On Spanning Trees and Small Cuts in Congested Clique and MPC

Krzysztof Nowicki

Ph.D. Thesis
Supervisor:
prof. Tomasz Jurdziński, University of Wrocław

Institute of Computer Science
University of Wrocław
2020

Abstract

In this dissertation we propose several algorithms for the problems related to broadly understood graph connectivity: the Connected Components problem, the Minimum Spanning Tree problem, the Edge Connectivity problem and the Minimum Cut problem, in the Massively Parallel Computation model, and closely related Congested Clique model. Even though we mostly focus on those two models, some of the developed techniques are quite general and find applications also in other models of computation.

The **M**assively **P**arallel **C**omputation model [*Karloff, Suri, Vassilvitskii, SODA'10*] (or shortly, the MPC model) was proposed as a theoretical abstraction for the MapReduce framework [*Dean, Ghemawat, OSDI'04*]. However, it also characterizes many other parallel and distributed models and frameworks, e.g., Spark framework [*Zaharia, Chowdhury, Franklin, Shenker, Stoica, HotCloud'10*], Congested Clique model [*Lotker, Patt-Shamir, Pavlov, Peleg, SPAA'03, SICOMP'05*], Bulk Synchronous Parallel model [*Valiant, Commun. ACM, 1990*], and Coarse-Grained Multicomputers model [*Dehne, Fabri, Rau-Chaplin, Int. J. Comput. Geom. Appl., 1996*]. In MPC, the computation is performed by a set of processors, in synchronous rounds. Each round consists of the phase of local computation and the phase of communication. The main complexity measure of an MPC algorithm is the number of rounds it needs to solve a considered algorithmic problem.

In Chapter 2 we present a deterministic $O(1)$ round algorithm for the Spanning Forest problem and the Minimum Spanning Tree problem, for Congested Clique and MPC models. This result concludes the studies of the MST problem in Congested Clique model, initiated in 2003 by Lotker et al. [*SPAA'03,SICOMP'05*]. Lotker et al. proposed a deterministic algorithm with round complexity $O(\log\log n)$. There is a sequence of papers, presenting randomized MST algorithms with improved round complexity: an $O(\log\log\log n)$ round algorithm [*Hegeman, Pandurangan, Pemmaraju, Sardeshmukh, Scquizzato, PODC 2015*], an $O(\log^* n)$ round algorithm [*Ghaffari, Parter, PODC 2016*] and finally our $O(1)$ round algorithm [*Jurdziński, Nowicki, SODA'18*]. The algorithm described in Chapter 2 further improves on our $O(1)$ round algorithm, as it is deterministic and much simpler. Furthermore, a technique used in both $O(1)$ round algorithms can be applied in Broadcast Congested Clique model, where it allowed us to design the first algorithm with sublogarithmic round complexity for the Spanning Forest problem, which we discuss in Chapter 5.

In Chapter 3 we present a new kind of contraction process that allowed us to improve the Edge Connectivity algorithms in several models of computation. In Chapter 3 we only discuss the application of our contraction process to MPC and Congested Clique, where it gives a randomized $O(1)$ round algorithm that solves the Edge Connectivity problem with high probability. One of the main advantages of this new contraction process is its simplicity,

which allows to apply it in a wide range of computational models. In particular, we discuss the applications to the distributed CONGEST model and the sequential model in Chapter 5.

In Chapter 4 we present our Minimum Cut algorithms for the MPC model: an $O(1)$-round exact algorithm with $\tilde{O}(n)$ memory per machine and $\tilde{O}(m)$ global memory and an $O(\log n \cdot \log\log n)$ round $(2+\varepsilon)$-approximation algorithm with $O(n^\alpha)$ memory per machine and $\tilde{O}(m)$ global memory. Both algorithms rely on algorithmic techniques introduced by Karger: a random contraction technique [*Karger, SODA'93, SODA'94*] and an approach to the Minimum Cut problem that gave the first sequential algorithm with $O(m \operatorname{poly}(\log n))$ time complexity [*Karger, STOC'96, JACM 2000*]. Our main contribution consists in adjustments that allow to use those techniques to obtain MPC algorithms with small round complexity and small global memory.