Albert-Ludwigs-Universität
Institut für Informatik
Prof. Dr. F. Kuhn
Mohamad Ahmadi, Hamid Ghodselahi June 08, 2016

Network Algorithms, Summer Term 2016
Problem Set 6

hand in by Wednesday, June 15, 2016

Exercise 1: Deterministic Maximal Independent Set Construction

In the lecture, we have seen a simple randomized algorithm to construct a maximal independent set (MIS) in expected $O(\log n)$ rounds. While an MIS can be computed very efficiently by a randomized distributed algorithm, finding a fast deterministic algorithm turns out to be a lot harder. In the following, our goal will be to get a deterministic algorithm with time complexity $o(n)$. We assume that nodes have unique identifiers that can be represented using $O(\log n)$ bits.
We first try to get an algorithm that is fast if the network graph has a small largest degree. Let Δ be the largest degree of the graph. Rather than directly computing an MIS, we first have a look at the problem of coloring the graph with $\Delta + 1$ colors. For simplicity, in all of the following questions, assume that all nodes know Δ, n, as well as the number of bits required to store the largest ID.

1. Assume that we have a partition V of the set of nodes V of the network graph G into two disjoint sets V_0 and V_1 (each node knows to which set it belongs). Further assume that we are given valid $(\Delta + 1)$-colorings of the graphs $G[V_0]$ and $G[V_1]$ induced by V_0 and V_1. Describe an algorithm that combines the colorings of $G[V_0]$ and $G[V_1]$ into a $(\Delta + 1)$-coloring of G. What is the time complexity of your algorithm?

2. Given node identifiers, a simple way to partition the nodes V into two sets V_0 and V_1 is to use, e.g., the first bit of the identifier (nodes with bit 0 go to V_0, nodes with bit 1 go to V_1). Use this partition idea and compute the $(\Delta + 1)$-colorings of $G[V_0]$ and $G[V_1]$ recursively. What is the time complexity of the resulting $(\Delta + 1)$-coloring algorithm for G?

3. Based on the coloring algorithm derived in (2), describe a deterministic MIS algorithm that is fast for networks with small largest degree. What is the time complexity of your MIS algorithm as a function of Δ (and n).

4. The algorithm so far is only fast if Δ is small. If we have a graph with large Δ, we need to first reduce the largest degree in order to be able to effectively apply the algorithm. Try to modify the simple distributed greedy algorithm from the lecture (Algorithm 29 – Slow MIS) so that it tries to reduce the largest degree in the graph. How long do you have to run the modified algorithm so that no node of degree larger than k remains?

5. Combine the algorithms of (3) and (4) to obtain an $o(n)$ time distributed deterministic MIS algorithm. What is the time complexity of your algorithm?

Exercise 2: (Local) Reductions

Many problems can be seen as—more or less obvious—variants of others and therefore can be solved by clever use of the same algorithms. In this exercise you may use the algorithms derived in the lecture as subroutines.

1. Given a graph $G = (V, E)$, a *dominating set* is a subset $D \subseteq V$ such that each node either is in D or has a neighbor in D. The *minimum dominating set* problem is to find a dominating set of minimum cardinality. Give a 3/2-approximation algorithm for this problem on rings which takes $O(\log^* n)$ time!

2. A family of graphs of *bounded independence* is a set of graphs where, for each node, the largest independent set in the one-hop neighborhood (i.e., the direct neighbors) has a size that is bounded by a constant C. Give a C-approximation algorithm to the minimum dominating set problem on graphs of bounded independence running in $O(\log n)$ time!