Proc. 32nd SOFSEM-06, Vol. 3831 in LNCS, pp. 137-147, Springer, 2006

A General Data Reduction Scheme for Domination in Graphs*

Jochen Alber[1], Britta Dorn[2], and Rolf Niedermeier[3]

[1] DIgSILENT GmbH, Power System Applications & Consulting,
Heinrich-Hertz-Str. 9, D-72810 Gomaringen, Germany. j.alber@digsilent.de
[2] Mathematisches Institut, Universität Tübingen,
Auf der Morgenstelle 10, D-72076 Tübingen, Germany. brdo@fa.uni-tuebingen.de
[3] Institut für Informatik, Friedrich-Schiller-Universität Jena,
Ernst-Abbe-Platz 2, D-07743 Jena, Germany. niedermr@minet.uni-jena.de

Abstract. Data reduction by polynomial-time preprocessing is a core concept of (parameterized) complexity analysis in solving NP-hard problems. Its practical usefulness is confirmed by experimental work. Here, generalizing and extending previous work, we present a set of data reduction preprocessing rules on the way to compute optimal dominating sets in graphs. In this way, we arrive at the novel notion of "data reduction schemes." In addition, we obtain data reduction results for domination in directed graphs that allow to prove a linear-size problem kernel for Directed Dominating Set in planar graphs.

1 Introduction

Data reduction and kernelization rules are one of the primary outcomes of research on parameterized complexity: Attacking computationally hard problems, it always makes sense to simplify and reduce the input instance by efficient preprocessing. In this work, considering the graph problem Dominating Set, we introduce and study the notion of a data reduction *schemes.*

Our work is based on two lines of research both concerned with solving NP-hard problems. On the one hand, there is the concept of polynomial-time approximation algorithms and, in particular, the concept of polynomial-time approximation schemes (PTAS) where one gets a better approximation guarantee at the cost of higher running times (see [4] for details). On the other hand, there is the paradigm of local search (see [1] for details). In this paper, we combine ideas from both research areas. More specifically, based on Dominating Set, generalizing and extending previous work [3], we develop a whole scheme of data reduction rules. The central goal is to gain a stronger data reduction at the cost of increased preprocessing time (thus relating to the PTAS paradigm) through an approach that searches through "increasing neighborhoods" of graph vertices (thus relating to local search).

* Supported by the Deutsche Forschungsgemeinschaft (DFG), project PEAL (parameterized complexity and exact algorithms), NI 369/1, and Emmy Noether research group PIAF (fixed-parameter algorithms), NI 369/4.

The Dominating Set problem is: given a graph $G = (V, E)$ and a positive integer k, find a dominating set of size at most k, i.e., a set $V' \subseteq V$, $|V'| \leq k$, and every vertex in $V \setminus V'$ is adjacent to at least one vertex in V'. When dealing with the corresponding optimization problem, we will use $\text{ds}(G)$ to denote the size of an optimal dominating set in G.

The idea of data reduction is to efficiently "cut away easy parts" of the given problem instance and to produce a new and size-reduced instance where then exhaustive search methods etc. can be applied. In [3] it is shown that, for planar graphs, with two easy to implement polynomial-time data reduction rules one can transform an instance (G, k) of Dominating Set into a new instance (G', k') with $k' \leq k$ and the number of vertices of G' bounded by $O(k)$ such that (G, k) is a yes-instance iff (G', k') is a yes-instance. Thus, by means of these rules in polynomial time one can usually find several vertices that are part of an optimal dominating set, whilst reducing the size of the input graph considerably.

In this work, we provide a whole scheme of data reduction for minimum domination in graphs. We develop a general framework of data reduction rules from which the two data reduction rules given in [3] can be obtained as easy special cases. In fact, the more complex one of these two rules is even improved. Moreover, we demonstrate that this extension makes it possible to handle graphs that are not amenable to the previous rules. Exploring the joint neighborhood of ℓ vertices for fixed $\ell \geq 1$, our data reduction rules run in $n^{O(\ell)}$ worst-case-time.[1] Besides introducing and analyzing the concept of a general data reduction scheme for domination in undirected graphs, we additionally demonstrate how to transfer data reduction for undirected graphs to directed graphs. Despite its practical significance (e.g., in biological and social network analysis [2]), domination in directed graphs so far has been neglected in parameterized algorithmics. First, we show a direct translation of undirected into directed reduction rules. Second, we present new reduction rules that make it possible to prove a linear-size problem kernel for Dominating Set on directed planar graphs.

Due to the lack of space, some details and proofs had to be omitted. Significant parts of this work are based on [6].

2 Preliminaries and Previous Work

A *data reduction rule* for, e.g., Dominating Set replaces, in polynomial time, a given instance (G, k) by a "simpler" instance (G', k') such that (G, k) is a yes-instance iff (G', k') is a yes-instance. A parameterized problem (the parameter is k) is said to have a *problem kernel* if, after the application of the reduction rules, the resulting reduced instance has size $g(k)$ for a function g depending only on k. For instance, Dominating Set restricted to planar graphs has a problem kernel consisting of at most $335 \cdot k$ vertices [3], recently improved to the upper bound $67 \cdot k$ [5]. Extensions to graphs of bounded genus appear in [7].

[1] Based on our experiences [2] with implementing the two reduction rules from [3], we would expect to get faster running times in practice.

All our data reduction rules have in common that they explore local structures of a given graph. Depending on these structures the application of a reduction rule may have the following two effects:

1. Determine vertices that can be chosen for an optimal dominating set.
2. Reduce/shrink the graph by removing edges and vertices.

We revisit two polynomial-time reduction rules which were introduced in [3].

Neighborhood of a single vertex. Consider a vertex $v \in V$ of a given graph $G = (V, E)$. Partition the vertices of the *open neighborhood* $N(v) := \{ u \in V \mid \{u, v\} \in E \}$ of v into three different sets:

- the *exit vertices* $N_{\text{exit}}(v)$, through which we can "leave" the *closed neighborhood* $N[v] := N(v) \cup \{v\}$,
- the *guard vertices* $N_{\text{guard}}(v)$, which are neighbors of exit vertices, and
- the *prisoner vertices* $N_{\text{prison}}(v)$, which have no neighboring exit vertex:

$$\begin{aligned} N_{\text{exit}}(v) &:= \{ u \in N(v) \mid N(u) \setminus N[v] \neq \emptyset \}, \\ N_{\text{guard}}(v) &:= \{ u \in N(v) \setminus N_{\text{exit}}(v) \mid N(u) \cap N_{\text{exit}}(v) \neq \emptyset \}, \\ N_{\text{prison}}(v) &:= N(v) \setminus (N_{\text{exit}}(v) \cup N_{\text{guard}}(v)). \end{aligned}$$

A vertex in $N_{\text{prison}}(v)$ can only be dominated by vertices from $\{v\} \cup N_{\text{guard}}(v) \cup N_{\text{prison}}(v)$. Since v will dominate at least as many vertices as any other vertex from $N_{\text{guard}}(v) \cup N_{\text{prison}}(v)$, it is safe to place v into an optimal dominating set we seek for, which we simulate by adding a suitable *gadget* to G.

Old-1-Rule. *Consider a vertex v of the graph. If $N_{prison}(v) \neq \emptyset$ then choose v to belong to the dominating set: add a "gadget vertex" v' and an edge $\{v, v'\}$ to G and remove $N_{guard}(v)$ and $N_{prison}(v)$ from G.*[2]

Neighborhood of a pair of vertices. Similar to Old-1-Rule, explore the union of the joint neighborhood $N(v_1, v_2) := (N(v_1) \cup N(v_2)) \setminus \{v_1, v_2\}$ of two vertices $v_1, v_2 \in V$. Setting $N[v_1, v_2] := N[v_1] \cup N[v_2]$, define

$$\begin{aligned} N_{\text{exit}}(v_1, v_2) &:= \{ u \in N(v_1, v_2) \mid N(u) \setminus N[v_1, v_2] \neq \emptyset \}, \\ N_{\text{guard}}(v_1, v_2) &:= \{ u \in (N(v_1, v_2) \setminus N_{\text{exit}}(v_1, v_2)) \mid N(u) \cap N_{\text{exit}}(v_1, v_2) \neq \emptyset \}, \\ N_{\text{prison}}(v_1, v_2) &:= N(v_1, v_2) \setminus (N_{\text{exit}}(v_1, v_2) \cup N_{\text{guard}}(v_1, v_2)). \end{aligned}$$

Here, we try to detect an optimal domination of the vertices $N_{\text{prison}}(v_1, v_2)$ in our local structure $N(v_1, v_2)$. A vertex in $N_{\text{prison}}(v_1, v_2)$ can only be dominated by vertices from $\{v_1, v_2\} \cup N_{\text{guard}}(v_1, v_2) \cup N_{\text{prison}}(v_1, v_2)$. The following rule determines cases in which it is "safe" to choose one of the vertices v_1 or v_2 (or both) to belong to an optimal dominating set we seek for.

Old-2-Rule. *Consider a pair of vertices $v_1 \neq v_2 \in V$ with $|N_{prison}(v_1, v_2)| > 1$ and suppose that $N_{prison}(v_1, v_2)$ cannot be dominated by a single vertex from $N_{guard}(v_1, v_2) \cup N_{prison}(v_1, v_2)$.*

[2] Of course, in practical implementations (as in [2]) one would directly put v into the dominating set. Similar observations hold for the other data reduction rules.

Case 1 *If $N_{prison}(v_1, v_2)$ can be dominated by a single vertex from $\{v_1, v_2\}$:*

(1.1) *If $N_{prison}(v_1, v_2) \subseteq N(v_1)$ as well as $N_{prison}(v_1, v_2) \subseteq N(v_2)$, then*
- *as a gadget add two new vertices w_1, w_2 and edges $\{v_1, w_1\}$, $\{v_2, w_1\}$, $\{v_1, w_2\}$, $\{v_2, w_2\}$ to G and*
- *remove $N_{prison}(v_1, v_2)$ and $N_{guard}(v_1, v_2) \cap N(v_1) \cap N(v_2)$ from G.*

(1.2) *If $N_{prison}(v_1, v_2) \subseteq N(v_1)$, but not $N_{prison}(v_1, v_2) \subseteq N(v_2)$, then*
- *add a gadget vertex v_1' and an edge $\{v_1, v_1'\}$ to G and*
- *remove $N_{prison}(v_1, v_2)$ and $N_{guard}(v_1, v_2) \cap N(v_1)$ from G.*

(1.3) *If $N_{prison}(v_1, v_2) \subseteq N(v_2)$, but not $N_{prison}(v_1, v_2) \subseteq N(v_1)$, then choose v_2: proceed as in (1.2) with roles of v_1 and v_2 interchanged.*

Case 2 *If $N_{prison}(v_1, v_2)$ cannot be dominated by a single vertex from $\{v_1, v_2\}$,*
- *add two gadget vertices v_1', v_2' and edges $\{v_1, v_1'\}$, $\{v_2, v_2'\}$ to G and*
- *remove $N_{prison}(v_1, v_2)$ and $N_{guard}(v_1, v_2)$ from G.*

The practical usefulness of these two rules on real-world graphs (e.g., Internet graphs) has been demonstrated in [2].

3 A Data Reduction Scheme for Domination

In this section we establish the "mother rule" from which Old-1-Rule and Old-2-Rule can be derived as easy special cases. The idea is to explore the joint neighborhood of ℓ distinct vertices for a given constant ℓ. To cope with this more complex setting we will introduce a new *gadget* which generalizes the easy gadget vertices as they were used in the above two basic reduction rules.

A General Gadget. Our general reduction rule will—on the fly—generate a boolean "constraint formula" for an optimal dominating set D of the given graph: We identify the vertices V of a graph $G = (V, E)$ with 0/1-variables, where the meaning of a 1(0)-assignment is that the corresponding vertex will (not) belong to D. A boolean formula over the variables V then can be thought of as a constraint on the choice of vertices for an optimal dominating set.

Definition 1. *Let $\mathcal{W} \subseteq 2^V$ be a collection of subsets of V. The* constraint *associated with $\mathcal{W}$ is a boolean formula $F_{\mathcal{W}}$ in disjunctive normal form:*

$$F_{\mathcal{W}} := \bigvee_{W \in \mathcal{W}} \bigwedge_{w \in W} w.$$

A set $D \subseteq V$ fulfills *constraint $F_{\mathcal{W}}$ if the assignment where each vertex in D is set to 1 and each vertex in $V \setminus D$ is set to 0 satisfies $F_{\mathcal{W}}$.*

A constraint that was generated by a reduction rule will be encoded by a corresponding gadget in our graph which "implements" the formula as a subgraph. To keep the gadget as small as possible, it is desirable that the constraint itself is as compact as possible. We use the following notion of "compactification." A set system $\mathcal{W} \subseteq 2^V$ is said to be *compact* if no two elements in $\mathcal{W}$ are subsets of each other, i.e., if for all $W, W' \in \mathcal{W}$ we have: $W \subseteq W' \Rightarrow W = W'$.

Lemma 1. *Let $\mathcal{W} \subseteq 2^V$. There exists a minimal compact subset $\widehat{\mathcal{W}} \subseteq \mathcal{W}$ such that $F_{\mathcal{W}}$ is logically equivalent to $F_{\widehat{\mathcal{W}}}$ and $\widehat{\mathcal{W}}$ can be found in polynomial time.*

In the remainder, we call $\widehat{\mathcal{W}}$ the *compactification* of $\mathcal{W}$.

The above mentioned gadgets will be of the following form.

Definition 2. *Let $G = (V, E)$ and let $F_{\mathcal{W}}$ be a constraint associated with some set system $\mathcal{W} = \{W_1, \ldots, W_s\} \subseteq 2^V$ of $\ell := |\bigcup_{i=1}^{s} W_i|$ vertices. An $F_{\mathcal{W}}$-*gadget *is a set of $p := \prod_{i=1}^{s} |W_i|$ new* selector *vertices*

$$S := \{u_{(x_1,\ldots,x_s)} \mid x_i \in \{1, \ldots, |W_i|\}\}$$

and if $p < \ell$ another $(\ell - p)$ blocker *vertices B which are connected to G by the following additional edges: For each $1 \leq i \leq s$ with $W_i = \{w_{i1}, \ldots, w_{i|W_i|}\}$ and each $1 \leq j \leq |W_i|$, we add edges between w_{ij} and all selector vertices in $\{u_{(x_1,\ldots,x_s)} \in S \mid x_i = j\}$ and between w_{ij} and all blocker vertices in B. We denote the resulting graph by $G \oplus F_{\mathcal{W}}$.*

The idea is, firstly, that a set of vertices $V' \subseteq V$ fulfills the constraint $F_{\mathcal{W}}$ iff V' dominates all selector vertices in the $F_{\mathcal{W}}$-gadget. And, secondly, the blocker vertices are used to enforce that we can always find an optimal dominating set of $G \oplus F_{\mathcal{W}}$ without using any selector or blocker vertex at all. Encoding a constraint $F_{\mathcal{W}}$ by an $F_{\mathcal{W}}$-gadget, indeed, has the desired effect:

Proposition 1. *Let $G = (V, E)$ and let $F_{\mathcal{W}}$ be a constraint associated with some set system $\mathcal{W} \subseteq 2^V$. Then the size of an optimal dominating set of G which fulfills $F_{\mathcal{W}}$ is equal to the size of an optimal dominating set of $G \oplus F_{\mathcal{W}}$. Moreover, there exists an optimal dominating set of $G \oplus F_{\mathcal{W}}$ which contains only vertices in V, i.e., it contains no selector or blocker vertex.*

A Reduction Rule for the Joint Neighborhood of ℓ Vertices. In analogy to Old-1-Rule and Old-2-Rule, we explore the union of the neighborhoods of ℓ vertices. As a convention, we let, for $V' \subseteq V$, $N(V') := (\bigcup_{v \in V'} N(v)) \setminus V'$ and $N[V'] := \bigcup_{v \in V'} N[v]$. Consider a fixed set of ℓ vertices $V_\ell := \{v_1, \ldots, v_\ell\} \subseteq V$ and set

$$\begin{aligned} N_{\text{exit}}(V_\ell) &:= \{\, u \in N(V_\ell) \mid N(u) \setminus N[V_\ell] \neq \emptyset \,\}, \\ N_{\text{guard}}(V_\ell) &:= \{\, u \in N(V_\ell) \setminus N_{\text{exit}}(V_\ell) \mid N(u) \cap N_{\text{exit}}(V_\ell) \neq \emptyset \,\}, \\ N_{\text{prison}}(V_\ell) &:= N(V_\ell) \setminus (N_{\text{exit}}(V_\ell) \cup N_{\text{guard}}(V_\ell)). \end{aligned}$$

The left-hand side of Figure 1 shows an example for these three sets for $\ell = 3$.

Definition 3. *For two sets $\emptyset \neq W, W' \subseteq V$, we say that W is* better *than W' if $|W| \leq |W'|$ and $N[W] \supseteq N[W']$. If W is better than W', we write $W \leq W'$. If $W' = \emptyset$ and $W \neq \emptyset$, then always $W \leq W'$.*

Checking whether $W \leq W'$ can be done in $O((|W| + |W'|) \cdot n)$ time if we use the adjacency matrix of the given graph.

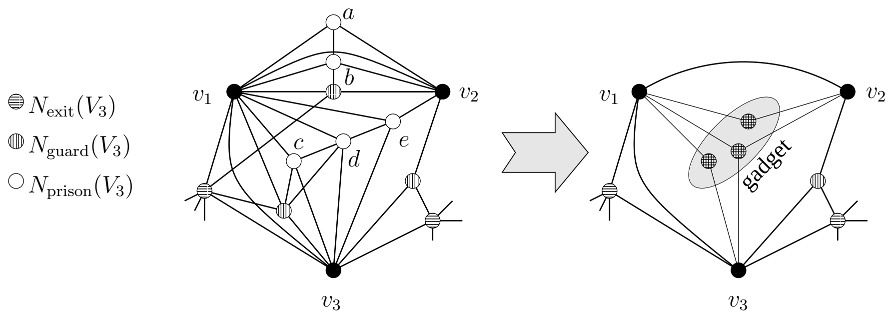

Fig. 1. Example for 3-Rule. The left-hand side shows the partitioning of $N(V_3)$ for $V_3 := \{v_1, v_2, v_3\}$ into the sets $N_{\text{exit}}(V_3)$, $N_{\text{guard}}(V_3)$, and $N_{\text{prison}}(V_3)$. The compactification of $\mathcal{W}$ (all subsets of vertices in V_3 that dominate $N_{\text{prison}}(V_\ell)$) is $\widehat{\mathcal{W}} = \{\{v_1\}, \{v_2, v_3\}\}$. The compactification of all alternative dominations of $N_{\text{prison}}(V_\ell)$ is $\widehat{\mathcal{W}}_{\text{altern}} = \{\{v_1\}, \{v_2, v_3\}, \{v_2, c\}, \{v_2, d\}, \{v_3, a\}, \{v_3, b\}, \{a, d\}, \{b, d\}\}$. Since, for each element in $\widehat{\mathcal{W}}_{\text{altern}}$, we find a better element in $\widehat{\mathcal{W}}$, 3-Rule applies. The compactified formula generated by 3-Rule is $F_{\widehat{\mathcal{W}}} = v_1 \vee (v_2 \wedge v_3)$. The right-hand side shows $N(V_3)$ after the application of 3-Rule. The $F_{\widehat{\mathcal{W}}}$-gadget is constructed according to Definition 2 using two selector vertices of degree 2 and one blocker vertex of degree 3.

Using this notation, for each $\ell \geq 1$, we obtain the following generalization of the first two reduction rules, yielding a whole scheme of reduction rules. The idea of the reduction scheme below is to deduce a constraint based on the question which vertices from a given set V_ℓ dominate $N_{\text{prison}}(V_\ell)$.

ℓ-Rule. *Consider ℓ pairwise distinct vertices $V_\ell := \{v_1, \ldots, v_\ell\} \subseteq V$ and suppose $N_{prison}(V_\ell) \neq \emptyset$.*

- *Compute the set*

$$\mathcal{W} := \big\{\, W \subseteq V_\ell \;\mid\; N_{prison}(V_\ell) \subseteq N[W] \,\big\}$$

of all vertex subsets of V_ℓ that dominate all prisoner vertices $N_{prison}(V_\ell)$, and the set of all alternatives to dominate $N_{prison}(V_\ell)$ with less than ℓ vertices:

$$\mathcal{W}_{altern} := \big\{\; W \subseteq N[N_{prison}(V_\ell)] \;\mid\; N_{prison}(V_\ell) \subseteq N[W] \text{ and } |W| < \ell \;\big\}.$$

- *Compute the compactifications $\widehat{\mathcal{W}}$ of $\mathcal{W}$ and $\widehat{\mathcal{W}}_{altern}$ of $\mathcal{W}_{altern}$.*
- *If ($\forall W \in \widehat{\mathcal{W}}_{altern} \exists W' \in \widehat{\mathcal{W}} : W' \leq W$), then*
  - *remove $\mathcal{R} := \big\{v \in N_{guard}(V_\ell) \cup N_{prison}(V_\ell) \mid N[v] \subseteq \bigcap_{W \in \widehat{\mathcal{W}}} N[W]\big\}$, and*
  - *put an $F_{\widehat{\mathcal{W}}}$-gadget to G for the constraint associated with $\widehat{\mathcal{W}}$.*

An example for $\ell = 3$ is given in Figure 1. If V_ℓ forms a size-ℓ dominating set, then ℓ-Rule actually solves the domination problem. Moreover, ℓ-Rule provides a mathematically more elegant formalism than for instance Old-2-Rule does. In addition, it generalizes Old-2-Rule:

Theorem 1. *For each ℓ, ℓ-Rule is correct, i.e., for every graph G, we have $\mathrm{ds}(G) = \mathrm{ds}(G')$, where G' denotes the graph obtained from G by applying the rule to ℓ distinct vertices. Moreover, 1-Rule is identical to Old-1-Rule and 2-Rule applies to even more cases than Old-2-Rule.*

Proof (Sketch). Observe that ℓ-Rule only applies if for all $W \in \widehat{\mathcal{W}}_{\mathrm{altern}}$ we find a $W' \in \widehat{\mathcal{W}}$ such that $W' \leq W$ $(*)$. Let $G'' := G \oplus F_{\widehat{\mathcal{W}}}$. We first of all argue that $\mathrm{ds}(G'') = \mathrm{ds}(G)$. It is clear that $\mathrm{ds}(G'') \geq \mathrm{ds}(G)$. Conversely, suppose that D is an optimal dominating set for G. We distinguish two cases. First, suppose that $N_{\mathrm{prison}}(V_\ell)$ needs less than ℓ vertices to be dominated. Then, by definition of $\mathcal{W}_{\mathrm{altern}}$, D has to fulfill $F_{\mathcal{W}_{\mathrm{altern}}}$. Hence, by the definition of compactification, we know that D also fulfills $F_{\widehat{\mathcal{W}}_{\mathrm{altern}}}$. In other words, this means that there has to be a $W \in \widehat{\mathcal{W}}_{\mathrm{altern}}$ with $W \subseteq D$. But then, by assumption $(*)$, we have a $W' \in \widehat{\mathcal{W}}$ with $W' \leq W$. Since W' is better than W, this implies that $D' := (D \setminus W) \cup W'$ is a dominating set for G which fulfills $F_{\widehat{\mathcal{W}}}$ and, hence, it is a dominating set for G'' (by Proposition 1) with $|D'| \leq |D|$. Second, suppose that $N_{\mathrm{prison}}(V_\ell)$ needs exactly ℓ vertices $D' \subseteq D$ to be dominated. Then, it is clear that $D'' := (D \setminus D') \cup V_\ell$ also forms a dominating set for G. But then, by construction, D'' dominates all $F_{\widehat{\mathcal{W}}}$-gadget vertices, and, thus, D'' is a dominating set for G'' with $|D''| = |D|$.

It remains to show $\mathrm{ds}(G'') = \mathrm{ds}(G')$. Observe that $G' = G'' \setminus \mathcal{R} = (G \oplus F_{\widehat{\mathcal{W}}}) \setminus \mathcal{R} = (G \setminus \mathcal{R}) \oplus F_{\widehat{\mathcal{W}}}$ with $\mathcal{R}$ as defined in ℓ-Rule. First of all, we show that $\mathrm{ds}(G'') \leq \mathrm{ds}(G')$. To see this, let D be an optimal dominating set for G'. Then, by Proposition 1, there exists a dominating set $D' \subseteq V(G)$ of equal size for $G \setminus \mathcal{R}$ which fulfills $F_{\widehat{\mathcal{W}}}$. This means that there exists a $W \in \widehat{\mathcal{W}}$ with $W \subseteq D'$. By definition of $\mathcal{R}$, this implies that $\mathcal{R} \subseteq N[\mathcal{R}] \subseteq N[\bigcap_{X \in \widehat{\mathcal{W}}} X] \subseteq N[W] \subseteq N_{G'=G''\setminus\mathcal{R}}[D'] \subseteq N_{G''}[D']$, which shows that D' is a dominating set for $G'' = G \oplus F_{\widehat{\mathcal{W}}}$ with $|D'| = |D|$. Similarly, one shows that $\mathrm{ds}(G') \leq \mathrm{ds}(G'')$.

It is not hard to see that 1-Rule is identical to Old-1-Rule and that 2-Rule applies whenever Old-2-Rule applies. In addition, there are examples where Old-2-Rule does not apply and where 2-Rule does apply. For instance, we can construct a graph where a single vertex v from $N_{\mathrm{prison}}(V_2)$ dominates $N_{\mathrm{prison}}(V_2)$. (i.e., Old-2-Rule does not apply) and 2-Rule still applies, since, e.g., $\{v\} \leq \{w\}$ where $w \in V_2$. □

The following proposition gives a simple worst-case estimate on the time needed to apply ℓ-Rule, and, together with the subsequent Theorem 2, shows that we have a relationship between "quality" of data reduction and running time as mentioned in the introductory section.

Proposition 2. *Let $G = (V, E)$. Applying ℓ-Rule for all size-ℓ vertex sets $V_\ell := \{v_1, \ldots, v_\ell\} \subseteq V$ takes $O(n^{2\ell})$ time for $\ell > 1$ and $O(n^3)$ time for $\ell = 1$.*

A graph $G = (V, E)$ is said to be *reduced with respect to ℓ-Rule* if there is no set of distinct vertices $v_1, \ldots, v_\ell$ for which ℓ-Rule can be applied. In a sense, the data reduction scheme given by ℓ-Rule builds a "strict hierarchy" of rules:

Theorem 2. *Let $\mathcal{H}_\ell := \{1\text{-Rule}, \ldots, \ell\text{-Rule}\}$, $\ell \geq 1$. Then, for each $\ell > 1$, $\mathcal{H}_\ell$ is strictly more powerful than $\mathcal{H}_{\ell-1}$.*

Proof (Sketch). For each level $\ell > 1$ of this hierarchy, we can construct a graph which is reduced with respect to all rules in $\mathcal{H}_{\ell-1}$ but which is still reducible with respect to ℓ-Rule. For example, let $G_\ell = \mathcal{P}_2 \times \mathcal{P}_{2\ell-1}$ be the complete grid graph of width 2 and length $2\ell - 1$. Then, it can be verified by induction on ℓ that G_ℓ has the above mentioned property. $\square$

4 Directed Dominating Set

In several applications we have to deal with directed graphs $\overrightarrow{G} = (V, A)$. Here, a vertex $v \in V$ is *dominated* iff it is in the dominating set or if there is an arc $(u, v) \in A$ (i.e., v is an outgoing neighbor of u) and u is in the dominating set.

Transforming Directed Graphs into Undirected Graphs. Let $\overrightarrow{G} = (V, A)$ be a directed graph. Construct an undirected graph $G' = (V', E)$, where $V' := \{u', u'' \mid u \in V\}$, and $E := \big\{\{u', u''\} \mid u \in V\big\} \cup \big\{\{u'', v'\}, \{u'', v''\} \mid (u, v) \in A\big\}$.

Proposition 3. *Using the notation above,* $\mathrm{ds}(\overrightarrow{G}) = \mathrm{ds}(G')$.

Clearly, in order to find an optimal dominating set for a directed graph $\overrightarrow{G}$, we can use the above transformation and then apply our undirected reduction rules (see Section 3) to the transformed instance G'. The drawback of this process is that the transformed graph G' contains twice as many vertices as $\overrightarrow{G}$. Moreover, the transformation in general does not preserve planarity. Hence, we subsequently modify the data reduction scheme for the undirected case to obtain a "directed data reduction scheme" for domination.

A Reduction Scheme for Directed Dominating Set. Let $\overrightarrow{G} = (V, A)$ be a directed graph. Define $N(v) := \{w \in V \mid (v, w) \in A\}$. For an ℓ-vertex set $V_\ell := \{v_1, \ldots, v_\ell\}$, explore $N(V_\ell) := \bigcup_{v \in V_\ell} N(v) \setminus V_\ell$.

Suppose we defined the partitioning $N_{\mathrm{exit}}(V_\ell)$, $N_{\mathrm{guard}}(V_\ell)$, and $N_{\mathrm{prison}}(V_\ell)$ and the reduction scheme in complete analogy to the undirected case, then we would run into the following problem: The vertices in $N_{\mathrm{prison}}(V_\ell)$ (if this set is non-empty) may also be dominated by vertices outside $N[V_\ell]$.[3] This difficulty is circumvented by slightly modifying the definition of the sets $N_{\mathrm{guard}}(V_\ell)$ and $N_{\mathrm{prison}}(V_\ell)$. More precisely, we additionally define the set

$$N_{\mathrm{enter}}(V_\ell) := \{\, u \in (N(V_\ell) \setminus N_{\mathrm{exit}}(V_\ell)) \mid \exists w \in (V \setminus N[V_\ell]) : (w, u) \in A\}.$$

[3] For example, there might be a single vertex v (with in-degree 0) which dominates $N(V_\ell)$, but which is not contained in $N(V_\ell)$. Then, clearly, it would be optimal to choose v. Deducing a constraint—based on the question which vertices from V_ℓ dominate $N_{\mathrm{prison}}(V_\ell)$ as done in the undirected case—would lead to a wrong result.

Herein, we used

$$N_{\text{exit}}(V_\ell) := \{\, u \in N(V_\ell) \mid \exists w \in (V \setminus N[V_\ell]) : (u,w) \in A\}.$$

The modified versions of $N_{\text{guard}}(V_\ell)$ and $N_{\text{prison}}(V_\ell)$ are defined as follows:

$$N_{\text{guard}}(V_\ell) := \{\, u \in (N(V_\ell) \setminus (N_{\text{exit}}(V_\ell) \cup N_{\text{enter}}(V_\ell))) \mid (N(u) \cap N_{\text{exit}}(V_\ell)) \neq \emptyset \,\},$$
$$N_{\text{prison}}(V_\ell) := N(V_\ell) \setminus (N_{\text{exit}}(V_\ell) \cup N_{\text{enter}}(V_\ell) \cup N_{\text{guard}}(V_\ell)).$$

In this way, we can build a data reduction scheme for Directed Dominating Set as a slight modification of ℓ-Rule in the undirected case.

Directed ℓ-Rule. *Consider ℓ pairwise distinct vertices $V_\ell := \{v_1, \ldots, v_\ell\} \subseteq V$ and suppose $N_{prison}(V_\ell) \neq \emptyset$. Compute the sets*

$$\mathcal{W} := \left\{\, W \subseteq V_\ell \;\mid\; N_{prison}(V_\ell) \subseteq N[W] \,\right\},$$
$$\mathcal{W}_{altern} := \left\{\, W \subseteq N[N_{prison}(V_\ell)] \;\mid\; N_{prison}(V_\ell) \subseteq N[W] \text{ and } |W| < \ell \,\right\},$$

and the compactifications $\widehat{\mathcal{W}}$ of $\mathcal{W}$ and $\widehat{\mathcal{W}}_{altern}$ of $\mathcal{W}_{altern}$.
If ($\forall W \in \widehat{\mathcal{W}}_{altern} \exists W' \in \widehat{\mathcal{W}} : W' \leq W$), then remove

- $\mathcal{R} := \left\{ v \in N_{enter}(V_\ell) \cup N_{guard}(V_\ell) \cup N_{prison}(V_\ell) \mid N[v] \subseteq \bigcap_{W \in \widehat{\mathcal{W}}} N[W] \right\}$ *and*
- *put an $F_{\widehat{\mathcal{W}}}$-gadget*[4]

Directed Dominating Set on Planar Graphs. Here, we provide a linear-size problem kernel for domination on directed planar graphs. To show this, we cannot make use of the transformation from directed to undirected graphs as described at the beginning of the section because the construction there does not preserve planarity. Hence, we use the Directed ℓ-Rules ($\ell = 1$ and $\ell = 2$ suffice and preserve planarity), yielding:

Theorem 3. Directed Dominating Set *on planar graphs has a linear-size problem kernel which can be found in $O(n^4)$ time. This implies that* Directed Dominating Set *on planar graphs is fixed-parameter tractable.*

We again omit the proof and just remark on the pitfalls behind: An ad-hoc idea to prove this result might be to take a reduced directed graph and to replace each arc by an undirected edge. If the directed graph was reduced, then one might hope that the corresponding undirected one is, too. But this is generally wrong. Moreover, even if the corresponding undirected graph was reduced, then still it is a problem that, as a rule, the undirected graph would have a usually smaller dominating set than the original directed one; however, there is no general relationship between these two set sizes. Hence one has to turn back to a direct analysis of the Directed ℓ-Rules. Fortunately, much of the proof work can be carried out by similar constructions as in the undirected case dealt with in [3].

[4] In contrast to the gadget with undirected edges as introduced in Definition 2, the newly introduced arcs now point *from* vertices in $W \in \widehat{\mathcal{W}}$ *to* selector vertices.

5 Outlook

We showed (Theorem 2) that the presented reduction rules form a strict hierarchy when considering larger and larger joint neighborhoods. It would be of hight interest to strengthen this result in the sense that one can mathematically relate the degree of increased reduction (e.g., by proving smaller problem kernel sizes) and the running time to be spent. Note that this would parallel relations that hold in the case of approximation schemes, and it would tie the notions of data reduction scheme and PTAS closer.

Presenting our reduction rules, for theoretical reasons we expressed boolean constraints as graph gadgets. From a practical point of view, in implementations it might make more sense not to use the graph gadgets (as has also been done when (successfully) experimentally testing the two reduction rules from [3] in [2]) but to use the boolean constraint formulas in a direct combination with the reduced graph instance. So far, this issue is completely unexplored.

From a parameterized complexity point of view, it would be interesting to gain further "tractability results" for W[1]-hard problems with respect to data reduction. More specifically, consider Dominating Set: Since Dominating Set is W[2]-complete, unless an unlikely collapse in parameterized complexity theory occurs, our data reduction scheme *cannot* serve for showing that using 1-Rule, 2-Rule, ..., c-Rule, for some constant c, generates a problem kernel in general graphs. A more realistic and nevertheless interesting kind of investigation would be to see what happens when c becomes dependent on the dominating set size k, e.g., $c = k/4$ or $c = \sqrt{k}$. If then the generated problem kernel consisted of $g(k)$ vertices, this would imply an algorithm with $O(2^{g(k)} + n^{2c})$ running time for Dominating Set, which might be considered as a significant (theoretical) improvement over the trivial exact algorithm running in $O(n^{k+2})$ time.

References

1. E. Aarts and J. K. Lenstra (eds). *Local Search in Combinatorial Optimization.* Wiley-Interscience Series in Discrete Mathematics and Optimization, 1997.
2. J. Alber, N. Betzler, and R. Niedermeier. Experiments on data reduction for optimal domination in networks. To appear, *Annals of Operations Research*, 2005.
3. J. Alber, M. R. Fellows, and R. Niedermeier. Polynomial time data reduction for Dominating Set. *Journal of the ACM*, 51(3):363–384, 2004.
4. G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Protasi. *Complexity and Approximation.* Springer-Verlag, 1999.
5. J. Chen, H. Fernau, I. A. Kanj, and G. Xia. Parametric duality and kernelization: Lower bounds and upper bounds on kernel size. In *Proc. 22d STACS*, volume 3404 of *LNCS*, pages 269–280. Springer, 2005.
6. B. Dorn. Extended data reduction rules for domination in graphs (in German). Student project, WSI für Informatik, Universität Tübingen, Germany, 2004.
7. F. V. Fomin and D. M. Thilikos. Fast parameterized algorithms for graphs on surfaces: linear kernel and exponential speed-up. In *Proc. 31st ICALP*, volume 3142 of *LNCS*, pages 581–592. Springer, 2004.