Fully Dynamic Algorithms for Transitive Reduction

Gramoz Goranci[1] Adam Karczmarz[2] Ali Momeni[3] Nikos Parotsidis[4]

[1]University of Vienna [2]University of Warsaw [3]UniVie Doctoral School Computer Science DoCS, University of Vienna [4]Google Research

Definition

Definition. *A minimal subgraph H of G that preserves reachability:*

there is an $x \to y$ path in $G \iff$ there is an $x \to y$ path in H.

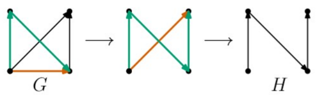


Figure 1. An example of computing H.

Combinatorial Algorithm

Main idea: maintain *redundant* edges in G.

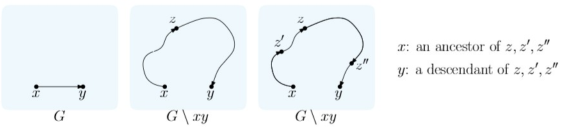


Figure 2. Edge xy is redundant in G.

1. For Directed Acyclic Graphs (DAGs)

Lemma 1. *Edge xy is redundant in G iff one of the cases in Figure 3 below happens.*

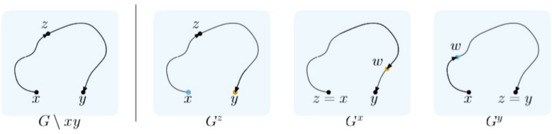


Figure 3. The info we need to declare xy redundant.

The data structure:

- For every vertex $v \in V$, maintains a graph G^v.
- After an insertion around v in G, G^v is reset as the snapshot of G.
- After edge deletions in G, the deletions are passed to G^v.

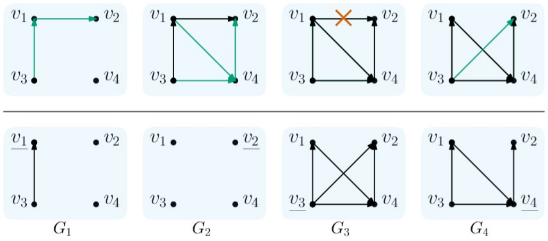


Figure 4. How data structure maintains the info.

Two variables:

1. Counter $c(xy)$: number of $z \notin \{x, y\}$ with $xy \in E^z$, $x \in \mathrm{Anc}^z$, and $y \in \mathrm{Desc}^z$ in G^z.
2. Binary value $t(xy)$: is set to 1 if
   - vertex x has an out-neighbor $z \in \mathrm{Anc}^y \setminus y \in G^y$, or
   - vertex y has an in-neighbor $z \in \mathrm{Desc}^x \setminus x \in G^x$.

Invariant. *Edge xy belongs to H iff $c(xy) = 0$ and $t(xy) = 0$.*

Theorem 2. *Our deterministic data structure maintains a transitive reduction H of a DAG G in $\mathcal{O}(m)$ amortized update time for extended updates.*

2. For General Graphs

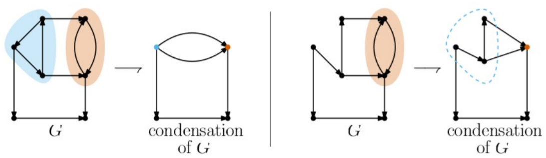


Figure 5. The challenge when G is a general graph.

Theorem 3. *Our deterministic data structure maintains a transitive reduction H of a general digraph G in $\mathcal{O}(m + n \log n)$ amortized update time for extended updates.*

Algebraic Algorithm

Main idea: maintain the inverse of a matrix A associated with G.

1. For Directed Acyclic Graphs (DAGs)

A: the adjacency matrix of G, where $A_{u,v} = 1$ iff $uv \in E$.

Fact. *$A^k_{u,v}$: number of walks $u \to v$ of length k.*

$$(I - A)\underbrace{(I + A + \cdots + A^{n-1})}_{\text{whole number of walks}} = I - A^n = I$$

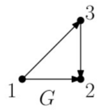
Invariant. *Edge xy belongs to H iff $(I - A)^{-1}_{u,v} = 1$.*

2. For General Graphs

$\tilde{A}$: a symbolic adjacency matrix from $(\mathbb{F}(X))^{n \times n}$, where

$$\tilde{A}_{u,v} = \begin{cases} x_{u,v} & \text{if } uv \in E, \\ 0 & \text{otherwise.} \end{cases}$$

We add self-loops to G, ensuring $\det(\tilde{A}) \neq 0$.



$$\tilde{A} = \begin{pmatrix} x_{1,1} & x_{1,2} & x_{1,3} \\ 0 & x_{2,2} & 0 \\ 0 & x_{3,2} & x_{3,3} \end{pmatrix}$$

Figure 6. An example of $\tilde{A}$.

Theorem 4. *Let $F = \{u_i v_i : i = 1, \ldots, k\}$ be a group of parallel inter-SCC edges between SCCs R, T of G. Let r, t be arbitrary vertices of R, T, respectively. The group F is redundant iff:*

$$\tilde{A}^{-1}_{r,t} \not\equiv -\sum_{i=1}^{k} (-1)^{u_i + v_i} \cdot x_{u_i,v_i} \cdot \tilde{A}^{-1}_{r,u_i} \cdot \tilde{A}^{-1}_{v_i,t}.$$

A: entries' evaluated for some uniformly random substitution $\bar{X} : X \to \mathbb{Z}/p\mathbb{Z}$, where p is a prime number $p = \Theta(n^{3+c})$, $c > 0$.

Lemma 5. *With high probability, A is invertible and group F of parallel inter-SCC edges is redundant iff:*

$$A^{-1}_{r,t} + \sum_{i=1}^{k} (-1)^{u_i + v_i} \cdot \bar{x}_{u_i,v_i} \cdot A^{-1}_{r,u_i} \cdot A^{-1}_{v_i,t} \neq 0.$$

Theorem 6. *Our randomized (Monte Carlo) data structure maintains a transitive reduction H of a general digraph G in:*

1. *$\mathcal{O}(n^2)$ worst-case update time for extended updates, and*
2. *$\mathcal{O}(n^{1.528} + m)$ worst-case update time for single-edge updates.*

The initialization time of the data structure is $\mathcal{O}(n^\omega)$.

References

[AGU72] A. V. Aho, M. R. Garey, and J. D. Ullman. *The Transitive Reduction of a Directed Graph.* *SIAM Journal on Computing* 1.2, pages 131–137, 1972.

[DI05] Camil Demetrescu and Giuseppe F. Italiano. *Trade-offs for fully dynamic transitive closure on DAGs: breaking through the $O(n^2)$ barrier.* *J. ACM* 52.2, pages 147–156, 2005.

[KS02] Valerie King and Garry Sagert. *A Fully Dynamic Algorithm for Maintaining the Transitive Closure.* *J. Comput. Syst. Sci.* 65.1, pages 150–167, 2002.

[RZ08] Liam Roditty and Uri Zwick. *Improved Dynamic Reachability Algorithms for Directed Graphs.* *SIAM J. Comput.* 37.5, pages 1455–1471, 2008.

[RZ16] Liam Roditty and Uri Zwick. *A Fully Dynamic Reachability Algorithm for Directed Graphs with an Almost Linear Update Time.* *SIAM Journal on Computing* 45.3, pages 712–733, 2016.

[San04] Piotr Sankowski. *Dynamic Transitive Closure via Dynamic Matrix Inverse (Extended Abstract).* In: *45th FOCS*, pages 509–517, 2004.

[San05] Piotr Sankowski. *Subquadratic Algorithm for Dynamic Shortest Distances.* In: *11th International Computing and Combinatorics Conference (COCOON)*. Vol. 3595, pages 461–470, 2005.