Exact (Exponential) Algorithms for the Dominating Set Problem

Fedor V. Fomin[1,⋆], Dieter Kratsch[2], and Gerhard J. Woeginger[3]

[1] Department of Informatics, University of Bergen, N-5020 Bergen, Norway
fomin@ii.uib.no

[2] LITA, Université de Metz, 57045 Metz Cedex 01, France
kratsch@sciences.univ-metz.fr

[3] Department of Mathematics and Computer Science, TU Eindhoven,
P.O. Box 513, 5600 MB Eindhoven, The Netherlands
g.j.woeginger@tue.nl

Abstract. We design fast exact algorithms for the problem of computing a minimum dominating set in undirected graphs. Since this problem is NP-hard, it comes with no big surprise that all our time complexities are exponential in the number n of vertices. The contribution of this paper are 'nice' exponential time complexities that are bounded by functions of the form c^n with reasonably small constants $c < 2$: For arbitrary graphs we get a time complexity of 1.93782^n. And for the special cases of split graphs, bipartite graphs, and graphs of maximum degree three, we reach time complexities of 1.41422^n, 1.73206^n, and 1.51433^n, respectively.

1 Introduction

Nowadays, it is common believe that NP-hard problems can not be solved in polynomial time. For a number of NP-hard problems, we even have strong evidence that they cannot be solved in sub-exponential time. For these problems the only remaining hope is to design exact algorithms with good exponential running times. How good can these exponential running times be? Can we reach 2^{n^2} for instances of size n? Can we reach 10^n? Or even 2^n? Or can we reach c^n for some constant c that is very close to 1? The last years have seen an emerging interest in attacking these questions for concrete combinatorial problems: There is an $O^*(1.2108^n)$ time algorithm for independent set (Robson [13]); an $O^*(2.4150^n)$ time algorithm for graph coloring (Eppstein [4]); an $O^*(1.4802^n)$ time algorithm for 3-Satisfiability (Dantsin & al. [2]). We refer to the survey paper [14] by Woeginger for an up-to-date overview of this field. In this paper, we study the *dominating set* problem from this exact (exponential) algorithms point of view.

Basic Definitions. Let $G = (V, E)$ be an undirected, simple graph without loops. We denote by n the number of vertices of G. The open *neighborhood* of a vertex v is denoted by $N(v) = \{u \in V : \{u, v\} \in E\}$, and the closed

⋆ F. Fomin is supported by Norges forskningsråd project 160778/V30.

J. Hromkovič, M. Nagl, and B. Westfechtel (Eds.): WG 2004, LNCS 3353, pp. 245–256, 2004.
© Springer-Verlag Berlin Heidelberg 2004

neighborhood of v is denoted by $N[v] = N(V) \cup \{v\}$. The degree of a vertex v is $|N(v)|$. For a vertex set $S \subseteq V$, we define $N[S] = \bigcup_{v \in S} N[v]$ and $N(S) = N[S] - S$. The subgraph of G induced by S is denoted by $G[S]$. We will write $G - S$ short for $G[V - S]$. A set $S \subseteq V$ of vertices is a *clique*, if any two of its elements are adjacent; S is *independent*, if no two of its elements are adjacent; S is a *vertex cover*, if $V - S$ is an independent set.

Throughout this paper we use the so-called *big-Oh-star* notation, a modification of the big-Oh notation that suppresses polynomially bounded terms: We will write $f = O^*(g)$ for two functions f and g, if $f(n) = O(g(n)\text{poly}(n))$ holds with some polynomial $\text{poly}(n)$. We say that a problem is solvable in *sub-exponential* time in n, if there is an effectively computable monotone increasing function $g(n)$ with $\lim_{n\to\infty} g(n) = \infty$ such that the problem is solvable in time $O(2^{n/g(n)})$.

The Dominating Set Problem. Let $G = (V, E)$ be a graph. A set $D \subseteq V$ with $N[D] = V$ is called a *dominating set* for G; in other words, every vertex in G must either be contained in D or adjacent to some vertex in D. A set $A \subseteq V$ *dominates* a set $B \subseteq V$ if $B \subseteq N[A]$. The *domination number* $\gamma(G)$ of a graph G is the cardinality of a smallest dominating set of G. The *dominating set problem* asks to determine $\gamma(G)$ and to find a dominating set of minimum cardinality. The dominating set problem is one of the fundamental and well-studied classical NP-hard graph problems (Garey & Johnson [6]). For a large and comprehensive survey on domination theory, we refer the reader to the books [8, 9] by Haynes, Hedetniemi & Slater. The dominating set problem is also one of the basic problems in parameterized complexity (Downey & Fellows [3]); it is contained in the parameterized complexity class W[2]. Further recent investigations of the dominating set problem can be found in Albers & al. [1] and in Fomin & Thilikos [5].

Results and Organization of This Paper. What are the best time complexities for dominating set in n-vertex graphs that we can possibly hope for? Well, of course there is the trivial $O^*(2^n)$ algorithm that simply searches through all the 2^n subsets of V. But can we hope for a sub-exponential time algorithm, maybe with a time complexity of $O^*(2^{\sqrt{n}})$? Section 2 provides the answer to this question: No, probably not, unless some very unexpected things happen in computational complexity theory ... Hence, we should only hope for time complexities of the form $O^*(c^n)$, with some small value $c < 2$. And indeed, Section 3 presents such an algorithm with a time complexity of $O^*(1.93782^n)$. This algorithm combines a recursive approach with a deep result from extremal graph theory. The deep result is due to Reed [12], and it provides an upper bound on the domination number of graphs of minimum degree three.

Furthermore, we study exact exponential algorithms for the dominating set problem on some special graph classes: In Section 4, we design an $O^*(1.41422^n)$ time algorithm for split graphs, and an $O^*(1.73206^n)$ time algorithm for bipartite graphs. In Section 5, we derive an $O^*(1.51433^n)$ time algorithm for graphs of maximum degree three. Note that for these three graph classes, the dominating set problem remains NP-hard (Garey & Johnson [6], Haynes, Hedetniemi & Slater [9]).