University of Zurich[UZH]

Department of Informatics

University of Zürich
Department of Informatics
Binzmühlestr. 14
CH-8050 Zürich
Phone. +41 44 635 43 11
Fax +41 44 635 68 09
www.ifi.uzh.ch/dbtg

UZH, Dept. of Informatics, Binzmühlestr. 14, CH-8050 Zürich

Prof. Dr. Michael Böhlen
Professor
Phone +41 44 635 43 33
Fax +41 44 635 68 09
boehlen@ifi.uzh.ch

Zürich, 30. September 2020

MSc Thesis
Topic: Implementing dynamic connectivity queries with spanning trees

Dynamic connectivity is one of the most fundamental problems in dynamic graph algorithms. In dynamic graphs, edges are inserted and deleted: the insertion of an edge can combine previously unconnected components, while the deletion of an edge can split two connected components. Keeping track of connected components in dynamic graphs is expensive. Among others, Thorup developed a method in which components are represented by forests of spanning trees [4], which is based on a dynamic tree data structure in [3, 1]. As some important details are not discussed in [4], the work [2] provides more details to implement the dynamic forest for answering dynamic connectivity query.

The goal of this Master thesis is to implement dynamic connectivity queries with spanning trees, applying some (or all) of the optimizations introduced in [4, 2] and to evaluate the implementation against a baseline.

Tasks

1. Study the relevant literature: (a) graph theory (b) data structures and algorithms (c) research papers on dynamic connectivity queries.
2. Implement a brute-force approach: (a) each connected component is represented with a spanning tree (b) two distinct connected components are merged when an inserted edge connect them (c) When an edge is deleted from one connected component, a brute-force search (e.g. via Breadth First Search (BFS) or Depth First Search (DFS) or some other search) is conducted to check whether the connected component is split into two or not. Reconstruct the spanning tree if the connected component is not split.
3. Introduce some optimizations to the brute-force approach described in the previous task. This may include some optimizations from [4, 2] or some other papers on dynamic connectivity via spanning trees.
4. Compare the performance of the different implementations experimentally.
5. Write the thesis.
6. Present the thesis in a DBTG meeting (25 minutes presentation).

References

[1] S. Alstrup, J. Holm, K. D. Lichtenberg, and M. Thorup. Maintaining information in fully dynamic trees with top trees. *ACM Trans. Algorithms*, 1(2):243–264, Oct. 2005.

[2] S.-E. Huang, D. Huang, T. Kopelowitz, and S. Pettie. Fully dynamic connectivity in $O(\log n(\log \log n)^2)$ amortized expected time. SODA '17, page 510–520, USA, 2017. Society for Industrial and Applied Mathematics.

[3] D. D. Sleator and R. E. Tarjan. A data structure for dynamic trees. STOC '81, page 114–122, New York, NY, USA, 1981. Association for Computing Machinery.

[4] M. Thorup. Near-optimal fully-dynamic graph connectivity. In *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*, STOC '00, page 343–350, New York, NY, USA, 2000. Association for Computing Machinery.

Supervisor: Sven Helmer (helmer@ifi.uzh.ch), Qing Chen (qing@ifi.uzh.ch)

Start date: 1 September 2020

End date: 28 February 2021

University of Zurich
Department of Informatics

Prof. Dr. Michael Böhlen
Professor