Exact Algorithms for Dominating Set

2005; Fomin, Grandoni, Kratsch

Dieter Kratsch
UFR MIM – D'informatique, Université de Metz,
Metz, France

Synonyms

Connected dominating set

Problem Definition

The dominating set problem is a classical NP-hard optimization problem which fits into the broader class of covering problems. Hundreds of papers have been written on this problem that has a natural motivation in facility location.

Definition 1 For a given undirected, simple graph $G = (V, E)$ a subset of vertices $D \subseteq V$ is called a *dominating set* if every vertex $u \in V - D$ has a neighbor in D. The minimum dominating set (MDS) problem is to find a *minimum dominating set* of G, i. e. a dominating set of G of minimum cardinality.

Problem 1 (MDS)
Input: *Undirected simple graph* $G = (V, E)$.
Output: *A minimum dominating set* D *of* G.

Various modifications of the dominating set problem are of interest, some of them obtained by putting additional constraints on the dominating set such as, for example, requesting it to be an independent set or to be connected. In graph theory there is a huge literature on domination dealing with the problem and its many modifications (see e. g. [9]). In graph algorithms the MDS problem and some of its modifications like Independent Dominating Set and Connected Dominating Set have been studied as benchmark problems for attacking NP-hard problems under various algorithmic approaches.

Known Results

The algorithmic complexity of MDS and its modifications when restricted to inputs from a particular graph class has been studied extensively (see e. g. [10]). Among others, it is known that MDS remains NP-hard on bipartite graphs, split graphs, planar graphs and graphs of maximum degree three. Polynomial time algorithms to compute a minimum dominating set are known, for example, for permutation, interval and k-polygon graphs. There is also a $O(4^k n^{O(1)})$ time algorithm to solve MDS on graphs of treewidth at most k.

The dominating set problem is one of the basic problems in parameterized complexity [3]; it is W[2]-complete and thus it is unlikely that the problem is fixed parameter tractable. On the other hand, the problem is fixed parameter tractable on planar graphs. Concerning approximation, MDS is equivalent to Minimum Set Cover under L-reductions. There is an approximation algorithm solving MDS within a factor of $1 + \log|V|$ and it cannot be approximated within a factor of $(1 - \epsilon)\ln|V|$ for any $\epsilon > 0$, unless NP $\subset$ DTIME$(n^{\log\log n})$ [1].

Moderately Exponential Time Algorithms

If P $\neq$ NP then no polynomial time algorithm can solve MDS. Even worse, it has been observed in [7] that unless SNP$\subseteq$ SUBEXP (which is considered to be highly unlikely), there is not even a subexponential time algorithm solving the dominating set problem.

The trivial $O(2^n(n+m))$ algorithm, which simply checks all the 2^n vertex subsets as to whether they are dominating, clearly solves MDS. Three faster algorithms were established in 2004. The algorithm of Fomin et al. [7] uses a deep graph-theoretic result due to B. Reed, stating that every graph on n vertices with minimum degree at least three has a dominating set of size at most $3n/8$, to establish an $O(2^{0.955n})$ time algorithm solving MDS. The $O(2^{0.919n})$ time algorithm of Randerath and Schiermeyer [11] uses very nice ideas including matching techniques to restrict the search space. Finally, Grandoni [8] established an $O(2^{0.850n})$ time algorithm to solve MDS.

The work of Fomin, Grandoni, and Kratsch [5] presents a simple and easy to implement recursive branch & reduce algorithm to solve MDS. The running time of the algorithm is significantly faster than the ones stated for previous algorithms. This is heavily based on the analysis of the running time by measure & conquer, which is a method to analyze the worst case running time of (simple) branch & reduce algorithms based on a sophisticated choice of the measure of a problem instance.

Key Results

Theorem 1 *There is a branch & reduce algorithm solving MDS in time* $O(2^{0.610n})$ *using polynomial space.*

Theorem 2 *There is an algorithm solving MDS in time* $O(2^{0.598n})$ *using exponential space.*

The algorithms of Theorem 1 and 2 are simple consequences of a transformation from MDS to Minimum Set Cover

(MSC) combined with new moderately exponential time algorithms for MSC.

Problem 2 (MSC)
Input: *Finite set $\mathcal{U}$ and a collection $\mathcal{S}$ of subsets $S_1, S_2, \ldots S_t$ of $\mathcal{U}$.*
Output: *A minimum set cover $\mathcal{S}'$, where $\mathcal{S}' \subseteq \mathcal{S}$ is a set cover of $(\mathcal{U}, \mathcal{S})$ if $\bigcup_{S_i \in \mathcal{S}'} S_i = \mathcal{U}$.*

Theorem 3 *There is a branch & reduce algorithm solving MSC in time $O(2^{0.305(|\mathcal{U}|+|\mathcal{S}|)})$ using polynomial space.*

Applying memorization to the polynomial space algorithm of Theorem 3 the running time can be improved as follows.

Theorem 4 *There is an algorithm solving MSC in time $O(2^{0.299(|\mathcal{S}|+|\mathcal{U}|)})$ using exponential space.*

The analysis of the worst case running time of the simple branch & reduce algorithm solving MSC (of Theorem 3) is done by a careful choice of the measure of a problem instance which allows one to obtain an upper bound that is significantly smaller than the one that could be obtained using the standard measure. The refined analysis leads to a collection of recurrences. Then random local search is used to compute the weights, used in the definition of the measure, aiming at the best achievable upper bound of the worst-case running time.

Since current tools to analyze the worst-case running time of branch & reduce algorithms do not seem to produce tight upper bounds, exponential lower bounds of the worst-case running time of the algorithm are of interest.

Theorem 5 *The worst-case running time of the branch & reduce algorithm solving MDS (see Theorem 1) is $\Omega(2^{n/3})$.*

Applications

There are various other NP-hard domination-type problems that can be solved by a moderately exponential time algorithm based on an algorithm solving MINIMUM SET COVER: any instance of the initial problem is transformed to an instance of MSC (preferably with $|\mathcal{U}| = |\mathcal{S}|$), and then the algorithm of Theorem 3 or 4 is used to solve MSC and thus the initial problem. Examples of such problems are TOTAL DOMINATING SET, k-DOMINATING SET, k-CENTER and MDS on split graphs.

Measure & Conquer and the strongly related quasi-convex analysis of Eppstein [4] have been used to design and analyze a variety of moderately exponential time algorithms for NP-hard problems: optimization, counting and enumeration problems. See for example [2,6].

Open Problems

A number of problems related to the work of Fomin, Grandoni, and Kratsch remain open. Although for various graph classes there are algorithms to solve MDS which are faster than the one for general graphs (of Theorem 1 and 2), no such algorithm is known for solving MDS on bipartite graphs.

The worst-case running times of simple branch & reduce algorithms like those solving MDS and MSC remain unknown. In the case of the polynomial space algorithm solving MDS there is a large gap between the $O(2^{0.610n})$ upper bound and the $\Omega(2^{n/3})$ lower bound. The situation is similar for other branch & reduce algorithms. Consequently, there is a strong need for new and better tools to analyze the worst-case running time of branch & reduce algorithms.

Cross References

► Vertex Cover Search Trees
► Data Reduction for Domination
► Connected Dominating Set

Recommended Reading

1. Ausiello, G., Crescenzi, P., Gambosi, G., Kann, V., Marchetti-Spaccalema, A., Protasi, M.: Complexity and Approximation. Springer, Berlin (1999)
2. Byskov, J.M.: Exact algorithms for graph colouring and exact satisfiability. PhD thesis, University of Aarhus, Denmark (2004)
3. Downey, R.G., Fellows, M.R.: Parameterized complexity. Springer, New York (1999)
4. Eppstein, D.: Quasiconvex analysis of backtracking algorithms. In: Proceedings of SODA 2004, pp. 781–790
5. Fomin, F.V., Grandoni, F., Kratsch, D.: Measure and conquer: Domination – A case study. In: Proceedings of ICALP 2005. LNCS, vol. 3380, pp. 192–203. Springer, Berlin (2005)
6. Fomin, F.V., Grandoni, F., Kratsch, D.: Measure and Conquer: A simple $O(2^{0.288n})$ Independent Set Algorithm. In: Proceedings of SODA 2006, pp. 18–25
7. Fomin, F.V., Kratsch, D., Woeginger, G.J.: Exact (exponential) algorithms for the dominating set problem. In: Proceedings of WG 2004. LNCS, vol. 3353, pp. 245–256. Springer, Berlin (2004)
8. Grandoni, F.: Exact Algorithms for Hard Graph Problems. PhD thesis, Università di Roma "Tor Vergata", Roma, Italy (2004)
9. Haynes, T.W., Hedetniemi, S.T., Slater, P.J.: Fundamentals of domination in graphs. Marcel Dekker, New York (1998)
10. Kratsch, D.: Algorithms. In: Haynes, T., Hedetniemi, S., Slater, P. (eds.) Domination in Graphs: Advanced Topics, pp. 191–231. Marcel Dekker CE2 (1998)
11. Randerath, B., Schiermeyer, I.: Exact algorithms for MINIMUM DOMINATING SET. Technical Report, zaik-469, Zentrum für Angewandte Informatik Köln (2004)
12. Woeginger, G.J.: Exact algorithms for NP-hard problems: A survey. In: Combinatorial Optimization – Eureka, You Shrink. LNCS, vol. 2570, pp. 185–207. Springer, Berlin (2003)

CE2 Please provide publisher location.

Editor's or typesetter's annotations (will be removed before the final TEX run)