Efficient unique perfect matching algorithms

Miklós Bartha

Department of Computer Science
Memorial University of Newfoundland
St. John's, NL, Canada

`bartha@mun.ca`

Finding a unique perfect matching in graphs appears to be easier than finding an arbitrary one. Indeed, the fastest unique perfect matching algorithm by Gabow et. al. [1] runs in $\mathcal{O}(m \cdot \log^4 n)$ time, as opposed to the $\mathcal{O}(m \cdot \sqrt{n})$ time complexity of the general maximum matching algorithm by Micali and Vazirani [2]. (The number of vertices and edges in the graph is denoted by n and m, respectively.) We conjecture that a unique perfect matching, if it exists, can always be found in $\mathcal{O}(m)$ time, and present such a linear-time algorithm for some special classes of graphs having a unique perfect matching.

The UPM-(unique perfect matching) set of a factor-critical graph G is the set of those vertices v for which G has a unique near-perfect matching that misses v. Graph G is called UPM-*elementary* if, by taking a new collector vertex c and connecting c to each vertex of the UPM-set of G, we obtain an elementary graph. Clearly, if the UPM-set is non-empty, then it must contain at least two vertices unless G itself is an isolated vertex.

The motivation for studying UPM-elementary graphs is that they are the first step toward constructing a linear-time algorithm to decide if a given graph has a unique perfect matching and if so, to specify that matching. As a partial result we provide a simple characterization of UPM-elementary factor-critical graphs in terms of a reduction algorithm [3], and show that this property of graphs is indeed decidable in linear time.

References

[1] H. N. Gabow, H. Kaplan, R. E. Tarjan, Unique maximum matching algorithms, *Journal of Algorithms* **40** (2001), 159–183.

[2] S. Micali, V.V. Vazirani, An $\mathcal{O}(m \cdot \sqrt{n})$ algorithm for finding a maximum matching in general graphs, *Proc. 21st IEEE Symp. Foundations of Computer Science*, 1109/SFCS **10** 1980, pp. 17–27.

[3] M. Bartha, M. Krész, A depth-first algorithm to reduce graphs in linear time, *Proc. 11th SYNASC*, Timisoara, Romania, 2009.