Partially dynamic approximate shortest paths in unweighted, undirected graphs

Monika Henzinger
University of Vienna, Austria

Sebastian Krinninger
University of Vienna, Austria

Danupon Nanongkai
Nanyang Technological University, Singapore

Definitions

- **APSP**: All-pairs shortest paths
- **SSSP**: Single-source shortest paths (with source node s)
- **Incremental** algorithm: allows edges insertions
- **Decremental** algorithm: allows edge deletions
- The **distance** between u and v in G is denoted by $d_G(u, v)$.
- The **level** $\ell(v)$ of a node v in a tree is the distance of v to the root.
- A distance estimate $d'(u, v)$ is an (α, β)-**approximation** of $d(u, v)$ if $d(u, v) \leq d'(u, v) \leq \alpha d(u, v) + \beta$

SSSP: Deterministic incremental algorithm

Result:
- Approximation: $(1 + \epsilon, 0)$
- Total update time: $\widetilde{\mathcal{O}}(m^{3/2} n^{1/4})$
- Query time: $\mathcal{O}(1)$

Idea: Try to be **lazy**, i.e., keep BFS tree for as long as possible

Algorithm: after insertion of edge (u, v)
If $\ell(v) < \ell(u) - \delta$: recompute BFS tree rooted at s
After κ insertions since last recomputation: recompute BFS tree

→ Algorithm guarantees **additive** error of $\kappa\delta$

Lemma: If G' results from G by inserting edges and there is v s.t. $d_{G'}(v, s) \leq d_G(v, s) - \delta$, then $\sum_x d_{G'}(x, s) \leq \sum_x d_G(x, s) - \Omega(\delta^2)$.

→ Number of BFS tree computations: $\mathcal{O}(m/\kappa + n^2/\delta^2)$

Additive approximation can be converted into $(1 + \epsilon, 0)$-approximation
Algorithm also works in distributed setting (CONGEST model)

APSP: Decremental algorithm

Result:
- Approximation: $(1 + \epsilon, 2)$
- Total update time: $\widetilde{\mathcal{O}}(n^{5/2})$
- Query time: $\mathcal{O}(1)$

Idea: Run algorithm of Roditty and Zwick [4] on sparse emulator
Problem: Edges might be **inserted** into emulator

We use an emulator H with the following properties:
- H is a weighted graph with $\widetilde{\mathcal{O}}(n^{3/2})$ edges that provides $(1 + \epsilon, 2)$-approximation of distances in G
- H can be maintained under deletions in G in total time $\widetilde{\mathcal{O}}(mn^{1/2})$
- H is **locally persevering**: Every path of G of length at most $2/\epsilon$ is either contained in H or can be $(1 + \epsilon, 2)$-approximated by a path P' in H whose edges are in H since the beginning.

Monotone Even-Shiloach tree
- Even-Shiloach tree maintains shortest paths from a root node r up to given depth (with corresponding levels $\ell(v)$)
  → Central tool in algorithm of Roditty and Zwick [4]
- Our extension to insertions of edges: if $\ell(v) + w_H(u, v) < \ell(u)$, then make v the parent of u and do *not* update the level of u.
  → Correctness follows from locally persevering property of H
  → Intuition: decreasing $\ell(u)$ to $\ell(v) + w(u, v)$ is not necessary as $\ell(u)$ already provides $(1 + \epsilon, 2)$-approximation of $d_G(u, r)$.
  → Inductive argument gives $(1+\epsilon, 2)$-approximation for all nodes

APSP: Deterministic decremental algorithm

Result:
- Approximation: $(1 + \epsilon, 0)$
- Total update time: $\mathcal{O}(mn \log n)$
- Query time: $\mathcal{O}(\log \log n)$

Idea: Derandomize algorithm of Roditty and Zwick [4]
Motivation: Probabilistic algorithms require **oblivious** adversary

Probabilistic center cover [4]
- Each node becomes center with probability $\frac{c \ln n}{\delta}$ (for constant c)
- With high probability, every node is in distance δ to a center (if connected component has size at least δ)
- Maintain Even-Shiloach trees of depth δ for all $\widetilde{\mathcal{O}}(n/\delta)$ centers
  → Total time: $\widetilde{\mathcal{O}}(m\delta n/\delta) = \widetilde{\mathcal{O}}(mn)$

Deterministic center cover
- Greedily open centers at uncovered nodes
  → Each center c covers all nodes in distance $2r_c = \delta$
  → Number of centers after initialization: $\mathcal{O}(n/\delta)$
- If connected component of center becomes too small:
  → **Move** center c out of component C and reduce r_c by $|C|$
  → Nodes in small component C become assigned to center
- At termination: all nodes in distance r_c assigned to center c
  → Each node is assigned to at most one center
  → Each center has $\Omega(\delta)$ assigned nodes
  → Total number of centers: $\mathcal{O}(n/\delta)$
- **Moving Even-Shiloach trees** on centers
  → Relocating tree to a neighboring node costs additional $\mathcal{O}(m)$
  → Total "moving distance" of all centers is at most n
- Total time $\mathcal{O}(m\delta n/\delta + mn) = \mathcal{O}(mn)$ for center cover

SSSP: Decremental algorithm

Result:
- Approximation: $(1 + \epsilon, 0)$
- Total update time: $\widetilde{\mathcal{O}}(n^{9/5+o(1)} + m^{o(1)})$
- Query time: $\mathcal{O}(1)$

Step 1: $\widetilde{\mathcal{O}}(mn^{4/5} + m^{3/2} n^{1/4})$ time algorithm
- Based on lazy recomputation of BFS tree and faster center cover

Step 2: Run modified algorithm on sparse emulator
- Thorup and Zwick [5]: $(1 + \epsilon, 2(1 + \frac{2}{\epsilon})^{k-2})$-emulator
- Monotone Even-Shiloach tree preserves this approximation
- Thorup-Zwick emulator provides further needed properties (as observed by Bernstein and Roditty [6])

References

[1] M. Henzinger, S. Krinninger, and D. Nanongkai. Dynamic Approximate All-Pairs Shortest Paths: Breaking the $\mathcal{O}(mn)$ Barrier and Derandomization. FOCS 2013.
[2] M. Henzinger, S. Krinninger, and D. Nanongkai. Sublinear-Time Maintenance of Breadth-First Spanning Tree in Partially Dynamic Networks. ICALP 2013.
[3] M. Henzinger, S. Krinninger, and D. Nanongkai. A Subquadratic-Time Algorithm for Dynamic Single-Source Shortest Paths. manuscript.
[4] L. Roditty and U. Zwick. Dynamic Approximate All-Pairs Shortest Paths in Undirected Graphs. FOCS 2004.
[5] M. Thorup and U. Zwick. Spanners and emulators with sublinear distance errors. SODA 2006.
[6] A. Bernstein and L. Roditty. Improved Dynamic Algorithms for Maintaining Approximate Shortest Paths Under Deletions. SODA 2011.

M. Henzinger and S. Krinninger are supported by the Austrian Science Fund (FWF): P23499-N23, the Vienna Science and Technology Fund (WWTF) grant ICT10-002, the University of Vienna (IK I049-N), and a Google Faculty Research Award. The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 317532. D. Nanongkai is supported in part by the following research grants: Nanyang Technological University grant M58110000, Singapore Ministry of Education (MOE) Academic Research Fund (AcRF) Tier 2 grant MOE2010-T2-2-082, and Singapore MOE AcRF Tier 1 grant MOE2012-T1-001-094. Part of this work was done while D. Nanongkai was visiting the University of Vienna.