Approximations of Independent Sets in Graphs

Magnús M. Halldórsson\textsuperscript{1,2}

\textsuperscript{1} Science Institute, University of Iceland, Reykjavik, Iceland. mmm@hi.is
\textsuperscript{2} Department of Informatics, University of Bergen, Norway.

1 Introduction

The independent set problem is that of finding a maximum size set of mutually non-adjacent vertices in a graph. The study of independent sets, and their alter egos, cliques, has had a central place in combinatorial theory.

Independent sets occur whenever we seek sets of items free of pairwise conflicts, e.g. when scheduling tasks. Aside from numerous applications (which might be more pronounced if the problems weren't so intractable), independent sets and cliques appear frequently in the theory of computing, e.g. in interactive proof systems [6] or monotone circuit complexity [2]. They form the representative problems for the class of subgraph or packing problems in graphs, are essential companions of graph colorings, and form the basis of clustering, whether in terms of nearness or dispersion.

As late as 1990, the literature on independent set approximations was extremely sparse. In the period since Johnson [31] started the study of algorithms with good performance ratios in 1974 – and in particular showed that a whole slew of independent set algorithms had only the trivial performance ratio of $n$ on general graphs – only one paper had appeared containing positive results [29], aside from the special case of planar graphs [34, 8]. Lower bounds were effectively non-existent, as while it was known that the best possible performance ratio would not be some fixed constant, there might still be a polynomial-time approximation scheme lurking somewhere.

Success on proving lower bounds for Independent Set has been dramatic and received worldwide attention, including the New York Times. Progress on improved approximation algorithms has been less dramatic, but a notable body of results has been developed. The purpose of this talk is to bring some of these results together, consider the lessons learned, and hypothesize about possible future developments.

The current paper is not meant to be the ultimate summary of independent set approximation algorithms, but an introduction to the performance ratios known, the strategies that have been applied, and offer glimpses of some of the results that have been proven.

We prefer to study a range of algorithms, rather than seek only the best possible performance guarantee. The latter is fine as far as it goes, but is not the only thing that matters; only so much information is represented by a single number. Algorithmic strategies vary in their time requirements, temporal access to data, parallelizability, simplicity and numerous other factors that are far from
irrelevant. Different algorithms may also be incomparable on different classes of graphs, e.g. depending on the size of the optimal solution. Finally, the proof techniques are perhaps the most valuable product of the analysis of heuristics.

We look at a slightly random selection of approximation results in the body of the paper. A complete survey is beyond the scope of this paper but is under preparation. The primary criteria for selection was simplicity, of the algorithm and the proof. We state some observations that have not formally appeared before, give some recent results, and present simpler proofs of other results.

The paper is organized as follows. We define relevant problems and definitions in the following section. In the body of the paper we present a number of particular results illustrating particular algorithmic strategies: subgraph removal, semi-definite programming, partitioning, greedy algorithms and local search. We give a listing of known performance results and finish with a discussion of open issues.

2 Problems and definitions

**INDEPENDENT SET**: Given a graph $G = (V, E)$, find a maximum cardinality set $I \subseteq V$ such that for each $u, v \in I$, $(u, v) \notin E$. The *independence number* of $G$, denoted by $\alpha(G)$, is the size of the maximum independent set.

**CLIQUE PARTITION**: Given a graph $G = (V, E)$, find a minimum cardinality set of disjoint cliques from $G$ that contains every vertex.

**$\kappa$-SET PACKING**: Given a collection $C$ of sets of size at most $\kappa$ drawn from a finite set $S$, find a minimum cardinality collection $C'$ such that each element in $S$ is contained in some set in $C'$.

These problems may also be weighted, with weights on the vertices (or on the sets in SET PACKING).

A set packing instance is a case of an independent set problem. Given a set system $(C, S)$, form a graph with a vertex for each set in $C$ and edge between two vertices if the corresponding sets intersect. Observe that if the sets in $C$ are of size at most $\kappa$, then the graph contains a $\kappa + 1$-claw, which is a subgraph consisting of a center node adjacent to $\kappa + 1$ mutually non-adjacent vertices. The independent set problem in $\kappa + 1$-claw free graphs slightly generalizes $\kappa$-SET PACKING, which in turn slightly generalizes $\kappa$-DIMENSIONAL MATCHING.

The performance ratio $\rho_A$ of an independent set algorithm $A$ is given by

$$\rho_A = \rho_A(n) = \max_{G : |V| = n} \frac{\alpha(G)}{A(G)}.$$