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, August 24, 2021

MSc Project: Implementing and evaluating different algorithms for connectivity queries on fully dynamic graphs

The connectivity problem, which checks if there is a connection between two nodes in a graph, is a fundamental problem in graph algorithms. Connectivity queries are important in application areas such as communication and transport networks to check their reliability and they are also relevant for social networks to investigate whether there are connections between users and the groups they belong to.

Computing connectivity on the fly using search strategies like breadth-first search (BFS) and depth-first search (DFS) is prohibitively expensive for large graphs. For static graphs, the connected components can be precomputed and information about them can be stored in auxiliary data structures, speeding up the query processing significantly. Updating data structures in the fully dynamic case is much more challenging.

The state-of-the-art algorithms for fully dynamic connectivity are quite complex and many of them have never been implemented and experimentally evaluated. This MSc project focuses on the implementation and evaluation of two such algorithms: the algorithm by Wulff-Nilsen [3] and an algorithm developed by Thorup [2].

The work is structured into the following tasks:

- **T1: Reading and understanding the literature**
  This encompasses reading the papers by Wulff-Nilsen and Thorup mentioned above (and any other relevant literature, such as technical reports and supplementary material) in order to gain a deeper understanding of their inner workings.
- **T2: Implementing the algorithms**
  The second task is about actually implementing the algorithms after studying them in the previous task. As these are complex algorithms, this may include the implementation of

(simpler) prototypes to test the feasibility of implementing (partial) data structures and procedures of the algorithms. During the implementation, the algorithms also have to be tested for correctness.

- **T3: Evaluating the algorithms**
  After both algorithms have been implemented and tested for correctness, the next step is to evaluate the performance of the algorithms. This includes such parameters as run-time for queries and updates and also storage requirements for the data structures. The experiments will be conducted on several real-world data sets.
- **T4: Summarize the findings in a report**
  Towards the end of the project, all the descriptions and findings need to be written down into a report.

(Optional task: depending on how quickly the work progresses, the work can be extended by looking at the approach taken by Nanongkai et al. [1].)

References

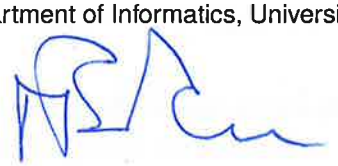
[1] D. Nanongkai, T. Saranurak, and C. Wulff-Nilsen. Dynamic minimum spanning forest with subpolynomial worst-case update time. In C. Umans, editor, *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA, October 15-17, 2017*, pages 950–961. IEEE Computer Society, 2017.

[2] 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.

[3] C. Wulff-Nilsen. Faster deterministic fully-dynamic graph connectivity. In *Proceedings of the twenty-fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1757–1769. SIAM, 2013.

Supervisor: Qing Chen, Sven Helmer

Department of Informatics, University of Zurich



Prof. Dr. Michael Böhlen