Electronic Colloquium on Computational Complexity, Report No. 27 (2004)

Parametric Duality: Kernel Sizes & Algorithmics

Henning Fernau

The University of Newcastle
School of Electrical Engineering and Computer Science
University Drive, NSW 2308 Callaghan, Australia
fernau@cs.newcastle.edu.au

Universität Tübingen
Wilhelm-Schickard-Institut für Informatik
Sand 13, D-72076 Tübingen, Germany
fernau@informatik.uni-tuebingen.de

February 20, 2004

Abstract

We derive certain properties for problems which are parameterized tractable both in their "primal" and in their "dual" parameterization. In particular, we derive the first ever lower bounds result for kernel sizes of parameterized problems. We discuss various consequences of this result. Moreover, we explain how to get improved non-parameterized algorithms from known parameterized algorithms by a "two-side attack."

ISSN 1433-8092

1 Introduction

Parameterized duality. It was observed that many problems which are parameterized tractable are becoming intractable when the parameter is "turned around," see [14, 21], although also the contrary has been found, cf. [4, 15]. For example, the MINIMUM VERTEX COVER problem is (as minimization problem) naturally parameterized by the entity to be minimized, in this case by an upperbound k on the number of vertices permitted in a feasible cover. If n denotes the number of vertices in the whole graph G, then it is well-known that (G, k) is a YES-instance of k-VERTEX COVER if and only if (G, k_d) is a YES-instance of k_d-INDEPENDENT SET, if we take $k_d = n - k$. In this sense, INDEPENDENT SET is the parametric dual problem to VERTEX COVER. While VERTEX COVER is parameterized tractable on general graphs, INDEPENDENT SET is parameterized intractable on general graphs. Similarly, while DOMINATING SET is parameterized intractable on general graphs, its parametric dual called NONBLOCKER is fixed-parameter tractable, instead.

The landscape changes when we turn our attention towards special graph classes, e.g., problems on *planar graphs* [4]. There, e.g., both INDEPENDENT SET and DOMINATING SET are parameterized tractable. In fact (and contrasting the "rule" stated above), there are quite a lot of problems for which both the problem itself and its dual are parameterized tractable.

The results of this paper. The beauty about problems which are in FPT together with their dual sibling problem is that this constellation allows, from an algorithmic standpoint, for a two-side attack on the original, non-parameterized problem. This means that, by playing a win-win game, we are able to arrive at new non-parameterized algorithms which may be superior to other published algorithms. From a complexity point of view, this two-side attack enables us to use classical complexity assumptions to prove that (most likely) certain problems have no tiny problem kernels. This first lower bound result on kernel sizes has a number of concrete consequences and opens up completely new lines of research.

2 Notions from parameterized complexity

A *parameterized problem* P is a usual decision problem together with a special entity called *parameter*. Formally, this means that the language of YES-instances of P, written $L(P)$, is a subset of $\Sigma^* \times \mathbb{N}$. An instance of a parameterized problem P is therefore a pair $(I, k) \in \Sigma^* \times \mathbb{N}$.

To formally specify what we mean by the dual of a parameterized problem, we explicitly need a proper notion of a size function. A mapping $s : \Sigma^* \times \mathbb{N} \to \mathbb{N}$ is called a *size function*

- if $0 \leq k \leq s(I, k)$,[1]
- if $s(I, k) \leq |I|$ (where $|I|$ denotes the length of the string I) and
- if $s(I, k) = s(I, s(I, k')$ for all appropriate k, k' *(independence).*

"Natural size functions" (in graphs, for example, these are entities as the "number of vertices" or the "number of edges") are independent. We can then also write $s(I)$.

For clarity, we denote a problem P together with its size function s as (P, s). To the *dual problem* P_d then corresponds the language of YES-instances $L(P_d) = \{(I, s(I) - k) \mid (I, k) \in L(P)\}$. The dual of the dual of a problem with size function is again the original problem due to the symmetry condition. Sometimes, we will call P the *primal problem* (distinguishing it from P_d). Then, k is the primal and k_d is the dual parameter.

Example 1 d-HITTING SET
Given: *A hypergraph* $G = (V, E)$ *with* edge degree *bounded by* d*, i.e.,* $\forall e \in E(|e| \leq d)$
Parameter: k
Question: *Is there a* hitting set *of size at most* k*, i.e.,*

$$\exists C \subseteq V, |C| \leq k, \forall e \in E(C \cap e \neq \emptyset)?$$

The special case $d = 2$ *is known as the* VERTEX COVER PROBLEM *in undirected graphs. Let* $L(d - HS)$ *denote the language of* d-HITTING SET. *Taking as size function* $s(G) = |V|$*, it is clear that the dual problem obeys* $(G, k_d) \in L(d - HS_d)$ *iff* G *has an* independent set *of cardinality* k_d*.*

[1]We deliberately ignore instances with $k > s(I, k)$ in this way, assuming that their solution is trivial. Moreover, it is hard to come up with a reasonable notion of "duality" if larger parameters are to be considered.

Example 2 DOMINATING SET
Given: *A (simple) graph* $G = (V, E)$
Parameter: k
Question: *Is there a* dominating set *of size at most* k*, i.e.,*

$$\exists D \subseteq V, |D| \leq k, \ \forall v \in V \setminus D \ \exists d \in D \ (\{d, v\} \in E)?$$

Taking as size function $s(G) = |V|$*, it is clear that the dual problem obeys* $(G, k_d) \in L(DS_d)$ *iff* G *has a* nonblocker set *of cardinality* k_d*.*

Example 3 FEEDBACK VERTEX SET
Given: *A (simple) graph* $G = (V, E)$
Parameter: k
Question: *Is there a* feedback vertex set *of size at most* k*, i.e.,*

$$\exists F \subseteq V, |F| \leq k, \forall c \in C(G)(F \cap c \neq \emptyset)?$$

Here, $C(G)$ *denotes the set of cycles of* G*, where a* cycle *is a sequence of vertices (also interpreted as a set of vertices)* $v_0, v_1, \dots, v_\ell$ *such that* $\{v_i, v_{(i+1) \bmod \ell}\} \in E$ *for* $i = 0, \dots, \ell - 1$*.*

This problem is (again) a vertex selection problem. Hence, we naturally take $s(G) = |V|$ *as the size function. Then, the dual problem can be described as follows.*

MAXIMUM VERTEX-INDUCED FOREST
Given: *A (simple) graph* $G = (V, E)$
Parameter: k_d
Question: *Is there a* vertex-induced forest *of size at least* k_d*, i.e.,*

$$\exists F \subseteq V, |F| \geq k_d, C(G[F]) = \emptyset?$$

Here, $G[F]$ *denotes the graph induced by* F *in* G*.*

In a similar fashion, one can define the problem FEEDBACK EDGE SET*, where we ask if it is possible to turn a graph into a forest by deleting at most* k *edges. Being an edge selection problem, the natural size function is now* $s(G) = |E|$*. The dual problem can be hence called* MAXIMUM EDGE-INDUCED FOREST*.*

Generally speaking, it is easy to “correctly” define the dual of a problem for selection problems as formalized in [5].

If P is a parameterized problem with $L(P) \subseteq \Sigma^* \times \mathbb{N}$ and $\{a, b\} \subseteq \Sigma$, then $L_c(P) = \{Iab^k \mid (I, k) \in L(P)\}$ is the *classical language associated to* P. So, we can also speak about $\mathcal{NP}$-hardness of a parameterized problem.

A parameterized problem P is called *fixed-parameter tractable* if there exists a solving algorithm for P running in time $\mathcal{O}(f(k)p(|I|))$ on instance (I, k) for some function f and some polynomial p.

As detailed in [13], d-HITTING SET and FEEDBACK VERTEX SET are known to be fixed-parameter tractable for general graphs.

A *kernelization* for a parameterized problem P with size function s is a polynomial-time computable reduction which maps an instance (I, k) onto (I', k') such that $s(I') \leq g(k)$ and $k' \leq k$, where g is an arbitrary function. I' is also called the *problem kernel* of I.

It is known (see [13]) that a parameterized problem is fixed-parameter tractable iff it has a kernelization. The complexity class characterized in this way is known as *FPT*. Similarly to the classical theory of $\mathcal{NP}$-hardness, there is also a "hardness theory" for parameterized problems, the most important notion being that of $W[1]$-hardness. For example, DOMINATING SET and INDEPENDENT SET are known to be $W[1]$-hard and it is therefore thought to be unlikely that these problems lie in FPT.

Of special interest in the following will be *linear kernels* where $g(k) = \alpha k$ for some α in the kernelization. Such small kernels are known, in particular, for graph problems restricted to planar graphs.

3 Small kernels

In a certain sense, kernels are the essential ingredient of parameterized algorithmics, since a problem is fixed-parameter tractable iff it admits a problem kernel. The smaller the problem kernel, the "more tractable" is the corresponding problem. As we will see in this section, we cannot hope for arbitrarily small kernels for $\mathcal{NP}$-hard problems (unless $\mathcal{P} = \mathcal{NP}$), especially if both primal and dual problem are fixed-parameter tractable.

Lemma 1 *If (P, s) is a parameterized problem with size function and if P admits a kernelization r such that $s(r(I, k)) \leq \alpha k$ for some $\alpha < 1$, then P is in $\mathcal{P}$.*

Proof. $s(I', k') \geq k'$ for each instance (I', k') according to our definition of a size function. This is in particular true for the parameter k' of the problem kernel instance $I' = r(I, k)$. So, $k' \leq \alpha k$ for some $\alpha < 1$. Repeatedly kernelizing, we arrive at a problem with arbitrary small parameter and hence of arbitrarily small size. Basically, we need $\mathcal{O}(\log k)$ many such kernelizations, each of them requiring polynomial time. Hence, the classical language $L_c(P)$ can be decided in polynomial time. Q.E.D.

We now discuss the case when both the primal and the dual version of a problem admits a linear kernel.

Theorem 3.1 *Let P be an $\mathcal{NP}$-hard parameterized problem with size function. If P admits an αk-size kernel and its dual P_d admits an $\alpha_d k_d$-size kernel ($\alpha, \alpha_d \geq 1$), then*

$$(\alpha - 1)(\alpha_d - 1) \geq 1$$

unless $\mathcal{P}$ equals $\mathcal{NP}$.

Proof. Let $r(\cdot)$ denote the assumed linear kernelization reduction for P. Similarly, $r_d(\cdot)$ is the linear kernelization for P_d. Consider the following program for a reduction R, given an instance (I, k) of P:

`if` $k \leq \frac{\alpha_d}{\alpha+\alpha_d} s(I)$ `then compute` $r(I, k)$
`otherwise, compute` $r_d(I, s(I) - k)$

For the size of the R-reduced instance I', we can compute:

- If $k \leq \frac{\alpha_d}{\alpha+\alpha_d} s(I)$, then $s(I') \leq \alpha k \leq \frac{\alpha\alpha_d}{\alpha+\alpha_d} s(I)$.
- Otherwise,

$$\begin{array}{rcl} s(I') & \leq & \alpha_d k_d \\ & = & \alpha_d (s(I) - k) \\ & < & \alpha_d \left(s(I) - \dfrac{\alpha_d}{\alpha + \alpha_d} s(I) \right) \\ & = & \dfrac{\alpha\alpha_d}{\alpha + \alpha_d} s(I) \end{array}$$

By repeatedly applying R, the problem P is solvable in polynomial time, if $\frac{\alpha\alpha_d}{\alpha+\alpha_d} < 1$. Q.E.D.

From the previous theorem, we can immediately deduce a couple of

Corollaries: Assuming $\mathcal{P}$ is not equal to $\mathcal{NP}$, we can conclude:

1. For any $\epsilon > 0$, there is no $(4/3 - \epsilon)k$ kernel for PLANAR k-VERTEX COVER.

   Proof. Recall that PLANAR k_d-INDEPENDENT SET has a $4k_d$ kernel due to the four-color theorem. Q.E.D.

   This "negative result" immediately transfers to more general graph classes in the following manner:

2. If there is any way to produce a kernel smaller than $4/3k$ for k-VERTEX COVER on general graphs, then the corresponding reduction rules must "somehow" possibly introduce $K_{3,3}$ or K_5 as subgraphs (or as minors) into the reduced instance.

   Proof. Assume that there were a kernelization algorithm which does not introduce $K_{3,3}$ or K_5 as subgraphs (or as minors) into the reduced instances. Then, this would also be a kernelization algorithm for PLANAR k-VERTEX COVER, since it would be planarity preserving due to Kuratowski's theorem. Therefore, Cor. 1 applies. Q.E.D.

3. Conversely, for any $\epsilon > 0$, there is no $(2 - \epsilon)k_d$ kernel for PLANAR k_d-INDEPENDENT SET. Likewise, there is no $(2 - \epsilon)k_d$ kernel for k_d-INDEPENDENT SET ON GRAPHS OF MAXIMUM DEGREE BOUNDED BY THREE. This is even true for the combination problem (which is still $\mathcal{NP}$-hard): There is no $(2 - \epsilon)k_d$ kernel for k_d-INDEPENDENT SET ON PLANAR GRAPHS OF MAXIMUM DEGREE BOUNDED BY THREE.

   Proof. The general k-VERTEX COVER has a $2k$ kernel based on a Theorem due to Nemhauser and Trotter [9]. For our purposes, it is enough to know that that rule identifies a subset of vertices V', $|V'| \leq 2k$ of the given graph instance $G = (V, E)$ and a parameter $k' \leq k$ such that G has a k-vertex cover iff the induced subgraph in $G[V']$ has a k'-vertex cover. Since the class of planar graphs, as well as the class of graphs of a specified bounded degree, are closed under taking induced subgraphs, the claims are true by Theorem 3.1. Q.E.D.

4. Based on a theorem due to Grötzsch (which can be turned into a polynomial-time coloring algorithm; see [8, 20]) it is known that planar triangle-free graphs are 3-colorable. This implies a $3k_d$ kernel for k_d-INDEPENDENT SET restricted to this graph class. Hence, a $(1.5 - \epsilon)k$ lower bound for a possible kernel for k-VERTEX COVER

RESTRICTED TO TRIANGLE-FREE PLANAR GRAPHS. Observe that the Nemhauser/Trotter kernelization preserves planarity and triangle-freeness. This is interesting due to the following result:

Lemma 2 *k-VERTEX COVER RESTRICTED TO TRIANGLE-FREE PLANAR GRAPHS is $\mathcal{NP}$-hard.*

Proof. (Sketch) In the standard construction [17] reducing (PLANAR) 3SAT to (PLANAR) VERTEX COVER, we replace each triangle clause gadget by a gadget which is a circle of length nine, where each third vertex is representing a literal and is hence connected to the corresponding variable gadget as before. The "non-literal" vertices are not connected to any vertices outside of the gadget. Such a gadget itself necessarily brings five vertices into the cover. This is the core observation for stating that, given a PLANAR 3SAT instance I with m clauses, I is satisfiable if the reduced VERTEX COVER instance I' has a cover of size $8m$. Q.E.D.

5. Since "Euler-type" theorems exist for graphs of arbitrary genus g, it can be shown that there is a constant c_g such that each graph of genus g is c_g-colorable. Hence, according lower bounds for kernel sizes of k-VERTEX COVER ON GRAPHS OF GENUS g can be derived. For triangle-free graphs of genus g, Thomassen has shown that the corresponding constant c'_g is in $\mathcal{O}(g^{1/3}(\log g)^{-2/3})$, see [18, 26].

6. There is no $(336/335 - \epsilon)k_d$ kernel for PLANAR k_d-NONBLOCKER for any choice of $\epsilon > 0$.

   Proof. A $335k$ kernel for k-DOMINATING SET ON PLANAR GRAPHS was recently derived [3]. Hence, the lower bound follows. Q.E.D.

7. For any $\epsilon > 0$, there is no $(2 - \epsilon)k$ kernel for k-DOMINATING SET ON PLANAR GRAPHS. This is also true when further restricting the graph class to planar graphs of maximum degree three (due to the known $\mathcal{NP}$-hardness of that problem).

   Proof. C. McCartin has derived a $2k_d$ kernel for k_d-NONBLOCKER on general graphs which does not introduce any possible violations of planarity or degree bounds.[2] Q.E.D.

[2] In ongoing work together with F. Dehne, M. Fellows, E. Prieto-Rodriguez, F. Rosa-

This opens a completely new line of research:

- Can we find examples of problems such that the derived kernel sizes are optimal (unless $\mathcal{P}$ equals $\mathcal{NP}$)?
- If not, can we close the gaps more and more? According to our previous discussion, PLANAR VERTEX COVER ON TRIANGLE-FREE GRAPHS is our "best match:" we know how to derive a kernel of size $2k$ (due to Nemhauser & Trotter), and (assuming $\mathcal{P} \neq \mathcal{NP}$) we know that no kernel smaller than $1.5k$ is possible.
- Are there other, possibly more sophisticated arguments for showing lower bounds on kernel sizes? Especially, it would be interesting to have arguments ruling out say the existence of a kernel of size $o(k^3)$ in a situation when a kernel of size $\mathcal{O}(k^3)$ has been obtained. The kind of algebra we used in the proof of Theorem 3.1 does not extend.
- Although we are only able to derive results for problems where both the primal and the dual parameterization allow for linear size kernels, this might already give a good starting point, especially for graph problems. Observe that many $\mathcal{NP}$-hard graph problems are still $\mathcal{NP}$-hard when restricted to the class of planar graphs. However, in the planar case, our general impression is that linear bounds can be obtained due to the known linear relationships amongst the numbers of edges, faces and vertices.

4 An algorithmic attack from two sides

It is natural that algorithms developped in the parameterized framework can also be used to solve the "non-parameterized" versions of the problem, in many cases simply by possibly testing all parameter values. As shown in the case of solving the INDEPENDENT SET PROBLEM ON GRAPHS OF MAXIMUM DEGREE THREE, sometimes upper bounds on the possible parameter values are known. In the mentioned example, the size of a mimum vertex cover is upperbounded by $2/3n$ for connected graphs, where n here and in the following

mond and U. Stege we have derived a $7/4k_d$ kernel for k_d-NONBLOCKER on general graphs. Unfortunately, the corresponding reduction rules do possibly increase the maximum degree in the graph and do possibly destroy planarity. To our knowledge, McCartin's result never got published.

is the number of vertices of the graph instance. Chen, Kanj and Xia [10] used this result to turn their $\mathcal{O}(1.194^k + kn)$ algorithm for k-VERTEX COVER ON GRAPHS OF MAXIMUM DEGREE THREE into an $\mathcal{O}(1.194^{2n/3}) = \mathcal{O}(1.1254^n)$ algorithm for INDEPENDENT SET PROBLEM ON GRAPHS OF MAXIMUM DEGREE THREE. So, knowing bounds on the possible parameter values helps considerably reduce the bounds on the computing time. In a similar spirit, the four-color theorem teaches us that each n-vertex planar graph has a minimum vertex cover of size at most $3/4n$. The known $\mathcal{O}(1.29^k + kn)$ algorithm for k-VERTEX COVER this way implies an $\mathcal{O}(1.285^{3n/4}) = \mathcal{O}(1.207^n)$ algorithm for PLANAR INDEPENDENT SET, which is slightly better than Robson's algorithm [25] (for general graphs) needing $\mathcal{O}(1.211^n)$ time.

With problems having both FPT algorithms for their primal and for their dual parameterizations, we have the possibility of converting both algorithms into one non-parameterized algorithm, kind of attacking the problem from two sides. This means that we can use either of the two FPT algorithms.

Theorem 4.1 *Let (P, s) be a parameterized problem with size function and P_d its dual. Assume that both P and P_d are in FPT. Let f be some monotone function. Assume that there is an algorithm A for solving P on instance (I, k), having running time $\mathcal{O}(f(\beta k)p(s(I)))$ for some polynomial p, and that A_d is an algorithm for solving P_d on instance (I, k_d) running in time $\mathcal{O}(f(\beta_d k_d)p_d(s(I)))$ for a polynomial p_d.*

Then, there is an algorithm A' for solving the non-parameterized problem instance I running in time

$$\mathcal{O}(f(\frac{\beta\beta_d}{\beta + \beta_d}s(I))p'(s(I)))$$

for some polynomial p'.

Proof. Algorithm A' will use A as long as it is better than using A_d. This means we have to compare

$$f(\beta k) \quad \text{versus} \quad f(\beta_d(s(I) - k_d))$$

Since f is monotone, this means we simply have to compare

$$\beta k \quad \text{versus} \quad \beta_d(s(I) - k_d)$$

Some algebra shows that the following algorithm A' is then "best" for the de-parameterized problem P, given an instance I:

```
For all parameter values k do:
   if k ≤ β_d/(β+β_d) s(I) then compute A(I,k)
   otherwise, compute A_d(I, s(I) − k)
output the 'best' of all computed solutions
```

Considering the boundary case $k = \frac{\beta_d}{\beta+\beta_d}s(I)$ gives the claimed worst case running time. Here, $p'(j) = j(p(j) + p_d(j))$. Q.E.D.

Let us explain this theorem by some *example computations*.

1. By taking the $\mathcal{O}(1.194^k+n)$ algorithm for k-VERTEX COVER ON GRAPHS OF MAXIMUM DEGREE THREE and the (trivial) $\mathcal{O}(4^{k_d}n)$ for the dual k_d-INDEPENDENT SET PROBLEM ON GRAPHS OF MAXIMUM DEGREE THREE, we obtain an $\mathcal{O}(1.171^n)$ algorithm for MAXIMUM INDEPENDENT SET ON GRAPHS OF MAXIMUM DEGREE THREE. This algorithm is worse than the one obtained by Chen, Kanj and Xia (see above). Why? The case distinction within the combined algorithm is at $k \leq 0.8866n$, while we *know* that always $k \leq 0.666n$. Hence, the parameterized independent set algorithm will be never employed.

2. We can play the same game for MAXIMUM INDEPENDENT SET ON PLANAR GRAPHS.

   Combining the $\mathcal{O}(6^{k_d} + p(n))$-algorithm for k_d-INDEPENDENT SET ON PLANAR GRAPHS and the known $\mathcal{O}(1.285^k+kn)$ algorithm for VERTEX COVER (on general graphs) [9], we get an $\mathcal{O}(1.246^n)$ algorithm, clearly worse than Robson's.

   By using some results of Borodin *et al.*, see [1], we can show:

   Theorem 4.2 k_d-PLANAR INDEPENDENT SET *can be solved in time* $\mathcal{O}(5.1623^{k_d} + p(n))$.

   This then yields an $\mathcal{O}(1.243^n)$ algorithm, still worse than Robson's.

   Alternatively, we can start with the parameterized algorithms of "type" $\mathcal{O}(c^{\sqrt{k}} + n)$ which are known for both problems. This means (in the setting of the theorem) that we let $f(x)$ to be $2^{\sqrt{x}}$.

   Plugging in the best-known constants, i.e.,

   - $\beta = 4.5^2 = 20.25$ in the case of k-VERTEX COVER ON PLANAR GRAPHS [16] and

- $\beta_d = 48$ in the case of k_d-INDEPENDENT SET ON PLANAR GRAPHS (long version of [4]),

we get an

$$\mathcal{O}(2^{3.773\sqrt{n}}) = \mathcal{O}(13.68^{\sqrt{n}})$$

algorithm for MAXIMUM INDEPENDENT SET ON PLANAR GRAPHS. Using that a minimum vertex cover in planar graphs has at most $3/4n$ vertices this time gives us a worse result, namely an algorithm running in time $\mathcal{O}(2^{\sqrt{20.25*3/4*n}}) = \mathcal{O}(2^{3.898\sqrt{n}}) = \mathcal{O}(14.90^{\sqrt{n}})$.

More precisely, taking (in the spirit of klam values as described in [13]) a value of 10^{20} "operations" as "benchmark" for how far each type of algorithm might take us, we see that with Robson's algorithm graphs with about 250 vertices are still manageable, while our new algorithm can cope with graphs with over more than 300 vertices.

By a completely different approach, namely by bounding the tree-width of any planar graph $G = (V, E)$ by $3.182\sqrt{|V|}$, Fomin and Thilikos were recently able to obtain an even better algorithm, running in time $\mathcal{O}(9.08^{\sqrt{n}})$. This means that actually planar graphs with up to 500 vertices are manageable.

3. We now consider FEEDBACK VERTEX SET. Since on general graphs the parameterized dual is hard [21], we again consider the problem *restricted to planar graphs.*

   Based on a coloring theorem of Borodin [7] and on the reasoning given by Goemans and Williamson [19], in parts explicitly formulated in terms of parameterized complexity in [23], the following two lemmas can be shown:

   Lemma 3 k_d-MAXIMUM VERTEX-INDUCED FOREST ON PLANAR GRAPHS *has a* $2.5k_d$ *kernel and can hence be solved in time*

$$\mathcal{O}\left(\left(2.5\left(\frac{2.5}{1.5}\right)^{1.5}\right)^{k_d} + p(n)\right) = \mathcal{O}(5.3792^{k_d} + p(n)).$$

   Lemma 4 k-FEEDBACK VERTEX SET ON PLANAR GRAPHS *can be solved in time* $\mathcal{O}(5^k n)$.

Taken together, this gives an algorithm for MAXIMUM VERTEX-INDUCED FOREST ON PLANAR GRAPHS running in time $\mathcal{O}(2.5914^n)$. Taking the 10^{20}-benchmark, this means that planar graphs with up to 50 vertices can be treated this way.

4. Let us finally consider PLANAR MINIMUM DOMINATING SET. We will only consider the c^k-type algorithms.

   - For k-DOMINATING SET ON PLANAR GRAPHS, an $\mathcal{O}(8^k + kn)$ algorithm is known [2].
   - For k_d-NONBLOCKER (on general graphs), an $\mathcal{O}(3.31^k + n)$ algorithm is has been recently developed, see footnote 2.

   Combining both algorithms leads to an $\mathcal{O}(5.67^n)$ algorithm for PLANAR MINIMUM DOMINATING SET. By an alternative approach, an $\mathcal{O}(32.97^{\sqrt{n}})$ algorithm has been exhibited [16].

Similar results can be also obtained for other graph families, as they are described in [11]. As a non-graph-theoretic example, let us mention the "tardy task problem" from [15]. Keeping the second "parameter" m described in that paper, our method provides an $\mathcal{O}(m^{mn/(m+1)})$ algorithm for this problem, where n is the number of tasks.

5 Conclusions

We had a closer look at parameterized problems P of the form that both P (the "parameterized primal") and the parameterized dual problem P_d are in FPT, where the dualization was based upon the availability of a "nice" size function.

These problems seem to form an interesting subclass of FPT (maybe to be called PD-FPT [primal-dual FPT]), since they show nice relations to the non-parameterized optimization problem, both regarding hardness results and new algorithms.

A natural question is especially if there are other subclasses of FPT problems for which lower bounds on the size of the kernels could be shown. If not, then PD-FPT surely deserves to be studied on its own account.

There is an alternative interpretation of our hardness results: namely, whenever somebody is insisting on getting a really small kernel for PLANAR VERTEX COVER, say $1.1k$, then this is only possible by a *parameterized* *tractable* kernelization reduction, i.e., a kernelization computing (I', k') from (I, k) running in time $\mathcal{O}(f(k)p(s(I)))$ for some polynomial p and an arbitrary function f, because the proof of the hardness result relies on the definition of a kernelization reduction function being computable in polynomial time. Of course, this would perfectly fit into the whole picture of parameterized algorithmics. So, in this interpretation, our hardness results mean that parameterized tractable kernelizations may be sometimes unavoidable.

Finally, possible connections to approximability come to one's mind. Let us explain this (again) by means of the example VERTEX COVER. Firstly, it appears to be that any progress that we make into designing kernelizations which give smaller kernels has its bearing into developing better approximation algorithms. Namely, we just take all vertices "left over" after kernelization into the vertex cover. In actual fact, Hochbaum's interpretation of Nemhauser and Trotter's kernelization theorem (this is how it could be read, although this interpretation is anachronistic) for VERTEX COVER is doing just this. Due to the lower bounds we derived in this paper, this approach to derive approximation algorithms cannot pursued "forever". In a certain sense, we already knew this, since MINIMUM VERTEX COVER is APX complete. But the kernelization lower bounds we got (1.3333) are (just by comparing the raw numbers) only slightly worse than the best known lower bounds on the (non-)approximability of VERTEX COVER (current record: 1.3607 [12]; actually, there is a link with a certain conjecture in "PCP games" established in [22]). Along this reasoning, the $7/6+\epsilon$ approximation algorithm for VERTEX COVER ON GRAPHS WITH MAXIMUM DEGREE THREE due to Berman and Fujito[3] cannot be matched by an approximation algorithm which is derived from a kernelization algorithm.

Are there any further (formalizable) connections between APX and PTAS on the one hand and FPT and PD-FPT (both with linear kernels)? Note that recently many interesting relationships between parameterized tractability and approximability had been established, see [13], so that (in principle) these connections would not come as such a surprise.

It is also possible to define the notion of a *kernelization scheme*, requiring that for each $\alpha > 1$ we find a polynomial time kernelization function r (whose running time can depend on the chosen α) such that $s(r(I, k)) \leq \alpha k$. Theorem 3.1 shows that whenever a problem (P, s) happens to have a kernelization scheme, then its parameterized dual P_d cannot have a linear kernel (unless

[3]All the approximability results are taken from [6].

$\mathcal{P}$ equals $\mathcal{NP}$). Hence, is is quite unlikely to find a kernelization scheme for VERTEX COVER ON PLANAR GRAPHS, while approximation schemes for this problem are well-known.

Acknowledgment: We are very grateful for remarks of Gerhard Woeginger on a draft version of this paper and for further hints due to Oleg Borodin and Klaus-Jörn Lange.

References

[1] V. A. Aksionov, O. V. Borodin, L. S. Mel'nikov, G. Sabidussi, M. Stiebitz, and B. Toft. Deeply asymmetric planar graphs. Technical Report PP–2000–14, University of Southern Denmark, IMADA, Odense, Denmark, September 2000.

[2] J. Alber, H. Fan, M. R. Fellows, H. Fernau, R. Niedermeier, F. Rosamond, and U. Stege. Refined search tree techniques for the PLANAR DOMINATING SET problem. In J. Sgall, A. Pultr, and P. Kolman, editors, *Mathematical Foundations of Computer Science (MFCS 2001)*, volume 2136 of *LNCS*, pages 111–122. Springer, 2001. Long version to appear in Journal of Computer and System Sciences.

[3] J. Alber, M. R. Fellows, and R. Niedermeier. Efficient data reduction for DOMINATING SET: a linear problem kernel for the planar case. In M. Penttonen and E. Meineche Schmidt, editors, *SWAT 2002*, volume 2368 of *LNCS*, pages 150–159. Springer, 2002.

[4] J. Alber, H. Fernau, and R. Niedermeier. Parameterized complexity: exponential speedup for planar graph problems. In F. Orejas, P. G. Spirakis, and J. v. Leeuwen, editors, *International Colloquium on Automata, Languages and Programming ICALP'01*, volume 2076 of *LNCS*, pages 261–272. Springer, 2001. Long version to appear in Journal of Algorithms.

[5] J. Alber, H. Fernau, and R. Niedermeier. Graph separators: a parameterized view. *Journal of Computer and System Sciences*, 67:808–832, 2003.

[6] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, M. Marchetti-Spaccamela, and M. Protasi. *Complexity and Approximation.* Springer, 1999.

[7] O. Borodin. On acyclic colorings of planar graphs. *Discrete Mathematics*, 25:211–236, 1979.

[8] N. de Castro, F. J. Cobos, J. C. Dana, A. Márquez, and M. Noy. Triangle-free planar graphs as segment intersection graphs. *Journal of Graph Algorithms and Applications*, 6:7–26, 2002.

[9] J. Chen, I. A. Kanj, and W. Jia. Vertex cover: further observations and further improvements. *Journal of Algorithms*, 41:280–301, 2001.

[10] J. Chen, I. A. Kanj, and G. Xia. Labeled search trees and amortized analysis: improved upper bounds for NP-hard problems. In T. Ibaraki, N. Katoh, and H. Ono, editors, *Proc. 14th Annual International Symposium on Algorithms and Computation*, volume 2906 of *LNCS*, pages 148–157, 2003.

[11] E. D. Demaine, F. V. Fomin, M. Hajiaghayi, and D. M. Thilikos. Subexponential parameterized algorithms on graphs of bounded genus and H-minor-free graphs. Technical Report 251, Department of Informatics, University of Bergen (Norway), 2003. To appear in Proc. SODA 2004.

[12] I. Dinur and S. Safra. The importance of being biased. Technical Report 104, Electronic Colloquium on Computational Complexity (ECCC), 2001. Extended abstract at STOC 2002.

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

[14] M. Fellows. Parameterized complexity: the main ideas and connections to practical computing. *Electronic Notes in Theoretical Computer Science*, 61, 2002.

[15] M. R. Fellows and C. McCartin. On the parametric complexity of schedules to minimize tardy tasks. *Theoretical Computer Science*, 298:317–324, 2003.

[16] F. V. Fomin and D. M. Thilikos. A simple and fast approach for solving problems on planar graphs. Technical Report 258, Department of Informatics, University of Bergen (Norway), 2003. To appear in Proc. STACS 2004.

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

[18] J. Gimbel and C. Thomassen. Coloring triangle-free graphs with fixed size. *Discrete Mathematics*, 219:275–277, 2000.

[19] M. X. Goemans and D. P. Williamson. Primal-dual approximation algorithms for feedback problems in planar graphs. *Combinatorica*, 18(1):37–59, 1998.

[20] H. Grötzsch. Ein Dreifarbensatz für dreikreisfreie Netze auf der Kugel. *Wiss. Zeitschrift der Martin-Luther-Univ. Halle-Wittenberg, Math.-Naturwiss. Reihe*, 8:109–120, 1959.

[21] S. Khot and V. Raman. Parameterized complexity of finding subgraphs with hereditary properties. *Theoretical Computer Science*, 289:997–1008, 2002.

[22] S. Khot and O. Regev. Vertex cover might be hard to approximate to within $2-\varepsilon$. In *Proc. of 18th IEEE Annual Conference on Computational Complexity (CCC)*, pages 379–386, 2003.

[23] T. Kloks, C. M. Lee, and J. Liu. New algorithms for k-face cover, k-feedback vertex set, and k-disjoint cycles on plane and planar graphs. In L. Kučera, editor, *Graph-Theoretic Concepts in Computer Science WG 2002*, volume 2537 of *LNCS*, pages 282–295. Springer, 2002.

[24] D. Lichtenstein. Planar formulae and their uses. *SIAM Journal Comput.*, 11:329–343, 1982.

[25] J. M. Robson. Algorithms for maximum independent sets. *Journal of Algorithms*, 7(3):425–440, September 1986.

[26] C. Thomassen. Grötzsch's 3-color theorem and its counterparts for the torus and the projective plane. *Journal of Combinatorial Theory, Series B*, 62:268–279, 1994.

Appendix: More details on the proof of Lemma 2

We give a simple modification of the proof of the $\mathcal{NP}$-hardness of PLANAR k-VERTEX COVER as presented by Lichtenstein [24]. In that reduction argument from PLANAR 3SAT, the only possibility to introduce triangles is in the gadget representing clauses. In fact, each such gadget is a triangle, where each vertex represents the occurrence of a literal in that clause. Moreover, these vertices are connected to the vertices representing positive or negative occurrences of variables in the variable gadgets. Obviously, each clause gadget needs two vertices from the gadget in the vertex cover. If one of the literals is indeed "selected" (by putting its neighboring vertex in the corresponding variable gadget into the cover), then the mentioned two necessary vertices in the clause gadget are enough to cover both the edges in the clause gadget and the yet uncovered edges incident with the clause gadget. Conversely, if none of the literals is "selected," then obviously three vertices are needed to cover the clause gadget and its incident edges. This is the core observation for stating that, given a PLANAR 3SAT instance I with m clauses, I is satisfiable if the reduced VERTEX COVER instance I' has a cover of size $5m$.

Now, we replace each triangle clause gadget by a gadget which is a circle of length nine, where each third vertex is representing a literal and is hence connected to the corresponding variable gadget as before. The "non-literal" vertices are not connected to any vertices outside of the gadget. Such a gadget itself necessarily brings five vertices into the cover. Those vertices can be arranged such that two of them are representing literals. But if we require that three of them represent literals (which means that none of the literals occurring in that specific clause is selected), then we will need three further vertices to cover the clause gadget. This is the core observation for stating that, given a PLANAR 3SAT instance I with m clauses, I is satisfiable if the reduced VERTEX COVER instance I' has a cover of size $8m$.

Appendix: An improved algorithm for PLANAR INDEPENDENT SET

In [1], the authors have shown the following result:

Theorem 5.1 *Every connected plane graph with at least two vertices has*

1. *two vertices with degree sum at most 5, or*
2. *two vertices of distance at most two and with degree sum at most 7, or*
3. *a triangular face with two incident vertices with degree sum at most 9, or*
4. *two triangular faces neighbored via an edge $\{u, v\}$ where the sum of the degrees of u and v is at most 11.*

Based on that theorem, we propose the following algorithm:

```
BOOLEAN PIS((V,E),k,S,S'):
IF k <= 0 THEN { S' = S; return TRUE;}
IF V is NOT empty THEN return FALSE;
Let v be a vertex of lowest degree;
IF δ(v) <= 1 THEN return PIS((V \ N[v],E),k-1,S ∪ {v},S');
IF δ(v) <= 4 THEN branch at v // 5 branches
IF δ(v) == 5 THEN // the fourth case of the above theorem applies
  // N(v) = {x,y,u,v_1,v_2}
  // N(u) = {x,y,v,u_1,u_2,u_3}
  // branch 1: v in IS ?
  IF PIS((V \ N[v],E),k-1,S ∪ {v},S') THEN return TRUE;
  // branch 2: v is not in IS, but u
  IF PIS((V \ N[u],E),k-1,S ∪ {u},S') THEN return TRUE;
  // branch 3: v,u is not in IS, but x
  IF PIS((V \ N[x],E),k-1,S ∪ {x},S') THEN return TRUE;
  // branch 4: v,u,x is not in IS, but y
  IF PIS((V \ N[y],E),k-1,S ∪ {y},S') THEN return TRUE;
  stop = FALSE;
  FOR i = 1,2; j = 1,2,3 AND WHILE NOT stop DO
     // branch 5(i,j): v,u,x,y is not in IS, but {v_i,u_j}
    stop=PIS((V \ (N[v_i] ∪ N[u_j]),k-2,S ∪ {v_i,u_j},S');
  return stop;
```

Initially, given an instance (G, k) of PLANAR INDEPENDENT SET, we call PIS with argument $(G, k, \emptyset, S')$, where S' is a set variable in which the solution will be stored. The correctness of the algorithm immediately follows from the quoted theorem. The running time T of the new algorithm, measured in terms of the number of leaves of the search tree, satisfies

$$T(k) \leq 4T(k-1) + 6T(k-2).$$

This recurrence can be solved, showing $T(k) \leq 5.1623^k$ as claimed.

This shows Theorem 4.2.

ECCC ISSN 1433-8092
http://www.eccc.uni-trier.de/eccc
ftp://ftp.eccc.uni-trier.de/pub/eccc
ftpmail@ftp.eccc.uni-trier.de, subject 'help eccc'