Chapter 5.

Open Problems

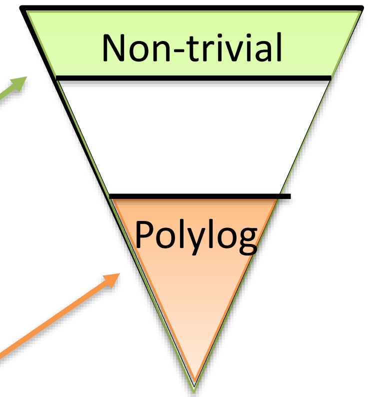
Danupon Nanongkai

KTH, Sweden

ADFOCS 2018

Last edited: Aug. 12, 2018

Challenge #1: Use amortization & randomization to minimize update time.



Non-trivial Single-Source Distances?

Light reading

Known: Incremental/decremental **O(n)-time** [Even-Shiloach'81] **(Next!)**

Easier(?): (1+ε)-approx [Sankowski FOCS'04+COCOON'05], [HKN FOCS'14], [BrandNS'17]

Also: Exact Global Mincut

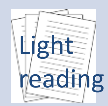
Polylog (2-ε)-approximate max bipartite matching?

Known: $n^{1/k}$-update time **(2-1/100^k)**-approx [BhattacharyNH STOC'16]. Also see [Gupta-Peng FOCS'13], [Bernstein-Stein ICALP'15, SODA'16]

Also: 3-edge connectivity, approx global min-cut, max-flow, sparsest cut, effective resistance, etc.

n = # of nodes, m=# of edges

Challenge #2: Close oblivious-adaptive-deterministic gaps

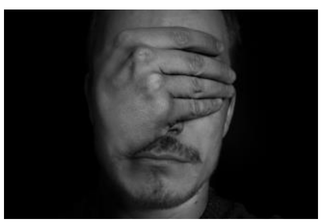
Problems	Oblivious adv. 	Adaptive adv.	Deterministic
Spanning Forest (worst case)	$\text{polylog}\, n$ [Kapron King Mountjoy SODA'13]	$n^{o(1)}$ [NSW FOCS'17]	$\sqrt{n}$ [EGIN FOCS'92]
Dec. Single-Source Shortest Path (decremental approximate amortized)	$n^{o(1)}$ [HKN FOCS'14]	$\min(\frac{n^2}{m}, n^{\frac{3}{4}})$ [Bernstein, Chechik STOC'16, SODA'17, ICALP'17]	$\min(\frac{n^2}{m}, n^{\frac{3}{4}})$ [Bernstein, Chechik STOC'16, SODA'17, ICALP'17]
(Δ+1)-coloring	polylog(n) [BCHN SODA'18] 	n [Trivial]	n [Trivial]
Dec. **Directed** Single-Source Shortest Paths (decremental amortized)	$n^{0.9}$ [HKN STOC'14] 	n [Even Shiloach JACM'81]	n [Even Shiloach JACM'81]
Maximal Matching	$O(1)$ [Solomon FOCS'16]	$\sqrt{m}$ [Neiman Solomon STOC'13]	$\sqrt{m}$ [Neiman Solomon STOC'13]
Cut Sparsifier (worst-case)	$\text{polylog}\, n$ [ADKKP FOCS'16]	m [trivial]	m [trivial]
Spanner (amortized)	$\text{polylog}\, n$ [BKS ESA06, SODA'08]	m [trivial]	m [trivial]

n = # of nodes, m=# of edges

Randomized Dynamic Algorithms

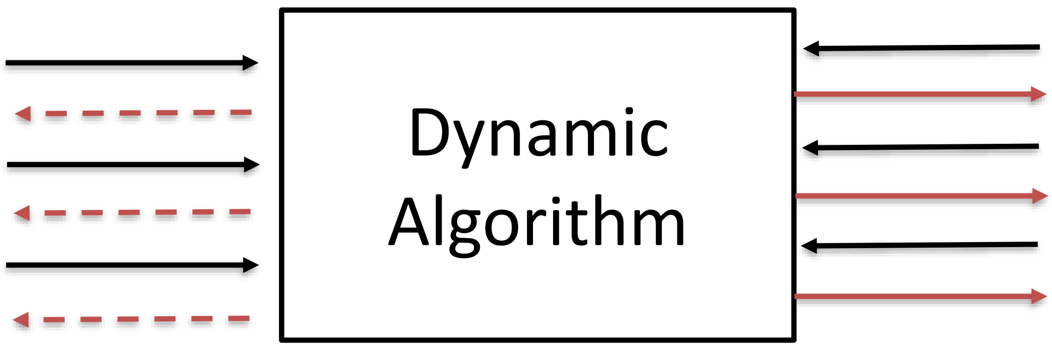
- Las Vegas: **Expected** update time
- Monte Carlo: **Wrong** output with small probability

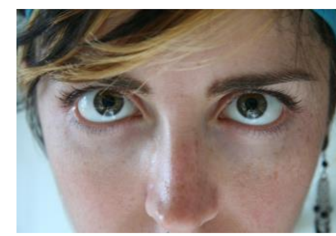
Assumption: *Oblivious* adversary.



Oblivious Adversary

e.g. social network data from hard disk

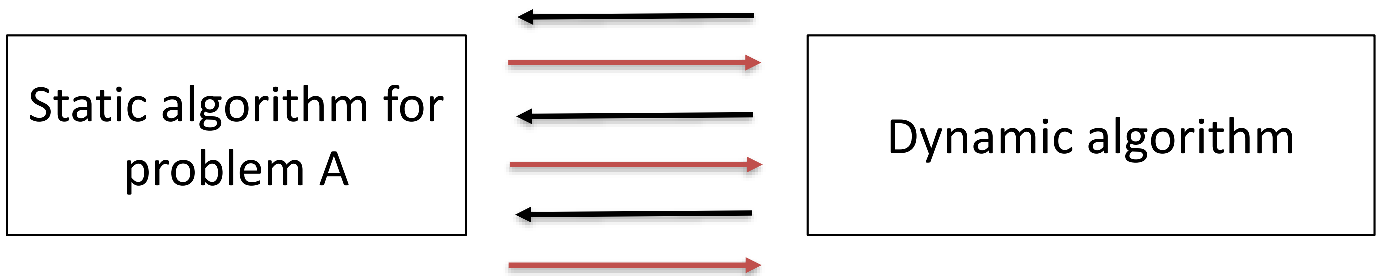




Adaptive Adversary

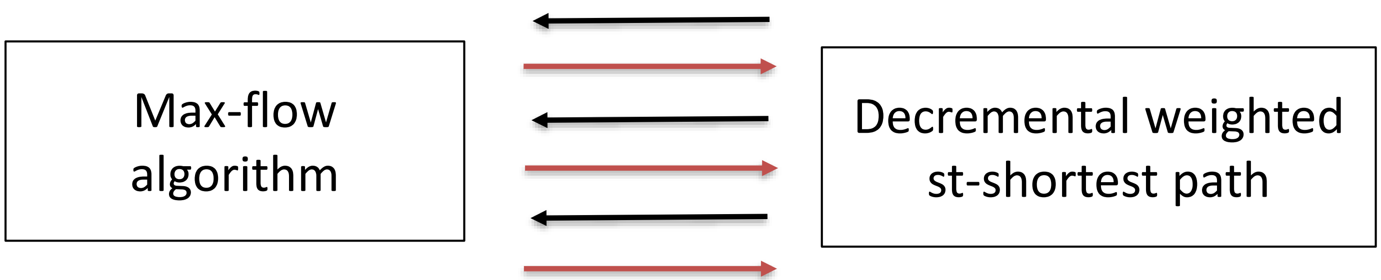
De-randomization Applications

Dynamic algorithm as data structure:



Adaptive Adversary

Example [Garg-Konemann FOCS'98]:

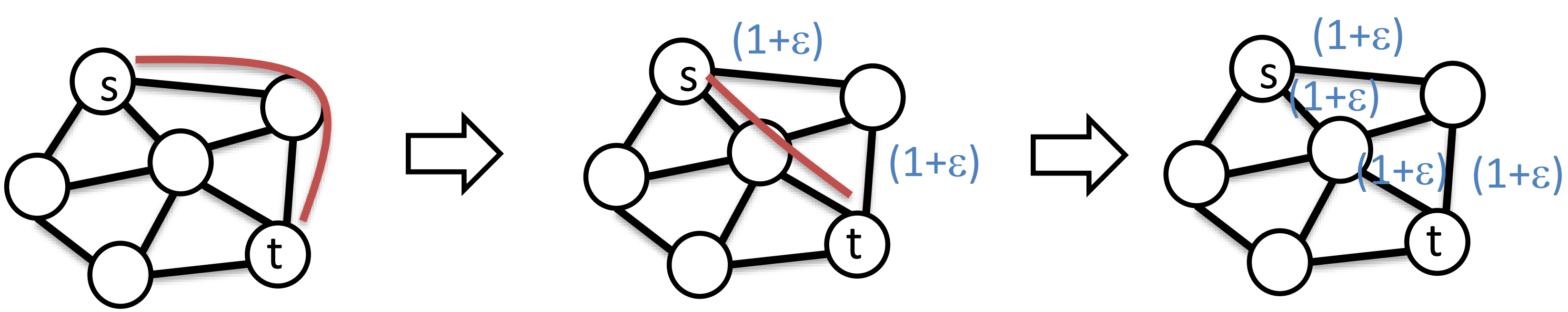


Dyn. Shortest Paths → Max Flow

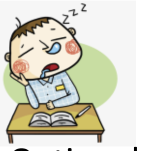
Known: rand. $n^{o(1)}$ update time for weighted $(1+\varepsilon)$-approx decremental st-shortest path [HenzingerKN. FOCS'14]

Garg-Konemann [FOCS'98], Madry [STOC'10]:
de-randomized → $n^{1+o(1)}$-time $(1+\varepsilon)$-approx max flow

Randomized algorithm against adaptive adversary is also enough.



Other examples: Interior point method, Tree packing, Interval packing, Traveling Salesperson.



Optional

Power of Randomization

Oblivious adversary takes **a long time** to destroy random solution

Example 1: 2Δ-coloring (Δ=max degree)

Goal: Maintain 2Δ-vertex-coloring

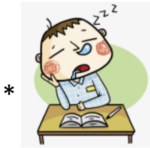
Algorithm: **Recolor** node with a **random color** from ≥ **Δ** available colors.

Cost: **O(Δ)** to recolor a node, i.e. to find available colors.

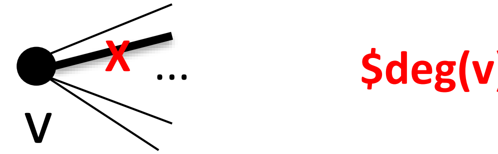
Adaptive adversary can force us to recolor and pay **O(Δ)**

Oblivious adversary takes more time to force a node to recolor

Example 2: maximal matching [Baswana, Gupta, Sen FOCS'11]*

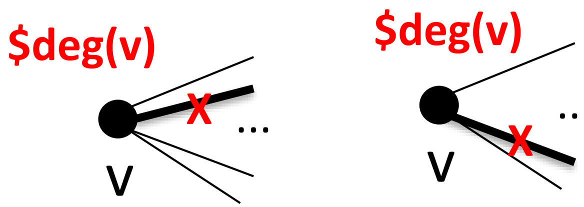


- **Degree(v)** time to rematch node v



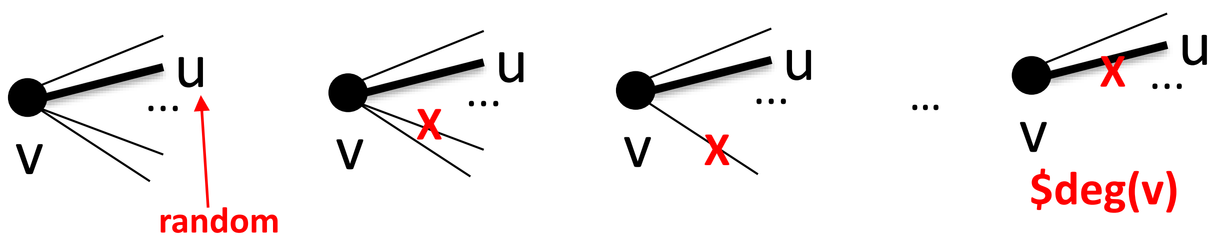
- **Adaptive adversary** can force us to always pay degree(v)

Adaptive Adversary



- **Solution:** Match **randomly**. Non-oblivious adversary will take some time to delete matched edge.

Oblivious Adversary



*Lots of details are hidden

Challenge #3 Worst-case update time

Weighted APSP (all-pairs shortest paths):
Maintain distances between every pair of nodes

Amortization may give more power!

Known amortized: $O(n^2)$ [Demetrescul FOCS'00]

Known worst-case: $O(n^{2+2/3})$ [AbhrahamCK SODA'17]

Conjecture: $\Theta\left(n^{2.5}\right)$

Some others:

Problems	Amortized	Worst-Case
2-edge connectivity	polylog(n) [HLT STOC'98]	$O(m^{1/2})$ [Frederickson FOCS'91]
Incremental SSSP	O(n) [EvenS JACM'81]	O(m)

Challenge #4: New Conjectures or Techniques to Separate

worst-case from **amortized** bounds

2-edge connectivity: **polylog(n)** amortized [HLT STOC'98] but **$O(n^{1/2})$** worst case [Frederickson FOCS'91]

deterministic from **randomized** algorithms

Dec. Single-Source Shortest Paths: **$n^{o(1)}$** randomized [HKN FOCS'14] but $min(\frac{n^2}{m}, n^{\frac{3}{4}})$ deterministic [Bernstein, Chechik STOC'16, SODA'17, ICALP'17]

incremental from **decremental** algorithms

Single-source Reachability: (amortized) **polylog(n)** incremental but **$O(n^{1/2})$** decremental [ChechikHILP STOC'16]

Cash Opportunities*

1. **5,000** SEK (ca. 500 Euros):
   Prove or disprove the **OMv** conjecture
2. **3,000 SEK**
   Prove or disprove the ***v-hinted Mv*** conjecture

Related to tight $\Theta(\mathbf{n}^{1.407})$ bound for st-reach, etc

v-hinted OMv (informal)
Input: Phase 1: Boolean matrix **M**, Phase 2: Boolean matrix **V**, Phase 3: index i
Output the matrix-vector product $\boldsymbol{MV_i}$, where V_i is the i-th column of **V** .
Naïve algorithm: Compute MV in phase 2 or MV_i in phase 3.
Conjecture: Nothing better than the naive algorithm.

*Expires Aug. 17, 2028