NONBLOCKER: Parameterized Algorithmics for MINIMUM DOMINATING SET

Frank Dehne[1], Michael Fellows[2], Henning Fernau[2,3],
Elena Prieto[2], Frances Rosamond[2]

[1] School of Computer Science
Carleton University, Canada
[2] School of Electrical Engineering and Computer Science,
The University of Newcastle, Australia
[3] also: University of Hertfordshire, College Lane,
Hatfield, Herts AL10 9AB, UK
and: Universität Tübingen, WSI für Informatik, Sand 13,
72076 Tübingen, Germany

May 9, 2005

We provide parameterized algorithms for NONBLOCKER, the parametric dual of the well known DOMINATING SET problem. We exemplify three methodologies for deriving parameterized algorithms that can be used in other circumstances as well, including the (*i*) use of extremal combinatorics (known results from graph theory) in order to obtain very small kernels, (*ii*) use of known exact algorithms for the (nonparameterized) MINIMUM DOMINATING SET problem, and (*iii*) use of exponential space. Parameterized by the size k_d of the non-blocking set, we obtain an algorithm that runs in time $\mathcal{O}^*(2.5154^{k_d})$. We also discuss planar and bipartite special graph classes.

1 Introduction

The problem of finding a minimum dominating set in a graph is arguably one of the most important combinatorial problems on graphs, having, together with its variants, numerous applications and offering various lines of research [10]. The problem of finding a set of at most k vertices dominating the whole n-vertex graph is not only $\mathcal{N}P$-complete but also hard to approximate [2, 9]. Moreover, this problem is also intractable when viewed as a parameterized problem [6]. The status is different if the problem is to find a set of at most $k = n - k_d$ vertices dominating a given n-vertex graph, where k_d ($k-$dual) is considered the parameter. Our focus in this paper is to present a new $\mathcal{O}^*(2.5154^{k_d})$-algorithm for this dual problem which we will henceforth call the NONBLOCKER problem.

On graphs of degree at least one, Ore [13] has shown (using different terminology) that the NONBLOCKER problem admits a kernel of size $2k_d$. Ore's result was improved by McCuaig and Shepherd [11] for graphs with minimum degree two; in fact, their result was a sort of corollary to the classification of graphs that satisfy Ore's inequality with equality. Independently, the result had been discovered by the Russian mathematician Blank [3] more than fifteen years ago, as noticed by Reed in [15]. More precisely, McCuaig and Shepherd have shown:

Theorem 1 *If a connected graph $G = (V, E)$ has minimum degree two and it not one of seven exceptional graphs (each of them having at most seven vertices), then the size of its minimum dominating set is at most* $2/5 \cdot |V|$.

Our paper provides two algorithms for the NONBLOCKER problem. Our second algorithm incorporates the results of Blank and McCuaig and Shepherd.

The first algorithm we present is easy to implement, addressing an important need of real programmers. It essentially consists only of exhaustively applying simple data reduction (preprocessing) rules and then doing brute-force search in the reduced problem space. (The mathematical analysis of our simple algorithm is quite involved and non-trivial, however.)

Our data reduction rules make use of several novel technical features. We introduce a special annotated *catalytic vertex*, a vertex which is doomed to be in the dominating set we are going to construct. The catalytic vertex is introduced by a *catalyzation rule* which is applied only once. The graph is reduced and when no further reduction rules are applicable, a special *de-catalyzation rule* is applied. The *de-catalyzation rule* also is applied only once. In the next section we provide a brief discussion of our approach and definitions. In Section 3 we describe our reduction rules and the catalytic vertex and provide our first kernelization algorithm for the NONBLOCKER problem. In Sections 4 and 5 we show two further ways to exploit the fact that we have derived a very small kernel for NONBLOCKER: we make use of known exact algorithms for the (nonparameterized) MINIMUM DOMINATING SET problem as exhibited in [7], and we employ exponential space (by memoization techniques or by assuming appropriate precomputations) to get the running time of our algorithm down to almost $\mathcal{O}^*(2.5^{k_d})$. In Sec. 6 we conclude with a review of open problems.

2 Definitions

We first describe the setting in which we will discuss MINIMUM DOMINATING SET in the guise of NONBLOCKER.

A *parameterized problem* $\mathcal{P}$ is a subset of $\Sigma^* \times \mathbb{N}$, where Σ is a fixed alphabet and $\mathbb{N}$ is the set of all non-negative integers. Therefore, each instance of the parameterized problem $\mathcal{P}$ is a pair (I, k), where the second component k is called the *parameter*. The language $L(\mathcal{P})$ is the set of all YES-instances of $\mathcal{P}$. We say that the parameterized problem $\mathcal{P}$ is *fixed-parameter tractable* [6] if there is an algorithm that decides whether an input (I, k) is a member of $L(\mathcal{P})$ in time $f(k)|I|^c$, where c is a fixed constant and $f(k)$ is a recursive function independent of the overall input length $|I|$. The class of all fixed-parameter tractable problems is denoted by $\mathcal{FPT}$.

The dual problems of DOMINATING SET and NONBLOCKER are defined as follows:

Problem name: DOMINATING SET (DS)
Given: A graph $G = (V, E)$
Parameter: a positive integer k
Output: Is there a *dominating set* $D \subseteq V$ with $|D| \leq k$?

Problem name: NONBLOCKER (NB)
Given: A graph $G = (V, E)$
Parameter: a positive integer k_d
Output: Is there a *non-blocking set* $N \subseteq V$ with $|N| \geq k_d$?

A subset of vertices V' such that every vertex in V' has a neighbor in $V \setminus V'$ is called a *non-blocking set*. Observe that the complement of a non-blocking set is a dominating set and vice versa. Hence, $G = (V, E)$ has a dominating set of size at most k if and only if G has a non-blocking set of size at least $k_d = n - k$. Due to this behavior, DOMINATING SET and NONBLOCKER are called *parametric duals*.

Let $\mathcal{P}$ be a parameterized problem. A *kernelization* is a function K that is computable in polynomial time and maps an instance (I, k) of $\mathcal{P}$ onto an instance (I', k') of $\mathcal{P}$ such that

- (I, k) is a YES-instance of $\mathcal{P}$ if and only if (I', k') is a YES-instance of $\mathcal{P}$
- $|I'| \leq f(k)$, and
- $k' \leq g(k)$ for arbitrary functions f and g.

The instance (I', k') is called the *kernel* (of I). The importance of these notions for parameterized complexity is due to the following characterization.

Theorem 2 *A parameterized problem is in $\mathcal{FPT}$ iff it is kernelizable.*

Hence, in order to develop $\mathcal{FPT}$-algorithms, finding kernelizations can be seen as the basic methodology. The search for a small kernel often begins with finding local reduction rules. The reduction rules reduce the size of the instance to which they are applied; they are exhaustively applied and finally yield the kernelization function. In this paper we introduce a small variation of this method; namely, we introduce a catalyzation and a (de-)catalyzation rule, both of which are applied only once. Contrary to our usual reduction rules, these two special rules increase the instance size.

We use this approach to solve the following *Catalytic Conversion* form of the problem.

Problem name: NONBLOCKER WITH CATALYTIC VERTEX (NBCAT)
Given: A graph $G = (V, E)$, a catalytic vertex c
Parameter: a positive integer k_d
Output: Is there a *non-blocking set* $N \subseteq V$ with $\|N\| \geq k_d$ such that $c \notin N$?

The catalytic conversion rules are special purpose reduction rules that allow vertices of small degree, including degree one, to be taken into the non-blocking set. The special annotated catalytic vertex is assumed to be in the dominating set (not the non-blocking set).

3 An $\mathcal{FPT}$ algorithm for NONBLOCKER using catalytic conversion

Our kernelization algorithm for solving NONBLOCKER uses the following two special rules to introduce and then to delete the catalytic vertex.

Reduction rule 1 (Catalyzation rule) *If* (G, k_d) *is an instance of* NONBLOCKER *with* $G = (V, E)$*, then* (G', c, k_d) *is an equivalent instance of* NONBLOCKER WITH CATALYTIC VERTEX*, where* $c \notin V$ *is a new vertex, and* $G' = (V \cup \{c\}, E)$*.*

Reduction rule 2 (De-catalyzation rule) *Let* (G, c, k_d) *be an instance of* NONBLOCKER WITH CATALYTIC VERTEX*. Then, perform the following surgery to obtain a new instance* (G', k'_d) *of* NONBLOCKER*:*

Connect the catalytic vertex c *to three new vertices* u*,* v*, and* w *by edges; moreover, introduce new edges* uv *and* vw*. All other vertices and edge relations in* G *stay the same. This describes the new graph* G'*. Set* $k'_d = k_d + 3$*. We may then forget about the special role of the catalyst.*

Our preprocessing uses the following reduction rules.

Reduction rule 3 (The Isolated Vertex Rule) *Let* (G, c, k_d) *be an instance of* NBCAT*. If* C *is a complete graph component of* G *that does not contain* c*, then reduce to* $(G - C, c, k_d - (|C| - 1))$*.*

Observe that Rule 3 applies to isolated vertices. It also applies to instances that do not contain a catalytic vertex. A formal proof of the soundness of the rule is contained in [14]. Notice that this rule alone gives a $2k_d$ kernel for general graphs with the mentioned result of Ore (details are shown below). We further improve on $2k_d$ kernel size by making use of the mentioned results of McCuaig and Shepard, that is, by getting rid of vertices of degree one.

Reduction rule 4 (The Catalytic Rule) *Let (G, c, k_d) be an instance of* NONBLOCKER WITH CATALYTIC VERTEX. *Let $v \neq c$ be a vertex of degree one in G with $N(v) = u$. Transform (G, c, k_d) into $(G', c', k_d - 1)$, where:*

- *If $u \neq c$ then $G' = G_{[c \leftrightarrow u]} \setminus v$, i.e., G' is the graph obtained by deleting v and merging u and c into a new catalytic vertex $c' = \langle c \leftrightarrow u \rangle$.*
- *If $u = c$ then $G' = G \setminus v$ and $c' = c$.*

Lemma 3 *Rule 4 is sound.*

Proof. "Only if:" Let (G, c, k_d) be an instance of NBCAT. Let $V' \subset V(G)$ be a non-blocking set in G with $|V'| = k_d$. The vertex v is a vertex of degree one in G. Let u be the neighbor of v in G. Two cases arise:

1. If $v \in V'$ then it must have a neighbor in $V(G) \setminus V'$ and thus $u \in V(G) \setminus V'$. Deleting v will decrease the size of V' by one. If $u = c$, then $(G', c', k_d - 1)$ is a YES-instance of NBCAT. If $u \neq c$, merging u and c will not affect the size of V' as both vertices are now in $V(G') \setminus V'$. Thus, $(G', c', k_d - 1)$ is a YES-instance of NBCAT.
2. If $v \in V(G) \setminus V'$, then two cases arise:
   - 2.1. If u is also in $V(G) \setminus V'$ then deleting v does not affect the size of V'. Note that this argument is valid whether $u = c$ or $u \neq c$.
   - 2.2. If $u \in V'$ then $u \neq c$. If we make $v \in V'$ and $u \in V(G) \setminus V'$, the size of V' remains unchanged. Since u did not dominate any vertices in the graph, this change does not affect $N(u) \setminus v$, and Case 1 now applies.

"If:" Conversely, assume that $(G', c', k_d - 1)$ is a YES-instance of NBCAT. Two cases arise:

1. If $u = c$, then we can always place v in V' and thus (G, c, k_d) is a YES-instance for NONBLOCKER WITH CATALYTIC VERTEX.
2. If $u \neq c$, getting from G' to G can be seen as (1) splitting the catalytic vertex c' into two vertices c and u, (2) taking c as the new catalytic vertex, and (3) attaching a pendant vertex v to u. As the vertex u is in $V(G) \setminus V'$, v can always be placed in V', increasing the size of this set by one. Thus (G, c, k_d) is a YES-instance for NBCAT, concluding the proof of Lemma 4.

∎

Reduction Rule 4 can be generalized as follows:

Reduction rule 5 (The Small Degree Rule) *Let (G, c, k_d) be an instance of* NONBLOCKER WITH CATALYTIC VERTEX. *Whenever you have a vertex $x \in V(G)$ whose neighborhood contains a non-empty subset $U \subseteq N(x)$ such that $N(U) \subseteq U \cup \{x\}$ and $c \notin U$, then you can merge x with the catalytic vertex c and delete U (and reduce the parameter by $|U|$).*

The overall kernelization algorithm for a given NONBLOCKER-instance (G, k_d) is the procedure listed in Algorithm 1.

Algorithm 1 A kernelization algorithm for NONBLOCKER

Input(s): an instance (G, k_d) of NONBLOCKER
Output(s): an equivalent instance (G', k'_d) of NONBLOCKER with $V(G') \subseteq V(G)$, $|V(G')| \leq 5/3 \cdot k'_d$ and $k'_d \leq k_d$ OR YES

```
if G has more than seven vertices then
   Apply the catalyzation rule.
   Exhaustively apply Rules 3 and 5 for neighborhoods U up to size two.
   Apply the de-catalyzation rule.
   {This leaves us with a reduced instance (G', k'_d).}
   if |V(G')| > 5/3 · k'_d then
      return YES
   else
      return (G', k'_d)
   end if
else
   Solve by table look-up and answer accordingly.
end if
```

Without further discussion, we now state those reduction rules that can be used to get rid of all consecutive degree-2-vertices in a graph:

Reduction rule 6 (The Degree Two Rule) *Let (G, c, k_d) be an instance of* NBCAT. *Let u, v be two vertices of degree two in G such that $u \in N(v)$ and $|N(u) \cup N(v)| = 4$, i.e., $N(u) = \{u', v\}$ and $N(v) = \{v', u\}$ for some $u' \neq v'$. If $c \notin \{u, v\}$, then merge u' and v' and delete u and v to get a new instance $(G', c', k_d - 2)$. If u' or v' happens to be c, then c' is the merger of u' and v'; otherwise, $c' = c$.*

Reduction rule 7 (The Degree Two, Catalytic Vertex Rule) *Let (G, c, k_d) be an instance of* NBCAT*, where $G = (V, E)$. Assume that c has degree two and a neighboring vertex v of degree two, i.e., $N(v) = \{v', c\}$. Then, delete the edge vv'. Hence, we get the new instance $((V, E \setminus \{vv'\}), c, k_d)$.*

Notice that all cases of two subsequent vertices u, v of degree two are covered in this way:

- If u or v is the catalytic vertex, then Rule 7 applies.
- Otherwise, if u and v have a common neighbor x, then Rule 5 is applicable; x will be merged with the catalytic vertex.

- Otherwise, Rule 6 will apply.

This allows us to eliminate all of the exceptional graphs of Theorem 1 (since all of them have two consecutive vertices of degree two).

Corollary 4 *Alg. 1 provides a kernel of size upperbounded by* $5/3 \cdot k_d + 3$ *for any* NONBLOCKER*-instance* (G, k_d)*, where the problem size is measured in terms of the number of vertices.*

4 Searching the space

4.1 Brute force

With a very small kernel, the remaining reduced NONBLOCKER-instance can be solved by brute-force search. Hence, we have to test all subsets of size k_d within the set of vertices of size at most $5/3 \cdot k_d$.

Stirling's formula gives:

Lemma 5 *For any* $a > 1$*,*

$$\binom{ak}{k} \approx a^k \left(\frac{a}{a-1}\right)^{(a-1)k}$$

Using Lemma 5, we can conclude:

Corollary 6 *By testing all subsets of size* k_d *of a reduced instance* (G, k_d) *of* NONBLOCKER*, the* NONBLOCKER *problem can be solved in time* $\mathcal{O}^*(3.0701^{k_d})$*.*

4.2 Using nonparameterized exact algorithmics

The above corollary can be slightly improved by making use of the following recent result of F. Fomin, D. Kratsch, and G. Woeginger [7] on general graphs:

Theorem 7 MINIMUM DOMINATING SET *can be solved in time* $\mathcal{O}^*(1.9379^n)$ *on arbitrary* n*-vertex graphs.*

Due to the $5/3 \cdot k_d$-kernel for NONBLOCKER, we conclude:

Corollary 8 *By applying the algorithm of Fomin, Kratsch and Woeginger [7] to solve* MINIMUM DOMINATING SET *on a reduced instance* (G, k_d) *of* NONBLOCKER*, the* NONBLOCKER *problem can be solved in time* $\mathcal{O}^*(3.0121^{k_d})$*.*

4.3 Trading time and space

Due to the fact that the kernel we obtained for NONBLOCKER is very small, it may be worthwhile looking for an algorithm that uses exponential space, as explained in [4, 12] for the case of VERTEX COVER. The basic algorithm would then be the following Algorithm 2

Algorithm 2 A sketch of an exponential-space algorithm for NONBLOCKER

Input(s): a graph $G = (V, E)$, k_d giving the size of the non-blocking set we are after

Output(s): YES iff G has a non-blocking set of size k_d

Kernelize according to Algorithm 1.
{Let k_d be also the new parameter value and $G = (V, E)$ the graph.}
Determine cut-off value α.
for all $X \subseteq V$ of size at most αk_d **do**
  Determine maximum non-blocking set with the algorithm of Fomin, Kratsch, and Woeginger. Put its size in a table OPT(X).
end for
Branch using the algorithm of Fomin, Kratsch, and Woeginger up to graphs with at most αk_d vertices.
For small graphs, look the solution up within OPT.

Computing the cut-off value α is straightforward, requiring balancing the time spent to compute the entries of OPT against the savings of the search-tree. If c is the exponential base of the algorithm of Fomin, Kratsch and Woeginger, then the initialization needs

$$\sum_{j=1}^{\alpha k_d} c^j \binom{5/3 \cdot k_d}{\alpha k_d} \approx c^{\alpha k_d + 1} \binom{5/3 \cdot k_d}{\alpha k_d}$$

time. Letting $\ell = \alpha k_d$, Lemma 5 gives

$$\binom{5/(3\alpha) \cdot \ell}{\ell} \approx (5/(3\alpha))^{\ell} \left(\frac{5/(3\alpha)}{5/(3\alpha) - 1} \right)^{(5/(3\alpha)-1)\ell}.$$

Since we stop the branching of the "usual" search-tree algorithm when the graph has αk_d vertices or less, the corresponding time has shrunk down to $\mathcal{O}^*(c^{(5/3-\alpha)\cdot k_d})$. Depending on c (for general graphs, $c \approx 1.93\ldots$) we get different cut-off values that are computed by equating:

$$(5/(3\alpha))^{\alpha k_d} \left(\frac{5/(3\alpha)}{5/(3\alpha) - 1} \right)^{(5/(3\alpha)-1)\alpha k_d} = c^{(5/3-2\alpha)\cdot k_d}.$$

In order to allow the preprocessing on small instances, it is important when applying this technique that the search-tree algorithm that is used only creates subinstances that are vertex-induced subgraphs.

With $\alpha \approx 0.2724$, we can conclude:

Corollary 9 *By using exponential space,* NONBLOCKER *can be solved in time (and space)* $\mathcal{O}^*(2.5154^{k_d})$.

5 Special graph classes

5.1 Planar graphs

Since the rules that merge the catalyst with other vertices may destroy planarity, we may only claim the $2k_d$ kernel in the case of planar graphs.

We now use the following result on planar graphs by Fomin and Thilikos [8]:

Theorem 10 *Every planar n-vertex graph has treewidth at most* $9/\sqrt{8} \cdot \sqrt{n} \leq 3.182\sqrt{n}$.

Together with the treewidth-based algorithm for MINIMUM DOMINATING SET as developed in [1], we can conclude:

Corollary 11 *The* NONBLOCKER *problem, restricted to planar graphs, can be solved in time* $\mathcal{O}^*(2^{9\sqrt{k_d}})$.

5.2 Bipartite graphs

In this section we adapt our results of Section 4.2 and Section 4.3 to bipartite graphs using the following result of Fomin, Kratsch and Woeginger [7]:

Corollary 12 *The* MINIMUM DOMINATING SET *problem can be solved in time* $\mathcal{O}^*((\sqrt{3})^n) \leq \mathcal{O}^*(1.7321^n)$ on bipartite graphs.

We have to adapt the reduction rules, since the merging of vertices may also destroy bipartiteness.

Separate catalytic vertices are introduced for each bipartition and the reduction rules are modified accordingly. The only tricky detail is the design of an appropriate de-catalyzation rule, see Figure 1. There, the full blue and red vertices are the two catalytic vertices (one for each of the two bipartitions). The vertices in light blue and red colors (indicating their membership to the according bipartition) have to be added. In other words, the only vertices that have neighbors other than those indicated are (possibly) the two (former) catalytic vertices. The claim is that an instance containing the two catalytic vertices has a nonblocking-set of size k_d if and only if the graph that is obtained by replacing the two catalytic vertices by the gadget from Figure 1 (now forgetting about the catalytic annotations) has a nonblocking-set of size $k_d + 8$.

We can apply the result of Blank, McCuaig, and Shepherd [3, 11] to conclude:

Corollary 13 *By applying the algorithm of Fomin, Kratsch and Woeginger [7] to solve* MINIMUM DOMINATING SET *on bipartite graphs to a reduced instance* (G, k_d) *of* NONBLOCKER*, restricted to bipartite graphs, the* NONBLOCKER *problem can be*

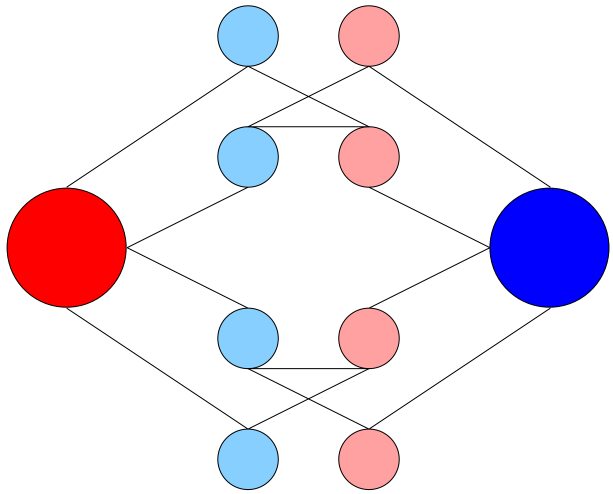

Figure 1: The schematics of the de-catalyzation rule.

solved in time $\mathcal{O}^*(2.4932^{k_d})$ *in polynomial space. Time can be further reduced to* $\mathcal{O}^*(2.2041^{k_d})$ *by also allowing for exponential space.*

6 Discussion: open questions

We have presented two efficient parameterized algorithms for the NONBLOCKER problem, the parameterized dual of DOMINATING SET. With the help of known (non-trivial) graph-theoretic results and new exact algorithms for MINIMUM DOMINATING SET, we were able to further reduce the involved constants.

It would be possible to use the result of Reed [15] to obtain a smaller kernel for NONBLOCKER if rules could be found to reduce vertices of degree two. Perhaps such rules may be possible for the bipartite case, or possibly there are better kernel sizes for the bipartite case.

Is it possible to find a better kernelization in the planar case? This would be interesting in view of recent lower bound results of J. Chen, H. Fernau, I. A. Kanj, and G. Xia that have shown there is no kernel smaller than $(67/66 - \epsilon)k_d$, see [5]. Such a result would immediately entail better running times for algorithms dealing with the planar case.

Finally, notice that our reduction rules get rid of all degree-two vertices that have another degree-two vertex as a neighbor. Is there an "intermediate" kernel size theorem (that somehow interpolates between the result of Blank, McCuaig and Shepherd and that of Reed)? Our use of the additional structural properties of the reduced graphs was to cope with the exceptional graphs from [11].

References

[1] J. Alber, H. L. Bodlaender, H. Fernau, T. Kloks, and R. Niedermeier. Fixed parameter algorithms for DOMINATING SET and related problems on planar graphs. *Algorithmica*, 33:461–493, 2002.

[2] G. Ausiello, P. Creczenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Protasi. *Complexity and Approximation; Combinatorial Optimization Problems and Their Approximability Properties*. Springer, 1999.

[3] M. Blank. An estimate of the external stability number of a graph without suspended vertices (in Russian). *Prikl. Math. i Programmirovanie Vyp.*, 10:3–11, 1973.

[4] L. Sunil Chandran and F. Grandoni. Refined memorization for vertex cover. In R. Downey, M. Fellows, and F. Dehne, editors, *International Workshop on Parameterized and Exact Computation IWPEC 2004*, volume 3162 of *LNCS*, pages 61–70. Springer, 2004.

[5] J. Chen, H. Fernau, I. A. Kanj, and Ge Xia. Parametric duality and kernelization: Lower bounds and upper bounds on kernel size. In V. Diekert and B. Durand, editors, *Symposium on Theoretical Aspects of Computer Science STACS 2005*, volume 3404 of *LNCS*, pages 269–280. Springer, 2005.

[6] R. G. Downey and M. R. Fellows. *Parameterized Complexity*. Springer, 1999.

[7] F. Fomin, D. Kratsch, and G. Woeginger. Exact (exponential) algorithms for the dominating set problem. In J. Hromkovic et al., editors, *30th International Workshop on Graph-Theoretic Concepts in Computer Science WG 2004*, volume 3353 of *LNCS*, pages 245–256. Springer, 2004.

[8] F. V. Fomin and D. Thilikos. A simple and fast approach for solving problems on planar graphs. In V. Diekert and M. Habib, editors, *Symposium on Theoretical Aspects of Computer Science STACS*, LNCS, pages 56–67. Springer, 2004.

[9] M. R. Garey and D. S. Johnson. *Computers and Intractability*. New York: Freeman, 1979.

[10] T. W. Haynes, S. T. Hedetniemi, and P. J. Slater. *Fundamentals of Domination in Graphs*, volume 208 of *Monographs and Textbooks in Pure and Applied Mathematics*. Marcel Dekker, 1998.

[11] B. McCuaig and B. Shepherd. Domination in graphs of minimum degree two. *Journal of Graph Theory*, 13:749–762, 1989.

[12] R. Niedermeier. *Invitation to Fixed-Parameter Algorithms*. Habilitationsschrift, Wilhelm-Schickard-Institut für Informatik, Universität Tübingen, Germany, October 2002.

[13] O. Ore. *Theory of Graphs*, volume XXXVIII of *Colloquium Publications*. American Mathematical Society, 1962.

[14] E. Prieto. *Systematic Kernelization in FPT Algorithm Design.* PhD thesis, The University of Newcastle, Australia, 2005.

[15] B. Reed. Paths, stars, and the number three. *Combinatorics, Probability and Computing*, 5:277–295, 1996.