A performance study of the greedy algorithm for dominating set

Jonathan Li*, Rohan Potru*, Texas Academy of Math and Science, UNT
Farhad Shahrokhi, UNT

Computing the smallest dominating set of a graph is known to be an NP-hard problem. The greedy algorithm for approximating this problem is known to have a worst case performance ratio of $O(\log(\Delta))$, where $\Delta$ is the maximum degree. It is known that improving the performance ratio $O(\log(\Delta))$ is also NP-hard. We report the performance of our linear time C++ implementation of the greedy algorithm by applying it to a variety of sparse graphs, including planar and k-planar graphs. To measure the performance ratio, the linear programming (LP) relaxation of the problem was also solved. Over the range of test data (1000-1000,000 vertices sparse graphs), the ratio of greedy solution to the LP solution was bounded by 1.26. An approximate dominating set on a planar graph of 1,000,000 vertices was computed in about 3.6 seconds of CPU time on a laptop with computational power (1.6GHz, 8GB RAM). The state of the art methods for the approximation of dominating set in sparse graphs require solving the LP relaxations and then rounding them. The results support the conjecture that the performance ratio of the greedy algorithm is bounded by a constant when applied to graphs with bounded arboricity. If the conjecture is true, then one could directly apply the greedy algorithm to sparse graphs instead of using LP. We also report the performance of a hybrid algorithm resulting from the combination of greedy and integer programming when applied to hypercubes.