$\leq m^d \log^d m$ *and* $\Omega\left(m^d \log^d m / \left(2d\,(\log\log m)^{d-1}\right)\right)$ *edges.*

Our Contributions

- Common abstraction for several applications [BGJRW09]
  - property testing
    - testing monotonicity of functions
    - testing if a function f:{1,…,n} $\rightarrow$ R is Lipschitz [Jha R]
  - access control
  - data structures
- Structural results on TC-spanners
  - path-separable graphs [BGJRW09]
  - directed hypercube/hypergrid [BGJRW]
  - low-dimensional posets [BGJRW]
- Computational results on directed spanners

  k-TC-Spanner: Given a graph, compute a sparsest k-TC-spanner

  - new algorithms, inapproximability results [BGJRW09]

Computational Results on k-TC-Spanner

Algorithms

Setting of k	Approxi-mability	Authors/ Technique	Notes
$k=2$	$O(\log n)$	[Kortsarz Peleg]	
$k=3$	$O(n^{2/3} \text{ polylog } n)$	[Elkin Peleg]	
$k > 2$	$O((n \log n)^{1-1/k})$	**CP+sampling**	Applies to directed spanners ,…. Simplifies [EP] for $k=3$
$k = \Omega(\log n)$	$O((n \log n)/k^2)$	**path sampling**	Better than directed spanners

Hardness

Setting of k	Inapproxi-mability	Assumption	Notes
$k = 2$	$\Omega(\log n)$	$P \neq NP$	Matches upper bound
constant $k > 2$	$\Omega(2^{\log^{1-\epsilon} n})$	$NP \subsetneq DTIME(n^{\text{polylog } n})$	Improvement $\Rightarrow$ breakthrough
$k = n^{1-\gamma}\ \forall \gamma > 0$	$\Omega(1+\epsilon)$	$P \neq NP$	

Open Questions

- Can TC-spanner perspective be used to get new algorithms for problems related to monotonicity testing?
  - approximating distance to monotonicity or the length of LIS
  - online/distributed reconstruction of monotone functions
  - in streaming
- Other applications of TC-spanners and Steiner TC-spanners?