Project Ideas

Election Due Date: October 16, 2020 2:59PM (CDT)

1. You should read the provided references to see which project best fit your interest or propose your own (option 4). However, proposing your own project would require you to convince me about its relevance to this course. There are no guarantees that it will be accepted. However feel free to discuss your idea (in theory/ algorithms) with me in case its turns out to be a good one.

2. We have not covered these topics in class. Hence you may find the materials difficult during your first reading. You should set up meetings with me if you need help with understanding the referenced papers.

3. Once you select a project you will have about two weeks to change your mind (not recommended) and select another one if you so desire. However you cannot change your topic later.

4. You must also provide bi-weekly update about your progress on your project. These can be verbal or written. This also a good time to discuss strategies with me. Depending on which project you choose and what you want to try within a project your research deliverables will be different. However you have to write a short paper about your work and give a 20-min presentation in class.

5. Projects may be done in groups of two.

6. Projects will be due early December and in leu of a final exam we will have class presentations. (Exact dates will be announced later)

Topic 1 Comparison Tree Complexity - Restricted sorting X is a set of n-elements with an unknown total order. Along with X we are given a graph $G(X, E)$. We are allowed to compare a pair of elements u, v in X only if they are adjacent in G. That is, uv is an edge in E. The classical problem of sorting n elements is a special case of this problem where the graph G is complete. Also the famous "matching nuts and bolts"problem is a special case where G is complete bipartite. If G is arbitrary then we do not know any better lower bound than the standard information theoretic bound of $\Omega(n \log n)$. Although there are some non-trivial upper bounds, they do not work for sparse graphs (where the number of edges are $o(n^2)$). In particular highly expanding bipartite graphs seem to be one of the hardest cases to deal with. In this project you will investigate the restricted sorting problem on special graph classes.

[Reference: J Komlos et al., Matching nuts and bolts in $O(n \log n)$ time]

Topic-2 : Online Computation The k-server problem is arguably the most famous problem in online computation. The k-server conjecture states that there is a k-competitive deterministic algorithm for this problem. The work function algorithm (WFA) gets close to resolving this conjecture. In fact it is believed that WFA is k-competitive for this problem. However, beyond some special cases it remains open whether WFA is k-competitive. WFA have been shown to be $2k - 1$ competitive for general metric spaces. Some special cases for which the conjecture may be possible to resolve are:

1. The metric space induced by a edge weighted tree. Or the simpler case where the points are only at the leaves.

2. When the number of servers is 3 but the metric space is arbitrary.

3. The circle metric with k servers.

[Reference: The k-server problem, E. Koutsoupias (2009)]

Topic-4: Graph Reconfiguration - Qbit routing Current quantum architectures have limited connectivity. In order to execute a logical quantum circuit the qbits are routed in the hardware quantum processor (using appropriate swap operations) before a gate operation can be applied on them. This problem can be formulated as a (partial) permutation routing on a undirected connected graph. Different physical assumption leads to different flavors of the problem. In most cases the goal is to find a routing scheme that minimizes the depth of the quantum circuit. In the permutation routing domain this translates in to a problem of minimizing the routing time. These problems are NP-hard except for the simplest of cases. In this project you will investigate approximation algorithms or give hardness results for the permutation routing model which uses matchings to route the qbits.

[References: 1) AM Childs et al., Circuit transformations for quantum architectures 2) N Alon et al., Routing permutations on graphs via matchings]

Topic-4: Open Propose your known project for acceptance. Your proposed problem should be of algorithmic in nature (similar to the ones given here). If you are proposing to solve a new algorithmic problem you have to justify its importance and discuss briefly your plan of action.