Maximum flow is approximable by deterministic constant-time algorithm in sparse networks
(Extended abstract)

Endre Csóka
Eötvös Loránd University, Budapest, Hungary

In the last decade it became apparent that a large number of the most interesting structures and phenomena of the world can be described by networks which are so large that the data about them can be collected only by indirect means like random local sampling. This yielded the motivation of property testing and parameter testing, which became an intensively studied field recently. A parameter tester of bounded-degree graphs means an algorithm which chooses a constant number of random nodes, and to these constant radius neighbourhoods, assigns an estimation (a number) which is at most ε far from the true parameter with at least $1-\varepsilon$ probability. We call a parameter testable if there exists a tester with arbitrary small errors.

There is a strongly connected concept called constant-time algorithm, introduced by Nguyen and Onak. For example, consider the maximum matching problem. Here, a constant-time algorithm is a "local function" that decides about each edge that whether it chooses to the mathing or not, so that the chosen edges form a matching, and its size is at most εn less than the size of the maximum matching, with at least $1-\varepsilon$ probability. Locality means that it depends only on the constant-size neighbourhood of the edge, including some random numbers, as follows. We assign independent random numbers to the nodes uniformly from $[0,1]$, and the neighbourhood consists not only of the induced subgraph of nodes at most constant far from the chosen node, but also of the random numbers of these nodes. In the paper of Nguyen and Onak, they showed a constant-time algorithm for this and some other problems.

About the connection of the two concepts, notice that if we have a constant-time algorithm producing the maximum matching then the ratio of the size of the maximum matching and the number of nodes is a testable parameter. Because we can make a tester which simply calculates the probability that the chosen random node would be covered by the matching produced by the constant-time algorithm, and the average of these probabilities is a good approximation.

In this paper, we show a constant-time *deterministic* algorithm for a version of the maximum flow problem. This determinism means that we will get the analogous result using no random numbers.

If we delete all edges from all sources or targets then the value of the maximum flow decreases to 0 while the distribution of the local neighbourhoods remains asymptotically the same. That is why the value of the maximum flow in a graph with 1, or even with $o(n)$ sources or targets cannot be tested in any reasonable way. Similarly, one new edge with high capacity between a source and a target would increase the value of the maximum flow by this arbitrary large value. These are some reasons why we will deal only with multiple sources and targets and bounded capacities.

The idea of our method is the following. We notice that if there are no short augmenting paths then the flow is almost maximal. That is why we use the Edmonds–Karp algorithm until we augmented on all paths with bounded length. We notice that if the algorithm augments on the same-length paths in random order then the flow on each edge can be determined with high probability only by its constant radius neighbourhood. Then we modify the algorithm to be surely local without changing the result so much. Thus we get an expectedly well-approximating constant-time random algorithm. As the space of flows is convex and the value of flows is linear, if we average on all random orderings, we get an approximating deterministic constant-time algorithm.