Safe approximation and its relation to kernelization

Jiong Guo[1]⋆, Iyad Kanj[2], and Stefan Kratsch[3]⋆⋆

[1] Universität des Saarlandes, Saarbrücken, Germany. jguo@mmci.uni-saarland.de
[2] DePaul University, Chicago, USA. ikanj@cs.depaul.edu
[3] Utrecht University, Utrecht, The Netherlands. kratsch@cs.uu.nl

Abstract. We introduce a notion of approximation, called *safe approximation*, for minimization problems that are subset problems. We first study the relation between the standard notion of approximation and safe approximation, and show that the two notions are different unless some unlikely collapses in complexity theory occur. We then study the relation between safe approximation and kernelization. We demonstrate how the notion of safe approximation can be useful in designing kernelization algorithms for certain fixed-parameter tractable problems. On the other hand, we show that there are problems that have constant-ratio safe approximation algorithms but no polynomial kernels, unless the polynomial hierarchy collapses to the third level.

1 Introduction

Studying the relation between parameterized complexity and approximation theory has attracted the attention of researchers from both areas. Cai and Chen initiated this study by showing that any optimization problem that has a fully polynomial time approximation scheme (FPTAS) is fixed-parameter tractable (FPT) [8]. This result immediately places a large number of optimization problems in the class FPT. Cesati and Trevisan [10] refined Cai and Chen's result by relaxing the condition that the problem has an FPTAS. A problem is said to have an *efficient polynomial time approximation scheme* (EPTAS), if the problem has a PTAS whose running time is of the form $f(1/\epsilon)n^{O(1)}$ (n is the input size and ϵ is the error bound). By definition, an FPTAS for a problem is also an EPTAS. Cesati and Trevisan [10] showed that having an EPTAS is a sufficient condition for a problem to be in FPT. Cai and Chen also showed in [8] that the class MaxSNP of maximization problems, defined by Papadimitriou and Yannakakis [24], and the class Min $F^+\Pi_1$ of minimization problems, defined by Kolaitis and Thakur [19], are subclasses of the class FPT.

In [13], Chen et al. introduced the notion of *efficient fixed-parameter tractability*, and gave a complete characterization of the relation between the class FPTAS

⋆ Supported by the DFG Excellence Cluster "Multimodal Computing and Interaction".

⋆⋆ Supported by the Netherlands Organization for Scientific Research (NWO), project "KERNELS: Combinatorial Analysis of Data Reduction".

and the class FPT. They showed that a parameterized problem has an FPTAS if and only if it is efficient fixed-parameter tractable [13], which complements the earlier result by Cai and Chen [8]. Moreover, to study the relation between EPTAS and FPT, Chen et al. [13] introduced the notion of the *planar W-hierarchy*, and showed that all problems in the *planar W-hierarchy*, which contains several known problems such as PLANAR VERTEX COVER and PLANAR INDEPENDENT SET, have EPTAS.

We note that the parameterized complexity framework has also been used to obtain negative approximation results (see [9, 12, 21], to name a few). For example, the above relations between approximation and parameterized complexity have been used to rule out the existence of EPTAS for certain problems that admit PTAS (see [9, 21]). For an extensive overview on the relation of parameterized complexity and approximation, as well as on combinations of these two paradigms, we refer the interested reader to the recent survey of Marx [22].

More recently, Kratsch [20] studied the relation between kernelization and approximation. He showed that two large classes of problems having constant-ratio approximation algorithms, namely MIN $F^+\Pi_1$ and MAXNP, the latter including MAXSNP, admit polynomial kernelization for their parameterized versions. His result extends Cai and Chen's results [8] mentioned above.

In this paper we investigate further the relation between approximation and kernelization. We focus our attention on minimization problems that are subset problems (i.e., the solution is a subset of the search space), and define the notion of *safe approximation* for subset minimization problems. Informally speaking, an approximation algorithm for a subset minimization problem is *safe* if for every instance of the problem the algorithm returns a solution that is guaranteed to contain (subset containment) an optimal solution. We note that many natural subset minimization problems admit safe approximation algorithms. We start by showing that the notion of safe approximation is different from the standard notion of approximation, in the sense that there are problems that admit approximation algorithms with certain ratios but do not admit safe approximation algorithms even with much worse ratios, under standard complexity assumptions. For example, we show that there are natural problems that have PTASs but do not even have constant-ratio safe approximation algorithms unless $W[1] = \text{FPT}$. We then proceed to study the relation between safe approximation and kernelization. We demonstrate, through some nontrivial examples, that the notion of safe approximation can be very useful algorithmically: we show how safe approximation algorithms for certain problems can be used to design kernelization algorithms for their associated parameterized problems. On the other hand, we show that safe approximation does not imply polynomial kernelization by proving that there are problems that have constant-ratio safe approximation algorithms but whose associated parameterized problems do not have polynomial kernels, unless the polynomial hierarchy collapses to the third level.

Due to the lack of space, most proofs are deferred to the full version of the paper.

2 Preliminaries

Parameterized complexity and kernelization. A *parameterized problem* Q is a subset of $\Sigma^* \times \mathbb{N}$, where Σ is a finite fixed alphabet and $\mathbb{N}$ is the set of non-negative integers. Therefore, each instance of the parameterized problem Q is a pair (x, k), where the second component, i.e., the non-negative integer k, is called the *parameter*. We say that the parameterized problem Q is *fixed-parameter tractable* [16], shortly FPT, if there is an algorithm that decides whether an input (x, k) is a member of Q in time $f(k)|x|^{O(1)}$, where $f(k)$ is a recursive function of k. Let FPT denote the class of all fixed-parameter tractable problems. A parameterized problem Q is *kernelizable* if there exists a polynomial-time reduction, the *kernelization*, that maps instances (x, k) of Q to other instances (x', k') of Q such that: (1) $|x'| \leq g(k)$, (2) $k' \leq g(k)$, for some recursive function g, and (3) (x, k) is a yes-instance of Q if and only if (x', k') is a yes-instance of Q. The instance (x', k') is called the *kernel* of (x, k). A kernelization is *polynomial* if $g(k)$ is bounded by a polynomial in k.

A hierarchy of fixed-parameter intractability, *the W-hierarchy* $\bigcup_{t\geq 0} W[t]$, has been introduced. Here, $W[t] \subseteq W[t+1]$ for all $t \geq 0$ and the 0-th level $W[0]$ is the class FPT. The hardness and completeness notions have been defined for each level $W[i]$ of the W-hierarchy, for $i \geq 1$ [16]. It is commonly believed that collapses in the W-hierarchy are unlikely (i.e., $W[i] \neq W[i-1]$, for any integer $i \geq 1$), and in particular, $W[1] \neq$ FPT (see [16]).

NP-optimization problems and approximability. An *NP optimization problem* Q is a 4-tuple (I_Q, S_Q, f_Q, g_Q), where: I_Q is the set of input instances, which is recognizable in polynomial time. For each instance $x \in I_Q$, $S_Q(x)$ is the set of feasible solutions for x, which is defined by a polynomial p and a polynomial-time computable predicate π (p and π depend only on Q) as $S_Q(x) = \{y : |y| \leq p(|x|) \wedge \pi(x, y)\}$. The function $f_Q(x, y)$ is the objective function mapping a pair $x \in I_Q$ and $y \in S_Q(x)$ to a non-negative integer. The function f_Q is computable in polynomial time. The function g_Q is the *goal* function, which is one of the two functions $\{\max, \min\}$, and Q is called a *maximization problem* if $g_Q = \max$, or a *minimization problem* if $g_Q = \min$. We will denote by $opt_Q(x)$ the value $g_Q\{f_Q(x, z) \mid z \in S_Q(x)\}$, and if there is no confusion about the underlying problem Q, we will write $opt(x)$ to denote $opt_Q(x)$.

In this paper we restrict our attention to optimization problems in NP that are minimization problems. An algorithm A is an *approximation algorithm* for a minimization problem Q if for each input instance $x \in I_Q$ the algorithm A returns a feasible solution $y_A(x) \in S_Q(x)$. The solution $y_A(x)$ has an *approximation ratio* $r(|x|)$ if it satisfies the following condition:

$$f_Q(x, y_A(x))/opt_Q(x) \leq r(|x|).$$

The approximation algorithm A has an *approximation ratio* $r(|x|)$ if for every instance x in I_Q the solution $y_A(x)$ constructed by the algorithm A has an approximation ratio bounded by $r(|x|)$.

An optimization problem Q has a *constant-ratio approximation algorithm* if it has an approximation algorithm whose ratio is a constant (i.e., independent from the input size). An optimization problem Q has a *polynomial time approximation scheme* (PTAS) if there is an algorithm A_Q that takes a pair (x, ϵ) as input, where x is an instance of Q and $\epsilon > 0$ is a real number, and returns a feasible solution y for x such that the approximation ratio of the solution y is bounded by $1 + \epsilon$, and for each fixed $\epsilon > 0$, the running time of the algorithm A_Q is bounded by a polynomial of $|x|$. Finally, an optimization problem Q has a *fully polynomial time approximation scheme* (FPTAS) if it has a PTAS A_Q such that the running time of A_Q is bounded by a polynomial of $|x|$ and $1/\epsilon$.

Definition 1. Let $Q = (I_Q, S_Q, f_Q, g_Q)$ be a minimization problem. The *parameterized version of* Q is $Q_{\leq} = \{(x, k) \mid x \in I_Q \wedge opt_Q(x) \leq k\}$. A parameterized algorithm A_Q *solves the parameterized version of* Q if on any input $(x, k) \in Q_{\leq}$, A_Q returns "yes" with a solution y in $S_Q(x)$ such that $f_Q(x, y) \leq k$, and on any input not in $Q_{\leq}$, A_Q simply returns "no".

The above definition allows us to consider the parameterized complexity of a minimization problem Q, which is the parameterized complexity of $Q_{\leq}$.

The problems discussed in the current paper all share the property that they seek a subset, of a given set (a "search space"), that satisfies certain properties. We call such problems *subset problems*. Most of the problems studied in parameterized complexity and combinatorial optimization are subset problems.[1]

3 Safe approximation

In this section we define a notion of approximation for subset minimization problems that we call *safe approximation*, and we study its relation to the standard notion of approximation.

Definition 2. Let Q be a subset minimization problem. An approximation algorithm $\mathcal{A}$ for Q is said to be *safe* if for every instance x of Q, $\mathcal{A}$ returns a solution $y_A(x)$ such that there exists an optimal solution $S_{opt}(x)$ of x satisfying $S_{opt}(x) \subseteq y_A(x)$. The notions of *constant-ratio safe approximation algorithm*, *safe PTAS*, and *safe FPTAS* are defined in a natural way.

Informally speaking, an approximation algorithm for a minimization subset problem is safe if the solution that it returns is guaranteed to contain an optimal solution.

Some natural questions to ask are the following: (1) Are there (NP-hard) subset minimization problems that admit safe approximation algorithms with

[1] In the case of optimization problems the subset sought is one that minimizes/maximizes the objective function, among all subsets satisfying the required properties. For most problems considered in this paper, the objective function is the cardinality of the subset sought.

“small” ratios? (2) Does every problem that has a constant-ratio approximation algorithm (resp. PTAS/FPTAS) have a constant-ratio safe approximation algorithm (resp. safe PTAS/FPTAS)?

The answer to question (1) is positive: many minimization problems admit safe approximation algorithms with “small” ratios (e.g., constant ratios). Those problems include VERTEX COVER (follows from a well-known theorem of Nemhauser and Trotter [4, 23]), many subset minimization problems on bounded-degree graphs (for many such problems we can simply return the whole set of vertices as the approximate solution), and many subset minimization problems on planar graphs (e.g., PLANAR DOMINATING SET).

We show next that, unless some unlikely collapses in complexity theory or parameterized complexity occur, the answer to question (2) is negative. First, we define the following problems.

A *vertex cover* in an undirected graph is a subset of vertices C such that every edge in the graph is incident to at least one vertex in C. The CONNECTED VERTEX COVER problem is: Given an undirected graph G, compute a subset of vertices C of minimum cardinality such that C is a vertex cover of G and the subgraph of G induced by C is connected.

A *dominating set* in an undirected graph is a subset of vertices D such that every vertex in the graph is either in D or has a neighbor in D. The DOMINATING SET problem is: Given an undirected graph G, compute a subset of vertices D of minimum cardinality such that D is a dominating set of G. A *unit disk graph* (UDG) is a graph on n points/vertices in the Euclidean plane such that there is an edge between two points in the graph if and only if their Euclidean distance is at most 1 (unit). The DOMINATING SET problem on UDGs, denoted UDG-DOMINATING SET, is the DOMINATING SET problem restricted to UDG's.

We answer question (2) negatively by showing that the DOMINATING SET problem, which has an approximation ratio $\lg n + 1$ [17] (n is the number of the vertices in the graph), is unlikely to have a safe approximation algorithm of ratio $c \lg n$, for any constant $c > 0$:

Theorem 1. *Unless $FPT = W[2]$,* DOMINATING SET *does not have a safe approximation algorithm of ratio $\rho \leq c \lg n$, for any constant $c > 0$.*[2]

Proof. Let (G, k) be an instance of DOMINATING SET$_{\leq}$. Suppose that DOMINATING SET has a safe approximation algorithm $\mathcal{A}$ of ratio $c \lg n$. We run $\mathcal{A}$ on G to obtain a solution D of G such that $|D|/|opt(G)| \leq c \lg n$. If $|D| > ck \lg n$, it follows that $opt(G) > k$, and we can reject the instance (G, k); so assume $|D| \leq ck \lg n$. Since $\mathcal{A}$ is a safe approximation algorithm, D contains a minimum dominating set. Therefore, in time $\sum_{i=1}^{k} \binom{ck \lg n}{i} n^2$ we can enumerate all subsets of D of size at most k, and check whether any of them is a dominating set. If we find any, then we accept the problem instance (G, k); otherwise, we reject

[2] As a matter of fact, under the same complexity hypothesis, we can rule out (using a similar proof) the existence of a safe approximation algorithm of ratio $n^{o(1)}$ for DOMINATING SET.

it. This shows that DOMINATING SET$_\leq$ is solvable in $f(k)n^c$ time for some constant c and completes the proof. □

By a similar argument, it follows that there are problems that have a PTAS but that are unlikely to have even a safe constant-ratio approximation algorithm:

Theorem 2. *The* UDG-DOMINATING SET *problem admits a PTAS but does not admit a constant-ratio safe approximation algorithm unless* $W[1] = FPT$.

Finally, the CONNECTED VERTEX COVER problem, which has an approximation algorithm of ratio 2 [25], does not admit a constant-ratio safe approximation algorithm unless the polynomial time hierarchy collapses to the third level:

Theorem 3. *Unless the polynomial time hierarchy collapses to the third level, the* CONNECTED VERTEX COVER *problem does not have a constant-ratio safe approximation algorithm.*[3]

4 Kernelization and safe approximation

At the surface, the notion of safe approximation seems to be closely related to the notion of kernelization in parameterized complexity. We clarify some of the differences between the two notions in the following remark.

Remark 1. It seems intuitive that problems with a safe approximation algorithm should have kernels of matching size. Of course, the two notions are not equivalent: The safe approximation solution is not necessarily a kernel, as simply "forgetting" everything outside the solution cannot be guaranteed to give an equivalent instance. Furthermore, kernelizations are not restricted to subproblems of the original instance and, hence, do not have to return a safe approximation. Still, even if one aims to compute a safe approximation and cleverly reduce the part outside the solution to small size, it can be showed (Theorem 4) that there are problems with constant-factor safe approximation but without polynomial kernels (assuming that the polynomial hierarchy does not collapse).

Remark 2. If a subset minimization problem has a constant-ratio safe approximation algorithm (in fact, any ratio of the form $f(opt)$ suffices, where f is a nondecreasing efficiently computable function) then its parameterized version must be FPT (enumerate all subsets of the solution returned by the safe approximation algorithm in FPT time).

The HITTING SET problem is defined as follows. Given a pair $(S, \mathcal{F})$ where S is a set of elements and $\mathcal{F}$ is a family of subsets of S, compute a smallest subset H of S that intersects every set in $\mathcal{F}$.

[3] Under the same complexity hypothesis, we can strengthen this result to rule out the existence of a safe approximation algorithm of ratio $opt(G)^{O(1)}$ for CONNECTED VERTEX COVER.

Theorem 4. *Unless the polynomial-time hierarchy collapses to the third level, there are problems that have constant-ratio safe approximation algorithms but no polynomial kernels.*

Proof. Consider the following restriction of HITTING SET, denoted PAIRED-HS, consisting of the set of all instances of HITTING SET of the form $(S, \mathcal{F})$, where $|S| = 2N$ for some natural number N, and $\mathcal{F}$ contains, in addition to other sets, N pairwise disjoint sets, each of cardinality 2, whose union is S. It is not difficult to see that the instances of PAIRED-HS are recognizable in polynomial time (e.g., by computing maximum matching). Moreover, it follows easily from the definition of PAIRED-HS that it has a safe approximation algorithm of ratio 2 (the algorithm returns the set S as the solution to the instance $(S, \mathcal{F})$). Note also that PAIRED-HS$_{\leq}$ is FPT, since any instance in which the parameter is smaller than $|S|/2$ can be rejected immediately, otherwise, a brute force algorithm enumerating all subsets of S and checking whether each subset is a solution, is an FPT algorithm that solves the problem.

We claim that PAIRED-HS$_{\leq}$ does not have a polynomial kernel[4], unless the polynomial hierarchy collapses to the third level. To prove this claim, consider the d-SAT problem that consists of the set of instances of CNF-SAT in which each clause has at most d literals, where $d \geq 3$ is an integer constant. It was shown in [15] that, unless the polynomial time hierarchy collapses to the third level, the d-SAT problem parameterized by the number of variables n, has no oracle communication protocol of cost at most $O(n^{d-\epsilon})$, for any $\epsilon > 0$; this can be easily seen to exclude also kernels as well as compressions into instances of other problems of size $O(n^{d-\epsilon})$ (cf. [15]).

Now proceed by contradiction. Assume that PAIRED-HS$_{\leq}$ has a polynomial kernel of size $O(k^c)$ for some integer constant $c > 1$, and consider the d-SAT problem where $d = c + 1$. We can reduce d-SAT to PAIRED-HS as follows. For each instance F on n variables, construct the instance $(S, \mathcal{F}, n)$ (with parameter n) of PAIRED-HS$_{\leq}$ where S consists of the set of n variables in F and their negations; thus, S has $2n$ elements. For each variable in F we associate a set of two elements in $\mathcal{F}$ containing the variable and its negation. Finally, for each clause in F we associate a set in $\mathcal{F}$ containing the literals in the clause. Clearly, the resulting instance is an instance of PAIRED-HS$_{\leq}$. Moreover, F is a yes-instance of d-SAT if and only if $(S, \mathcal{F}, n)$ is a yes-instance of PAIRED-HS$_{\leq}$; the key observation is that the paired elements and the maximum size of n encode the selection of a truth assignment, the other sets check that it is satisfying. It follows that this reduction compresses instances of d-SAT into instances of size $O(n^c) = O(n^{d-1})$, which implies a collapse of the polynomial hierarchy to the third level. This completes the proof. □

In the remainder of this section we study further the relation between safe approximation and kernelization. We show that the notion of safe approximation can be useful for obtaining kernelization algorithms for FPT problems. The VERTEX COVER problem is a trivial example showing how a safe approximation

[4] The kernel size for HITTING SET$_{\leq}$ is the sum of the cardinalities of all sets in $\mathcal{F}$.

algorithm can be used to obtain a kernelization algorithm: no edge has both endpoints outside the safe approximation solution, and if an edge has one, we may safely take the other.[5] The NT-theorem [4, 23], which is a local-ratio approximation algorithm of ratio 2 for VERTEX COVER, is at the same time a safe approximation algorithm. This algorithm has been used in [14] to obtain a kernel for VERTEX COVER of size at most $2k$, which currently stands as the best upper bound on the kernel size for VERTEX COVER.

It is not always as simple to get a kernelization from a safe approximation algorithm as in the case of VERTEX COVER. Therefore, it is interesting to investigate which safe approximation algorithms (for subset minimization problems) can be turned into kernelization algorithms. In addition to its theoretical importance, this question has an interesting algorithmic facet: given a solution to the instance that contains an optimal solution (the "important" part), can we "deal with" the remaining part of the instance (the "left overs")?

We illustrate next, through a few examples, how the existence of safe approximation algorithms implies the existence of kernelization algorithms for certain problems. These results should mainly be seen as illustrative examples of using safe approximation as a technique for obtaining kernelization algorithms; in most cases matching or better kernels are known. The problems under consideration are: EDGE MULTICUT, VERTEX MULTICUT, PLANAR DOMINATING SET, PLANAR FEEDBACK VERTEX SET, FEEDBACK VERTEX SET, and a generalization of FEEDBACK VERTEX SET, called FEEDBACK VERTEX SET WITH BLACKOUT VERTICES. Both PLANAR DOMINATING SET and PLANAR FEEDBACK VERTEX SET admit PTAS [3], and FEEDBACK VERTEX SET and its generalization with blackout vertices admit approximation algorithms of ratio 2 [2]. Both PLANAR DOMINATING SET$_\leq$ [1] and PLANAR FEEDBACK VERTEX SET$_\leq$ [6] have linear kernels, and FEEDBACK VERTEX SET has a quadratic kernel [26].

4.1 Planar dominating set

The PLANAR DOMINATING SET problem is the DOMINATING SET problem restricted to planar graphs. We show next that any safe approximation algorithm of ratio ρ for PLANAR DOMINATING SET can be used to design a kernelization algorithm for PLANAR DOMINATING SET$_\leq$ that computes a kernel with at most $10\rho k$ vertices. For a vertex v in a graph, we denote by $N(v)$ the set of neighbors of v. Two vertices u and v in a graph are said to be *twins* if $N(u) = N(v)$.

Theorem 5. *If* PLANAR DOMINATING SET *has a safe approximation algorithm* $\mathcal{A}$ *of constant ratio* ρ *then* PLANAR DOMINATING SET$_\leq$ *has a kernelization algorithm* $\mathcal{A}'$ *that computes a kernel with at most* $10\rho k$ *vertices.*

Proof. Given an instance (G, k) of PLANAR DOMINATING SET$_\leq$, the kernelization algorithm $\mathcal{A}'$ starts by invoking the algorithm $\mathcal{A}$ to compute a set of vertices S of G whose cardinality is at most $\rho|opt(G)|$, and that contains a minimum

[5] This is also true for the d-HITTING SET; we may forget all elements that are outside the safe approximation solution, and shrink the sets accordingly.

dominating set of G. If $|S| > \rho k$ then clearly $opt(G) > k$ and the algorithm $\mathcal{A}'$ rejects the instance (G, k); so assume $|S| \leq \rho k$. Let $\overline{S} = V(G) \setminus S$. The algorithm $\mathcal{A}'$ applies the following reduction rules to G in the respective order.

Reduction Rule 1 Remove all the edges in $G[\overline{S}]$.

Reduction Rule 2 For any set of degree-1 vertices (degree taken in the current graph) in $\overline{S}$ that are twins, remove all of them except one vertex.

Reduction Rule 3 For any set of degree-2 vertices in $\overline{S}$ that are twins (i.e., all of them are twins), remove all of them except two vertices.

Let G' be the resulting graph from G after the application of the above rules. Note that $S \subseteq V(G')$. The algorithm $\mathcal{A}'$ returns the instance (G', k). Since S contains an optimal solution, it is not difficult to verify that (G', k) is an equivalent instance of (G, k). Next, we upper bound the number of vertices in G'.

Let $I = V(G') \setminus S$, and note that I is an independent set by Reduction Rule 1. We partition I into three sets: I_1 is the set of degree-1 vertices (degree taken in G'), I_2 is the set of degree-2 vertices, and $I_{\geq 3}$ is the set of vertices in I of degree at least 3. Next, we upper bound the cardinality of each of these three sets.

To upper bound the cardinality of $I_{\geq 3}$, we define the multihypergraph $\mathcal{H}$ as follows. The vertex-set of $\mathcal{H}$ is S. A subset of vertices e is an edge in $\mathcal{H}$ if and only if there exists a vertex $u \in I_{\geq 3}$ such that $N(u) = e$. Since the incidence graph of $\mathcal{H}$ is a subgraph of G', and hence is planar, the multihypergraph $\mathcal{H}$ is planar. It follows from Lemma 4.4 in [18] that $\mathcal{H}$ has at most $2|V(\mathcal{H})| - 4 = 2|S| - 4$ edges. Since the number of edges in $\mathcal{H}$ is exactly the number of vertices in $I_{\geq 3}$, it follows that $|I_{\geq 3}| \leq 2|S| - 4$. By Reduction Rule 2, we have $|I_1| \leq |S|$. To upper bound $|I_2|$, we construct a planar multigraph $\mathcal{G}$ whose vertex set is S, and such that there is an edge between two vertices u and v in $\mathcal{G}$ if and only if there exists a vertex $w \in I_2$ whose neighbors are u and v. Since G' is planar, $\mathcal{G}$ is planar, and by Reduction Rule 3, there are at most 2 edges between any two vertices in $\mathcal{G}$. It follows from Euler's formula that the number of edges in $\mathcal{G}$, and hence the number of vertices in I_2, is at most $2(3|V(\mathcal{G})| - 6) = 6|S| - 12$.

Thus $|V(G')| = |I| + |S| \leq 10|S| - 16 < 10\rho k$, completing the proof. □

4.2 Feedback vertex set

Let G be an undirected graph. A set of vertices F in G is a *feedback vertex set* of G if the removal of F breaks all cycles in G, that is, if $G - F$ is acyclic. The FEEDBACK VERTEX SET problem is to compute a feedback vertex set of minimum cardinality in a given graph. The PLANAR FEEDBACK VERTEX SET problem is the restriction of the FEEDBACK VERTEX SET problem to planar graphs. We show first that a constant-ratio safe approximation for FEEDBACK VERTEX SET gives a kernel with a cubic number of vertices for FEEDBACK VERTEX SET$_{\leq}$, using only one reduction rule plus a simple marking procedure. We then consider a generalization of FEEDBACK VERTEX SET$_{\leq}$, which allows for the presence of blackout vertices, and asks for a feedback vertex set excluding all blackout vertices. We call this problem FEEDBACK VERTEX SET WITH BLACKOUT VERTICES, FVSBV for short. This problem was introduced by Bar-Yehuda [5], and has applications in Bayesian inference. We show that the cubic kernel can be improved for this generalization to match the known quadratic kernel by Thomassé [26], as the blackout annotation allows a more efficient processing of the trees that are outside the safe approximation, using simpler and different arguments. (Note that the quadratic upper bound does not carry to the standard FEEDBACK VERTEX SET$_\leq$ problem due to the presence of blackout vertices.) Finally, we show that a ratio ρ safe approximation for PLANAR FEEDBACK VERTEX SET gives a kernel with at most $3\rho k$ vertices for PLANAR FEEDBACK VERTEX SET$_\leq$.

Theorem 6. *If* FEEDBACK VERTEX SET *has a constant-ratio safe approximation, then* FEEDBACK VERTEX SET$_\leq$ *has a cubic kernel.*

Corollary 1. *If* FVSBV *has a constant-ratio safe approximation, then* FVSBV$_\leq$ *admits a quadratic kernel.*

Theorem 7. *If* PLANAR FEEDBACK VERTEX SET *has a safe approximation algorithm with constant ratio ρ then* PLANAR FEEDBACK VERTEX SET$_\leq$ *has a kernel with at most $3\rho k$ vertices.*

4.3 Multicut problems

The EDGE MULTICUT problem is defined as follows: Given a graph $G = (V, E)$ and a set of pairs $T = \{(s_1, t_1), \ldots, (s_\ell, t_\ell)\}$ of vertices in G, compute a set of edges E' in G of minimum cardinality whose removal disconnects all pairs in T (i.e., there is no path from s_i to t_i, for $i = 1, \ldots, \ell$, in $(V, E \setminus E')$).

Theorem 8. *If* EDGE MULTICUT *has an $f(opt)$ safe approximation algorithm, where f is a nondecreasing efficiently computable function, then* EDGE MULTICUT$_\leq$ *has a polynomial kernel with at most $3f(k)$ vertices.*

A similar result holds for VERTEX MULTICUT$_\leq$, where the task is to delete at most k non-terminal vertices to disconnect all given terminal pairs.

Theorem 9. *If* VERTEX MULTICUT *has an $f(opt)$ safe approximation algorithm, where f is a nondecreasing efficiently computable function, then* VERTEX MULTICUT$_\leq$ *has a polynomial kernel with at most $2f(k)$ vertices.*

5 Conclusion

We presented the notion of safe approximation and studied its relation to the notion of kernelization in parameterized complexity.

Even though we have shown that the notions of safe approximation and kernelization are different for subset minimization problems, we illustrated through some nontrivial examples how safe approximation can be useful for obtaining kernelization algorithms. Some of those results imply linear kernelization algorithms for the problems under consideration. For example, it can be shown that PLANAR DOMINATING SET has a constant-ratio safe approximation algorithm, which, when combined with Theorem 5, gives a linear kernelization algorithm for PLANAR DOMINATING SET$_{\leq}$. Unfortunately, the obtained upper bound on the kernel size does not come close to the currently-best upper bound on the kernel size for PLANAR DOMINATING SET$_{\leq}$ [11]. This, however, may not be discouraging due to the mere fact that kernelization algorithms for PLANAR DOMINATING SET$_{\leq}$ have been extensively studied, whereas the notion of safe approximation was not considered before. Maybe a celebrated example that can be used to illustrate how safe approximation can be useful for designing kernelization algorithms is the example of VERTEX COVER. An approximation algorithm of ratio 2, the NT-theorem, for VERTEX COVER existed since 1975 [23]. Buss and Goldsmith [7], in 1993, presented a kernelization algorithm that gives a quadratic ($2k^2$) kernel for VERTEX COVER$_{\leq}$. This upper bound on the kernel size was subsequently used in several parameterized algorithms for VERTEX COVER$_{\leq}$, until Chen et al. [14] observed in 2001 that the approximation algorithm given by the NT-theorem is *safe* (this notion was not defined at that point), and implies a $2k$ kernel for VERTEX COVER$_{\leq}$. We believe that the existence of the notion of safe approximation may bridge the gap between approximation and kernelization.

Several interesting questions arise from the current research. Many parameterized problems admit polynomial kernels and their optimization versions have constant-ratio approximation algorithms. Do these optimization versions admit constant-ratio safe approximation algorithms? For example, FEEDBACK VERTEX SET has a ratio 2 approximation algorithm [2] and a quadratic kernel [26], does it have a constant-ratio safe approximation algorithm? One can ask whether a sufficient condition (based on parameterized complexity) exists, such that if a problem satisfying this condition has an approximation algorithm then it must have a safe approximation algorithm.

Acknowledgment. The authors would like to thank Bart Jansen for suggesting the nice example used in the proof of Theorem 4.

References

1. J. Alber, M. Fellows, and R. Niedermeier. Polynomial-time data reduction for dominating set. *J. ACM*, 51(3):363–384, 2004.
2. V. Bafna, P. Berman, and T. Fujito. A 2-approximation algorithm for the Undirected Feedback Vertex Set problem. *SIAM J. Discrete Math.*, 12(3):289–297, 1999.
3. B. Baker. Approximation algorithms for NP-complete problems on planar graphs. *J. ACM*, 41(1):153–180, 1994.
4. R. Bar-Yehuda and S. Even. A local-ratio theorem for approximating the Weighted Vertex Cover problem. *Annals of Discrete Mathematics*, 25:27–46, 1985.
5. R. Bar-Yehuda, D. Geiger, J. Naor, and R. Roth. Approximation algorithms for the Feedback Vertex Set problem with applications to constraint satisfaction and bayesian inference. *SIAM J. Comput.*, 27(4):942–959, 1998.

6. H. Bodlaender and E. Penninkx. A linear kernel for planar feedback vertex set. In *IWPEC*, volume 5018 of *LNCS*, pages 160–171. Springer, 2008.
7. J. Buss and J. Goldsmith. Nondeterminism within P. *SIAM J. Comput.*, 22:560–572, 1993.
8. L. Cai and J. Chen. Fixed parameter tractability and approximability of NP-hard optimization problems. *J. Comput. Syst. Sci.*, 54:465–474, 1997.
9. L. Cai, M. Fellows, D. Juedes, and F. Rosamond. The complexity of polynomial-time approximation. *Theory Comput. Syst.*, 41(3):459–477, 2007.
10. M. Cesati and L. Trevisan. On the efficiency of polynomial time approximation schemes. *Inf. Process. Lett.*, 64:165–171, 1997.
11. J. Chen, H. Fernau, I. Kanj, and G. Xia. Parametric duality and kernelization: Lower bounds and upper bounds on kernel size. *SICOMP*, 37(4):1077–1106, 2007.
12. J. Chen, X. Huang, I. Kanj, and G. Xia. Linear FPT reductions and computational lower bounds. *J. Comput. Syst. Sci.*, 72(8):1346–1367, 2006.
13. J. Chen, X. Huang, I. Kanj, and G. Xia. Polynomial time approximation schemes and parameterized complexity. *Discrete Appl. Mathematics*, 155(2):180–193, 2007.
14. J. Chen, I. Kanj, and W. Jia. Vertex cover: further observations and further improvements. *J. Algorithms*, 41:280–301, 2001.
15. H. Dell and D. van Melkebeek. Satisfiability allows no nontrivial sparsification unless the polynomial-time hierarchy collapses. In *STOC*, pages 251–260, 2010.
16. R. Downey and M. Fellows. *Parameterized Complexity.* Springer, New York, 1999.
17. D. Johnson. Approximation algorithms for combinatorial problems. *J. Comput. Syst. Sci.*, 9:256–278, 1974.
18. I. Kanj, M. Pelsmajer, G. Xia, and M. Schaefer. On the induced matching problem. In *STACS*, volume 08001 of *LIPIcs*, pages 397–408, 2008.
19. P. Kolaitis and M. Thakur. Approximation properties of NP minimization classes. *J. Comput. Syst. Sci.*, 50:391–411, 1995.
20. S. Kratsch. Polynomial kernelizations for MIN $F^+\Pi_1$ and MAX NP. In *STACS*, volume 3 of *LIPIcs*, pages 601–612, 2009.
21. D. Marx. Efficient approximation schemes for geometric problems?. In *ESA*, volume 3669 of *LNCS*, pages 448–459. Springer, 2005.
22. D. Marx. Parameterized complexity and approximation algorithms. *Comput. J.*, 51(1):60–78, 2008.
23. G. Nemhauser and L. Trotter. Vertex packing: structural properties and algorithms. *Mathematical Programming*, 8:232–248, 1975.
24. C. Papadimitriou and M. Yannakakis. Optimization, approximation, and complexity classes. *J. Comput. Syst. Sci.*, 43:425–440, 1991.
25. C. Savage. Depth-first search and the vertex cover problem. *Inf. Process. Lett.*, 14(5):233–237, 1982.
26. S. Thomassé. A $4k^2$ kernel for feedback vertex set. *ACM Transactions on Algorithms*, 6(2), 2010.