ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

FS 2008 Prof. R. Wattenhofer / Dr. F. Kuhn / T. Locher / Y. A. Oswald / C. Lenzen

Principles of Distributed Computing
Exercise 5

1 Greedy Dominating Set

The distributed version of the greedy dominating set (DS) algorithm presented in the lecture computes a $\ln \Delta$-approximation in $O(n)$ rounds.

Construct a graph $G = (V, E)$ for which the approximation ratio is as large as possible, i.e., the size of the computed DS is a factor $\Omega(\log \Delta)$ larger than the optimal DS! Try to find a graph for which Δ is as large as possible!

2 Fast Dominating Set

The second algorithm discussed in the lecture only needs $O(\log^2 \Delta \log n)$ rounds to compute an $O(\log \Delta)$-approximation. More precisely, the algorithm requires $O(\log^2 \Delta \log n)$ phases, where each phase (i.e., one iteration of the *while* loop) consists of a constant number of rounds.

Write down the communication steps (node v sends/receives ... to/from ...) for a single phase in detail! How many rounds are required exactly in each phase?

3 Dominating Set on Regular Graphs

Now we want to compute a DS on δ-regular graphs. A graph is called *δ-regular* if the degree of each node is δ. Consider the following algorithm:

Algorithm 1 DS Algorithm for δ-regular graphs.

```
With probability p = ln(δ+1)/(δ+1) join the DS
Send decision joined/not joined to neighbors
Receive decisions from all neighbors
if not joined and no neighbor joined then
    Join the DS
end if
```

a) What is the time complexity of this algorithm?

b) What is the expected number of nodes that join the DS?
Hint: Use the inequality $(1 - \frac{x}{n})^n \leq e^{-x}$ for $x < n \in \mathbb{N}$ to bound the probability that no neighbor joins the DS!

c) At least how many nodes have to join the DS (in an optimal solution)? Combine this result and b) to determine the expected approximation ratio of this algorithm!