

The Complexity of Polynomial-Time Approximation ¹

Liming Cai², Michael Fellows³, David Juedes⁴, and Frances Rosamond⁵

Abstract

In 1996, Khanna and Motwani [KM96] proposed three logic-based optimization problems constrained by planar structure, and offered the hypothesis that these putatively fundamental problems might provide insight into characterizing the class of optimization problems that admit a polynomial-time approximation scheme (PTAS). The main contribution of this paper is to explore this program from the point of view of parameterized complexity.

Problems of optimization are naturally parameterized by the parameter $k = 1/\epsilon$. An optimization problem admits a PTAS if and only if there is an algorithm Φ , that, given an input of size n , and a parameter value $k = 1/\epsilon$, produces a solution that is within a multiplicative factor of $(1 + \epsilon)$ of optimal, in a running time that is polynomial for every fixed value of ϵ . This definition admits the possibility that the degree of the polynomial that bounds the running time of Φ may increase, even quite dramatically, as a function of $k = 1/\epsilon$. In fact, amongst the PTASs that are currently known, for an error of 20%, polynomial time bounds no better than $O(n^{2000})$ are quite common [Dow03]. Viewing $k = 1/\epsilon$ as a problem parameter, in the sense of parameterized complexity, leads naturally to the question of whether an *efficient PTAS* (EPTAS) might be possible for a given optimization problem. An EPTAS is simply an FPT algorithm with respect to this parameter.

We offer a number of results concerning the problems PLANAR TMIN, PLANAR TMAX and PLANAR MPSAT defined by Khanna and Motwani:

- (1) We show that each of these problems of approximation, naturally parameterized by $k = 1/\epsilon$, is hard for $W[1]$, and thus it is highly unlikely that they admit EPTASs. (One could also interpret this as indicating that PTASs for these problems are unlikely to be a useful way of coping with intractability in the sense of [GJ79].)
- (2) We show that there are EPTASs for some subproblems described by syntactic restrictions, and establish some limits on how far these positive results can be extended.

CLASSIFICATION: computational and structural complexity.

1 Introduction

Polynomial-time approximation provides various approaches for producing acceptably good solutions to intractable optimization problems. A general approach is the notion of a polynomial-time approximation schemes (PTAS) that provides a solution with cost that is within a multiplicative factor of $(1 + \epsilon)$ of optimal, where the error ϵ can be chosen arbitrarily close to zero [GJ79]. While many problems do not have polynomial-time approximation schemes unless $P = NP$, for those that do, approximation is performed by algorithms that provide additional precision at the expense of additional running time. The running time for such algorithms is typically of the form $n^{O(1/\epsilon)}$ or $2^{O(1/\epsilon)}n^c$, where $c > 0$ is a constant. While both time bounds are exponential in $\frac{1}{\epsilon}$, the time

¹This work was supported in part by the National Science Foundation research grant CCR-000246 (support of the first author), and by the National Science and Engineering Research Council of Canada and the New Zealand Marsden Fund (support of the second author).

²Department of Computer Science, The University of Georgia, Athens, Georgia 30602-7404 USA, lcai@cs.uga.edu

³School of Electrical Engineering and Computer Science, University of Newcastle, University Drive, Callaghan, NSW 2308, Australia, mfellows@cs.newcastle.edu.au

⁴School of Electrical Engineering and Computer Science, Ohio University, Athens, OH 45701 USA, juedes@ohiou.edu

⁵Retreat for the Arts and Sciences, Newcastle, NSW 2300, Australia, retreat4artscience@yahoo.com

bound $n^{O(1/\epsilon)}$ may become intolerable for even moderate values of n and reasonable error bounds ϵ . For this reason, some work has been done on obtaining PTAS's with more efficient running times, such as $2^{O(1/\epsilon)}n$. For some optimization problems, the worst case running time of PTAS algorithms have been improved through more sophisticated techniques or by more thorough analysis, yielding PTASs where the exponent of the polynomial does not increase with the error bound [GGR96, Ar97, Ma05b, DH05, DH06]. For many other optimization problems that admit PTASs, however, the existence of a PTAS where the degree of the polynomial running time does not depend on the degree of approximation — that is, an EPTAS — remains a challenging open question.

Khanna and Motwani [KM96] introduced three proposed fundamental syntactic (i.e., logic-based) classes of optimization problems (the classes may also be viewed simply as “very general” optimization problems) in a program that aimed to characterize a large number of optimization problems on planar structures. Furthermore, they showed that a number of optimization problems that, at the surface, are *not* about planar structure, and *not* about logic, admitted PTASs that could (surprisingly) be understood as based on this “hidden planar logical structure”. In a related line of research, Hunt *et al.* [HMRRRS98] investigated many problems with near planar structures and many problems on geometric graphs. All of these problems admit a PTAS with a running time of the form $O(n^{g(1/\epsilon)})$ for some function g .

For many years it was not known whether these problems admit a PTAS with a more efficient running time such as $2^{O(1/\epsilon)}n$ (an EPTAS). In the present paper, we show that an EPTAS for the three planar logic optimization problems of Khanna and Motwani is impossible unless $FPT = W[1]$. In a recent paper, Daniel Marx has shown a similar result for the problems on geometric graphs considered by Hunt *et al.* [Ma05a]. Results such as these are based on the techniques and results recently developed in parameterized complexity theory.

Parameterized complexity was introduced by Downey and Fellows [DF99] to study computational problems for which efficient algorithms for small or moderate parameters are natural and useful. Research results show that parameterized tractability and polynomial-time approximability are closely related. In particular, under a natural parameterization framework, Cai and Chen [CC97] observed that the existence of a fully polynomial-time approximation scheme (FPTAS) for an optimization problem implies fixed-parameter tractability for the naturally associated parameterized decision problem, where the parameter is the cost (or value) of a solution.

Bazgan [Baz95], and Cesati and Trevisan [CT97] independently strengthened this connection, introducing the notion of an *efficient polynomial-time approximation scheme* (EPTAS), which is precisely fixed-parameter tractability of the optimization problem with respect to the natural parameterization $k = 1/\epsilon$. According to [CT97], an optimization problem admits an EPTAS if it admits a PTAS of running time $f(1/\epsilon)p(n)$, where f is a function and $p(n)$ is some polynomial. What Bazgan [Baz95] and Cesati and Trevisan [CT97] showed is that the weaker hypothesis that an optimization problem admits an EPTAS implies that the associated parameterized decision problem is fixed-parameter tractable.

Fixed-parameter tractability for the parameterized problem where the parameter is the cost (or value) of a solution is thus a necessary condition for an optimization problem to admit an EPTAS. In other words, if the associated parameterized decision problem is $W[1]$ -hard, then the optimization problem cannot admit an EPTAS unless $FPT = W[1]$. We use this connection in this paper in order to explore whether the planar logic problems introduced by Khanna and Motwani admit EPTASs.

The converse, of course, does not hold: if the associated parameterized decision problem is fixed-parameter tractable, this does not imply that the optimization problem admits an EPTAS. For example, all MAX SNP-complete problems are fixed-parameter tractable [CC97], but they do not admit even a PTAS, much less an EPTAS, unless $P = NP$.

In the present paper we develop techniques to obtain upper and lower bounds on the parameterized complexity for various optimization problems on planar structures. These results have direct implications on the existence of EPTAS as well as upper and lower time bounds on EPTAS for these problems. Our research focuses on a large number of (non-weighted) optimization problems captured by the syntactic classes PLANAR MPSAT, PLANAR TMIN, and PLANAR TMAX which were introduced by Khanna and Motwani [KM96]. In particular, our work and technical results address the following issues.

- (1) *The hardness of admitting an EPTAS.* PLANAR MPSAT, PLANAR TMIN, and PLANAR TMAX are natural extensions from planar versions of the better-known classes Max SNP, Min $F^+ \Pi_2$, and RMAX(2), respectively, by allowing a minterm (i.e., a conjunction of literals) in place of each literal. Our research shows that the existence of an EPTAS for such problems is very sensitive to the size of each minterm. We are able to show that PLANAR TMIN₄ (PLANAR TMIN restricted to minterms of size four or smaller), PLANAR TMAX, and PLANAR MPSAT are $W[1]$ -hard, and hence the possibility of an EPTAS is excluded, unless $FPT = W[1]$.
- (2) *The application of parameterized tractability methods in obtaining EPTASs.* Polynomial-time approximation schemes for many optimization problems on planar graphs were developed based on the method of finding a small separator and then solving each of the separated components optimally [Bak94]. Such an idea is generalized in [KM96] for problems in PLANAR TMIN, PLANAR TMAX, and PLANAR MPSAT. We observe that since solving each component can be done by a standard tree decomposition-based algorithmic schema [Bod88, Bod97], parameterized algorithms employing a standard tree decomposition-based schema can be used to solve each component optimally, thus leading to an EPTAS. In particular, we prove that essentially all problems in PLANAR TMIN₁, including PLANAR VERTEX COVER and PLANAR DOMINATING SET, are solvable in time $2^{O(\sqrt{k})}n$.

Almost all of the above results in (1) and (2) apply to the classes PLANAR TMAX and PLANAR MPSAT.

The paper is organized as follows. In §2, we introduce the necessary concepts from parameterized complexity theory and establish the connection between an EPTAS and parameterized tractability. In §3, we give $W[1]$ -hardness proofs for problems in PLANAR TMIN, PLANAR TMAX, and PLANAR MPSAT, thereby demonstrating the existence of problems in these classes that do not admit an EPTAS unless $FPT = W[1]$. In §4, EPTASs are described for subclasses of problems in PLANAR TMIN, PLANAR TMAX, and PLANAR MPSAT. We summarize our results in §5, and point to some open questions.

2 Preliminaries

Here we briefly introduce the necessary concepts concerning optimization problems and the theory of parameterized complexity. For additional information, we refer readers to the comprehensive text on parameterized complexity by Downey and Fellows [DF99] and the classic text on NP-completeness by Garey and Johnson [GJ79].

A parameterized problem Π is defined over the set $\Sigma^* \times \mathbb{N}$, where Σ is a finite alphabet and \mathbb{N} is the set of natural numbers. Therefore, each instance of the problem Π is a pair $\langle I, k \rangle$, where k is called the *parameter*.

Definition 2.1 A parameterized problem Π is fixed-parameter tractable (*FPT*) if there is an algorithm running in time $O(f(k)p(|I|))$ that solves the parameterized problem Π for some polynomial p and some function f .

Parameterized tractability is usually stated in terms of a decision problem, i.e., determining whether some instance $\langle I, k \rangle \in \Pi$. However, it is occasionally necessary to consider algorithms that produce witnesses. In this context, we consider algorithms that produce witnesses to the fact that $\langle I, k \rangle \in \Pi$, if such a witness exists, in $O(f(k)p(|I|))$ steps. In this case, we say that Π is *parameterized tractable with witness*, or that Π is *solvable with witness* in time $O(f(k)p(|I|))$. This slightly stronger definition of parameterized tractability will be used in §4.

The complexity class *FPT* contains all fixed-parameter tractable problems. Parameterized problems are classified into the *W*-hierarchy $FPT \subseteq W[1] \subseteq W[2] \subseteq \dots \subseteq W[P]$. Complete problems for these classes are defined based on reductions that preserve parameterized tractability [DF99]. For example, INDEPENDENT SET and CLIQUE are $W[1]$ -complete. The k -STEP HALTING PROBLEM for nondeterministic Turing machines (of unlimited nondeterminism) is also complete for $W[1]$, and hence (the parameterized analog of Cook's Theorem) the hypothesis that $FPT \neq W[1]$ rests on much the same intuitive foundation as the conjecture $P \neq NP$ (for a discussion, see [Dow03]).

Many parameterized problems are naturally obtained from optimization problems through parameterizations. Following the earlier work of Cai and Chen [CC97], we use a standard parameterization of optimization problems. For each optimization problem Π , the *standard parameterized (decision) version* Π^* of Π is to determine, given an instance I of Π and an integer k , whether the optimal solution cost $OPT_{\Pi}(I)$ is $\geq k$ for maximization problems or $\leq k$ in the case of a minimization problem.

Definition 2.2 ([CT97]) An optimization problem admits an efficient polynomial-time approximation scheme (*EPTAS*) if for each fixed error $\epsilon > 0$, there is an (uniform) $f(1/\epsilon)p(|I|)$ -time approximation algorithm A that guarantees $|A(I) - OPT(I)| \leq \epsilon OPT(I)$ for every given instance I , where $f(k)$ is a function and $p(n)$ is some polynomial.

The fact that the existence of an EPTAS implies parameterized tractability was first shown by Bazgan [Baz95] and later (and independently) by Cesati and Trevisan [CT97]. The following proposition provides a detailed account of this result in terms of time complexity.

Proposition 2.1 Let Π be an NP optimization problem. If Π admits an $O(f(1/\epsilon)n^c)$ -time EPTAS, then the associated standard parameterized decision version Π^* is solvable in time $O(f(2k)n^c)$.

We next present our notation concerning the syntactic classes defined by Khanna and Motwani [KM96]. To begin, given a collection of variables, a *minterm* is simply a conjunction of literals. A literal is *positive* if it is x_i for some variable x_i . A literal is *negative* if it is $\neg x_j$ for some variable x_j . A minterm is *positive (negative)* if all of the literals are positive (negative). A first order formula (FOF) is a disjunction of minterms. An FOF F is *positive (negative)* if each minterm in F is positive (negative). The *width* of an FOF is the number of minterms in the formula. The *size* of a minterm is the number of literals in the minterm.

Definition 2.3 Let $s(n), w(n)$ be two functions in n .

- (1) $TMIN_{s(n)}^{w(n)}$ is the class of all NP optimization problems that can be expressed as follows: given a collection C of positive FOFs over n variables with width bounded by $w(n)$ and the maximum size of minterms bounded by $s(n)$, find a minimum weighted truth assignment T that satisfies all FOFs in C .

- (2) $\text{TMAX}_{s(n)}^{w(n)}$ is the class of all NP optimization problems that can be expressed as follows: given a collection C of negative FOFs over n variables with width bounded by $w(n)$ and the maximum size of minterms bounded by $s(n)$, find a maximum weighted truth assignment T that satisfies all FOFs in C .
- (3) $\text{MPSAT}_{s(n)}^{w(n)}$ is the class of all NP-optimization problems that can be expressed as follows: given a collection C of FOFs over n variables with width bounded by $w(n)$ and the maximum size of minterms bounded by $s(n)$, find a truth assignment T that maximizes the number of FOFs in C that are satisfied.

Notice that the size of any satisfiable minterm is bounded by n . Hence, we will only consider values of $s(n) \leq n$.

Given a collection C of FOFs over n variables, the *incidence graph* of C is the bipartite graph with edges between the set of FOFs and the set of variables such that there is an edge between a formula and a variable if and only the variable occurs in the formula. Let the syntactic class \mathcal{C} be either MPSAT, TMIN, or TMAX. Then Planar \mathcal{C} is the class of problems in \mathcal{C} restricted to instances with planar incidence graphs.

We now define the following subclasses of problems. Let $\mathcal{C} \in \{\text{TMAX}, \text{TMIN}, \text{MPSAT}\}$. Planar $\mathcal{C}_{s(n)} = \bigcup_{w \in \text{poly}} \text{Planar } \mathcal{C}_{s(n)}^{w(n)}$, Planar $\mathcal{C}^{w(n)} = \text{Planar } \mathcal{C}_n^{w(n)}$, Planar $\mathcal{C} = \bigcup_{w \in \text{poly}} \text{Planar } \mathcal{C}^{w(n)}$, and Planar $\mathcal{C}^{\text{polylog}} = \bigcup_{c \geq 0} \text{Planar } \mathcal{C}^{\log^c n}$.

In §4, we address the complexity of a generic *canonical problem* for each of these classes, defined as follows.

Definition 2.4 Let $s(n)$ and $w(n)$ be two functions. Define

- (1) $\text{PLANAR KTMIN}_{s(n)}$: given a collection of FOFs over n variables with each minterm size bounded by $s(n)$, find a minimum weight assignment that satisfies all the FOFs.
- (2) $\text{PLANAR KTMIN}^{w(n)}$: given a collection of FOFs over n variables with each formula width bounded by $w(n)$, find a minimum weight assignment that satisfies all the FOFs.
- (3) $\text{PLANAR KTMIN}^{\text{polylog}}$: given a collection of FOFs over n variables with each formula width bounded by $\log^c n$ for some $c \geq 0$, find a minimum weight assignment that satisfies all the FOFs.
- (4) PLANAR KTMIN : given a collection of FOFs over n variables, find a minimum weight assignment that satisfies all the FOFs.

Canonical problems for the corresponding subclasses of PLANAR TMAX and PLANAR MPSAT are defined in a similar fashion.

3 The hardness of achieving EPTAS

In this section, we show that there are problems in PLANAR TMIN₄, the subclasses of PLANAR TMIN with restriction to minterms of size at most 4, which do not admit an EPTAS unless $FPT = W[1]$. We consider the canonical problem PLANAR KTMIN₄ in the class PLANAR TMIN₄ and prove $W[1]$ -hardness for its parameterized complexity. Similar results hold for PLANAR TMAX and PLANAR MPSAT.

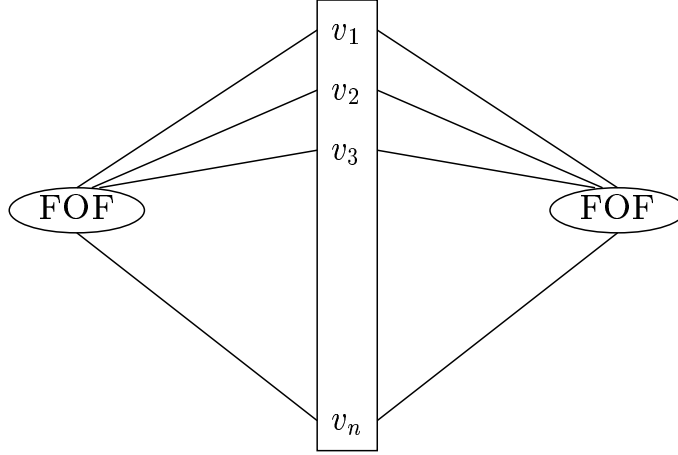


Figure 1: A block of n variables.

The $W[1]$ -hardness proofs for these canonical problems are based on parameterized reductions from CLIQUE. Briefly, let $\langle G, k \rangle$ be an instance of CLIQUE. Assume that G has n vertices. From G and k , we construct a collection C of FOFs over $f(k)$ blocks of n variables. C will contain at most $2f(k)$ FOFs and the incidence graph of C will be planar. In the case of PLANAR KTMIN, each minterm in each FOF will contain at most 4 variables. The collection C is constructed so that G has a clique of size k if and only if C has a weight $f(k)$ satisfying assignment with exactly one variable set to true in each block of n variables. Here we have that $f(k) = O(k^4)$.

Theorem 3.1 PLANAR KTMIN₄ is $W[1]$ -hard.

Proof: We show that CLIQUE is parameterized reducible to PLANAR KTMIN₄. Since CLIQUE is $W[1]$ -complete, it will follow that PLANAR KTMIN₄ is $W[1]$ -hard.

To begin, let $\langle G, k \rangle$ be an instance of CLIQUE. Assume that G has n vertices. From G and k , we will construct a collection C of FOFs over $f(k)$ blocks of n variables. C will contain at most $2f(k)$ FOFs and the incidence graph of C will be planar. Moreover, each minterm in each FOF will contain at most 4 variables. The collection C is constructed so that G has a clique of size k if and only if C has a weight $f(k)$ satisfying assignment with exactly one variable set to true in each block of n variables. Here we have that $f(k) = O(k^4)$. To maintain planarity in the incidence graph for C , we ensure that each block of n variables appears in at most 2 FOFs. If this condition is maintained, then we can draw each block of n variables as seen in Figure 4.

We describe the construction in two stages. In the first stage, we use k blocks of n variables and a collection C' of $k(k-1)/2 + k$ FOFs. In a weight k satisfying assignment for C' , exactly one variable $v_{i,j}$ in each block of variables $b_i = [v_{i,1}, \dots, v_{i,n}]$ will be set to true. We interpret this event as “vertex j is the i th vertex in the clique of size k .” The $k(k-1)/2 + k$ FOFs are described as follows. For each i with $1 \leq i \leq k$, let f_i be the FOF $\bigvee_{j=1}^n v_{i,j}$. This FOF ensures that at least one variable in b_i is set to true. For each pair $1 \leq i < j \leq k$, let $f_{i,j}$ be the FOF $\bigvee_{(u,v) \in E} v_{i,u} v_{j,v}$. Each FOF $f_{i,j}$ ensures that there is an edge in G between the i th vertex in the clique and the j th vertex in the clique.

It is straightforward to argue that $C' = \{f_1, \dots, f_k, f_{1,2}, \dots, f_{k-1,k}\}$ has a weight k satisfying assignment if and only if G has a clique of size k . First notice that any weight k satisfying assignment

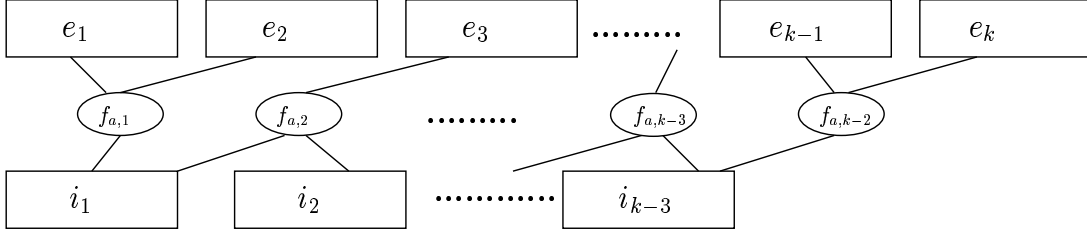


Figure 2: The widget A_k .

for C' must satisfy exactly 1 variable in each block b_i . Each first order formula $f_{i,j}$ ensures that there is an edge between the i th vertex in the potential clique and the j th vertex in the potential clique. Notice too that, since we assume that G does not contain edges of the form (u, u) , the FOF $f_{i,j}$ also enforces that the i th vertex in the potential clique is not the j th vertex in the potential clique. This completes the first stage.

This first stage can be drawn in the shape of the complete graph K_k on k vertices. To see this, place each block of variables b_i at vertex v_i of K_k and label each edge (i, j) in K_k by the FOF $f_{i,j}$. Finally, attach each FOF f_i to vertex v_i on the exterior. Notice that this drawing of the incidence graph for this collection of FOFs is not planar. We fix this problem in the second stage.

In the second stage we achieve planarity by removing crossovers in the incidence graph for C' . We use two types of widgets to remove crossovers while keeping the number of variables per minterm bounded by 4. The first widget A_k consists of $k + k - 3$ blocks of n variables and $k - 2$ FOFs. This widget consists of $k - 3$ internal and k external blocks of variables. Each external block $e_i = [e_{i,1}, \dots, e_{i,n}]$ of variables is connected to exactly one FOF inside the widget. Each internal block $i_j = [i_{j,1}, \dots, i_{j,n}]$ is connected to exactly two FOFs inside the widget. The $k - 2$ FOFs are given as follows. The FOF $f_{a,1}$ is $\bigvee_{j=1}^n e_{1,j}e_{2,j}i_{1,j}$. For each $2 \leq l \leq k - 3$, the FOF $f_{a,l} =$

$\bigvee_{j=1}^n i_{l-1,j}e_{l+1,j}i_{l,j}$. Finally, $f_{a,k-2} = \bigvee_{j=1}^n i_{k-3,j}e_{k-1,j}e_{k,j}$. These $k - 2$ FOFs ensure that the settings of variables in each block is the same if there is a weight $2k - 3$ satisfying assignment to the $2k - 3$ blocks of n variables. The widget A_k is shown in Figure 2.

Since each internal block is connected to exactly two FOFs, the incidence graph for this widget can be drawn on the plane without crossing any edges.

The second widget removes crossover edges from the first stage of the construction. In the first stage, crossovers in our drawing can occur in the incidence graph because two FOFs cross from one block of variables to another. To eliminate this, consider each edge i, j in K_k with $i < j$ as a directed edge from i to j . In the next stage of the construction, we send a copy of the variables in block i to block j . At each crossover point from the direction of blocks $u = [u_1, \dots, u_n]$ and $v = [v_1, \dots, v_n]$, insert a widget B that introduces 2 new blocks of n variables $u_1 = [u_{1,1} \dots u_{1,n}]$ and $v_1 = [v_{1,1} \dots v_{1,n}]$ and a FOF $f_B = \bigvee_{j=1}^n \bigvee_{l=1}^n u_j u_{1,j} v_l v_{1,l}$. The FOF f_B ensures that u_1 and v_1 are copies of u and v . As shown in Figure 3, the incidence graph for the widget B is also planar.

To complete the construction, we replace each of the original k blocks of n variables from the first stage with a copy of the widget A_k . At each crossover point in the graph, we introduce a copy of widget B . We attach each f_i to its associated A_k , attaching it to one of the external blocks (leaving $k - 1$ of these blocks to communicate with the other $k - 1$ “vertex choice” gadgets of the same kind). Finally, for each directed edge between block i and j , we insert the original FOF $f_{i,j}$

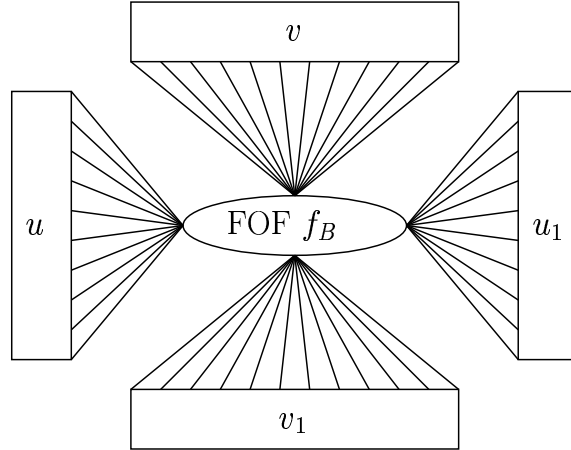


Figure 3: A planar drawing of the widget B .

between the last widget B and the destination widget A_k . Since one of the new blocks of variables created by the widget B is a copy of block i , the effect of the FOF $f_{i,j}$ in this new collection of FOFs is the same as before.

Figure 4 provides a diagram that shows the full construction when $k = 5$. Since the incidence graph of each widget in this drawing is planar, the entire collection C of first order formulas has a planar incidence graph.

Now, if we assume that there are $c(k) = O(k^4)$ crossover points in standard drawing of K_k , then our collection has $c(k)$ B widgets. Since each B widget introduces 2 new blocks of n variables, this gives $2c(k)$ new blocks. Since we have k A_k widgets, each of which has $2k - 3$ blocks of n variables, this gives an additional $k(2k - 3)$ blocks. So, in total, our construction has $f(k) = 2c(k) + 2k^2 - 3k = O(k^4)$ blocks of n variables. Note also that there are $g(k) = k(k - 1)/2 + k + k(k - 2) + c(k) = O(k^4)$ FOFs in the collection C .

As shown in our construction C has a weight $f(k)$ satisfying assignment (i.e., each block has exactly one variable set to true) if and only if the original graph G has a clique of size k . Since the incidence graph of C is planar and each minterm in each FOF contains at most four variables, it follows that this construction is a parameterized reduction from CLIQUE to PLANAR KTMIN₄. This completes the proof. \square

Theorem 3.2 PLANAR KTMAX is $W[1]$ -hard.

Proof: The proof is very similar to the proof of Theorem 3.1. Given an instance $\langle G, k \rangle$ of CLIQUE, convert that instance into the collection of FOFs C as described in the proof of Theorem 3.1. Replace each positive variable v_i in each minterm with a conjunction $\neg v_1 \neg v_2 \cdots \neg v_{i-1} \neg v_{i+1} \cdots \neg v_n$ of negated variables from the same block. The process converts each positive minterm to a negative minterm. Note that this new collection C' has weight $f(k)$ satisfying assignment if and only if G has a clique of size k . Since $\langle C', f(k) \rangle$ is an instance of PLANAR KTMAX, this gives a parameterized reduction from CLIQUE to PLANAR KTMAX. \square

Theorem 3.3 PLANAR KMPSAT is $W[1]$ -hard.

Proof: The proof is very similar to the proof of Theorem 3.1. Given an instance $\langle G, k \rangle$ of CLIQUE, convert that instance into the collection of FOFs C as described in the proof of Theorem 3.1. For

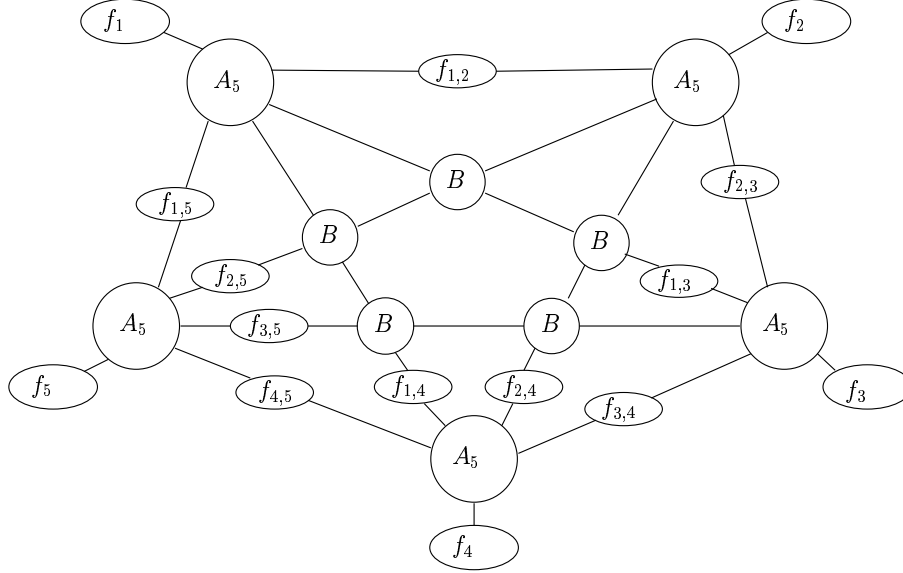


Figure 4: The construction when $k = 5$.

each FOF f_i in C , replace f_i with the formula

$$f'_i = \bigvee_{j=1}^n \neg v_{i,1} \neg v_{i,2} \dots \neg v_{i,j-1} \neg v_{i,j+1} \dots \neg v_{i,n},$$

which creates a new collection C' of FOFs. The incidence graph for this new collection of FOFs is still planar. Moreover, notice that each formula f'_i is true when at most one variable in each block is set to true. Therefore, there is a satisfying assignment to all the $g(k)$ FOFs in the collection if and only if G has a clique of size k . Since $\langle C', g(k) \rangle$ is an instance of PLANAR KMPSAT₄, this gives a parameterized reduction from CLIQUE to PLANAR KMPSAT. \square

Note that from the results of Khanna and Motwani [KM96], all three problems PLANAR KTMIN₄, PLANAR KTMAX, and PLANAR KMPSAT have polynomial-time approximation schemes. However, as we show here, these problems do not have efficient polynomial-time approximation schemes unless $FPT = W[1]$.

Corollary 3.1 PLANAR KTMIN₄, PLANAR KTMAX, and PLANAR KMPSAT do not have efficient polynomial approximation schemes unless $FPT = W[1]$.

Proof: By Proposition 2.1, the existence of an efficient polynomial-time approximation scheme for PLANAR KTMIN₄, PLANAR KTMAX, and PLANAR KMPSAT implies that parameterized versions of these problems are in FPT. Since each of these problems is $W[1]$ -hard, this implies that $FPT = W[1]$. \square

4 Upper bounds on the running time for EPTAS

In this section we describe some positive results regarding the running time of EPTASs. The work in this section uses the following concepts.

Definition 4.1 *An outerplanar graph is a planar graph that has an embedding on the plane with all vertices appearing on the outer face. An r -outerplanar graph is an outerplanar graph when $r = 1$, or a $(r - 1)$ -outerplanar graph by deleting all vertices on the outer face when $r > 1$.*

The layer L_1 of an r -outerplanar graph consists of the vertices on the boundary of the outer face, and for $i > 1$, the layer L_i is the set of vertices that lie on the boundary of the outer face in the embedding of the subgraph $G - (L_1 \cup \dots \cup L_{i-1})$.

Intuitively, a graph is r -outerplanar if it has an “onion structure” of depth r — r removals of “the outer layer” suffice to reduce the graph to the empty graph.

Definition 4.2 *A tree decomposition D of a graph $G = (V, E)$ consists of a tree $T = (I, F)$ and a collection of subsets of V , $\{X_i \mid i \in I\}$, one for each node in the tree T , that collectively satisfy the following conditions:*

1. $\bigcup_{i \in I} X_i = V$, i.e., each $v \in V$ is in some X_i ,
2. for each edge $(u, v) \in E$, there exists some i such that $u, v \in X_i$, and
3. for each $v \in V$, the set $\{i \in I \mid v \in X_i\}$ induces a subtree of T .

Item (3) is often written as “for all $i, j, k \in I$, if j lies on the path between i and k , then $X_i \cap X_k \subseteq X_j$.” Notice that this implies that the set of vertices in the subtrees below some bag X_i are disjoint except for the vertices in X_i .

The *width* of a tree decomposition D is the maximum cardinality of any X_i in D . The *treewidth* of a graph G is the minimum width needed by every tree decomposition D of G . The notions of outerplanarity and treewidth are closely connected.

Proposition 4.1 ([Bod97]) *An r -outerplanar graph has treewidth at most $3r - 1$.*

Polynomial-time approximation schemes for many problems on planar structures (e.g., on planar graphs) can be developed based on the method of finding a small separator and then solving each separated component optimally [Bak94, KM96]. Such an idea has been generalized by Khanna and Motwani for problems in PLANAR TMIN, PLANAR TMAX, and PLANAR MPSAT as follows: given a t -outerplanar embedding of a planar incidence graph, decompose the layers L_1, \dots, L_t of the t -outerplanar graph into r disconnected components, each consisting of at most c/ϵ layers, where $r = O(\epsilon t)$. Then, find an optimal solution for each component and assemble the solutions for all the components to form a $(1 + \epsilon)$ -approximation for the original instance. Since each component is an $O(1/\epsilon)$ -outerplanar graph, it has treewidth $O(1/\epsilon)$ by Proposition 4.1. A *standard tree decomposition-based algorithmic schema* (basically, bounded treewidth dynamic programming) [Bod88, KM96, Bod97, ABFKN00] is applied to obtain a PTAS. In general, for problems in PLANAR TMIN, PLANAR TMAX, and PLANAR MPSAT, the running time of these PTAS is $O(n^{O(1/\epsilon)})$.

On the other hand, many recent parameterized algorithms for problems on planar graphs, such as PLANAR DOMINATING SET [ABFKN00], have been developed via the tree decomposition-based algorithmic schema. In particular, in order to give a positive answer to the relationship between $OPT(I)$ and the parameter k (where I is an instance of the optimization problem), the incidence graph has to have treewidth bounded by $w(k)$ for some function w . This observation leads to the following technical lemma. (For related results, see [DH05, DH06].)

Lemma 4.1 *Let Π be an optimization problem in PLANAR TMIN, PLANAR TMAX, or PLANAR MPSAT. If Π^* is solvable with witness in time $O(f(w(k))p(n))$ via the standard tree decomposition-based algorithmic schema and a positive answer to the relationship between $OPT(I)$ and k (where*

I is an instance of P_i), implies that the incidence graph has treewidth bounded by $w(k)$, then Π has an EPTAS running in time

$$O\left(\frac{1}{\epsilon} f\left(O\left(\frac{1}{\epsilon}\right)\right) p(n)\right)$$

for some polynomial p .

Proof: The proof of this lemma follows the approach given in the work of Baker [Bak94] and Khanna and Motwani [KM96], with some minor modifications. Here, we give the proof for PLANAR TMIN. The proofs for PLANAR TMAX and PLANAR MPSAT are similar. Note that the theorem holds only for the unweighted versions of these problems.

Let Π be a problem in PLANAR TMIN, and let I be an instance of Π . Then, I can be cast as a problem of determining a minimum weighted assignment to n variables that satisfies a collection C of m first order formulae. Since Π is in PLANAR TMIN, the incidence graph G for C is planar. Since G has $n + m$ vertices and each layer has at least one vertex, we have that G is t -outerplanar for some $t \leq n + m$. So, let L_1, \dots, L_t be the layers in G . Notice that it is possible to build G and then create the layers L_1, \dots, L_t in polynomial-time by first computing a planar embedding of G , and then iteratively removing the outermost layers of G .

We describe an efficient polynomial-time approximation scheme for Π . Let $p = \lceil \frac{1}{\epsilon} \rceil$, and notice that $\frac{1}{p} \leq \epsilon$. We break the layers of G into overlapping groups of size at most $2p + 1$. Let S_j be the graph that consists of the layers $L_{2jp+2i-1}$ to $L_{2(j+1)p+2i}$ for some i , $1 \leq i \leq p$, where j ranges through all possible values where this makes sense. The graphs S_j and S_{j+1} overlap at layers $L_{2(j+1)p+2i-1}$ and $L_{2(j+1)p+2i}$. For each S_j , build the subproblem C'_j consisting of just the FOFs in the layers $L_{2jp+2i-1}$ through $L_{2(j+1)p+2i}$ whose variables also appear in these same layers. This guarantees that all FOFs in layers L_{2jp+2i} through $L_{2(j+1)p+2i-1}$ appear in C'_j . We now solve each of these subproblems C'_j exactly using the parameterized algorithm mentioned in the statement of theorem. Since each S_j is $2p + 1$ -outerplanar graph, we know that it has treewidth bounded by that of the incidence graph of C'_j (which is a subgraph of S_j) and therefore is at most $3(2p + 1) - 1$ by Proposition 4.1. Since Π^* is solvable in time $O(f(w(k))p(n))$ steps via the standard tree decomposition-based algorithm, we can solve the problem exactly in $O(f(\tau)p(n))$ steps, where τ is the treewidth of the problem. Since each subproblem has treewidth $\tau \leq 3(2p + 1) - 1 = 6p + 2$, the minimum weighted value of the solution can be found in $O(f(6p + 2)p(n))$ steps.

Given $\epsilon > 0$, let $p = \lceil \frac{1}{\epsilon} \rceil$. Since each minterm in the collection of FOFs C is positive, the union of the solutions to each C'_j provides a solution for C . We repeat this process for each i ranging from 1 to p , and we return the minimum weighted solution. The total running time of this algorithm is

$$O\left(\frac{1}{\epsilon} f\left(O\left(\frac{1}{\epsilon}\right)\right) p(n)\right)$$

To see that the solution is within $\frac{1}{p}$ of optimal, consider an optimal solution for C . Notice that this is a solution for each C'_j . So, since each C'_j was solved optimally, we have that the total cost of the solution produced by this algorithm is bounded below by the cost of the optimal solution plus the number of variables set in the overlapping layers (layers $L_{2jp+2i-1}$ and L_{2jp+2i}). By the pigeonhole principle, we know for some i , $1 \leq i \leq p$, that the total number of variables set to true in the overlapping layers $L_{2jp+2i-1}$ and L_{2jp+2i} is at most $\frac{OPT(C)}{p}$. Hence the solution produced by this algorithm has weight at most $OPT(C) + \frac{OPT(C)}{p} \leq (1 + \epsilon)OPT(C)$. \square

To obtain an EPTAS or time upper bounds on EPTAS for problems under consideration, it suffices to consider the parameterized tractability via the standard tree-decomposition algorithmic schema.

Lemma 4.2 *Let $G = (V_1 \cup V_2, E)$ be a planar bipartite graph such that there exists a $V' \subseteq V_1$ of size k which dominates every vertex in V_2 . Then G has treewidth $O(\sqrt{k})$.*

Proof: We can assume that G has no isolated vertices. Therefore V' is a 2-dominating set in G . That is, every vertex in $V_1 \cup V_2$ is at a distance of at most 2 from a vertex of V' . The lemma follows from the bidimensionality theory recently developed by Demaine and Hajiaghayi [DH04, DH05, DH06]. More specifically, the lemma follows immediately from Theorem 3 of [DH06] (Theorem 5 of [DH05]), since 2-domination is contraction bidimensional. \square

We next use Lemma 4.2 to show that subclasses of PLANAR TMIN are parameterized tractable. We apply Lemma 4.2 to show that the incidence graphs for all problems in PLANAR TMIN have small treewidth.

Theorem 4.1 $\text{PLANAR TMIN}_1 \subseteq \text{FPT}$. *Moreover, for every problem Π in PLANAR TMIN_1 , Π has a $O(2^{O(\sqrt{k})}p(n))$ parameterized algorithm.*

Proof: Let Π be any problem in PLANAR TMIN_1 . Let I be an instance of the problem and k be an integer. Then I can be expressed as a collection C of first-order formula. Let x_1, \dots, x_n be the set of variables in C and let f_1, \dots, f_m be the first-order formula in C . Then the incidence graph is a planar bipartite graph $G = (V_{var} \cup V_{fof}, E)$, where $V_{var} = \{v_{x_1}, \dots, v_{x_n}\}$ and $V_{fof} = \{v_{f_1}, \dots, v_{f_m}\}$.

If there is an weight k truth assignment that satisfies every formula f_j , $j = 1, \dots, m$, then there exists a subset $D \subseteq V_{var}$ of size k in G that dominates every vertex in the set V_{fof} . The vertices in D can be picked such that a vertex v_x is in D if and only the corresponding variable x is assigned the value “true.” Because each formula f is satisfied (by the weight k truth assignment) and all x_{i1}, \dots, x_{it} occur in f as positive literals, at least one of these variable must be assigned with value “true” and that corresponding vertex dominates the vertex v_f .

By Lemma 4.2, the treewidth for the incidence graph is $O(\sqrt{k})$. Since the size of each minterm is one, we can construct a $2^{O(\sqrt{k})}N$ -time algorithm to determine if there is an weight k truth assignment that satisfies every formula in C by using the standard dynamic programming approach [Bod88]. For each node X_i in the tree-decomposition of the incidence graph, build a table with one entry for every possible setting of the variables and FOFs in X_i . The table stores the value of the minimum weighted satisfying assignment in the boolean formula constructed from the subtree rooted at X_i that is consistent with the given table entry. Note that since each minterm has size one, we can use true or false as a possible setting for each FOF. This gives a table of size at most $2^{O(\sqrt{k})}$. The minimum weighted satisfying assignment can be found by building the tables for each node in the tree decomposition in a bottom-up fashion. This approach is very similar to the algorithms given in the work of Khanna and Motwani [KM96]. \square

The next theorem requires a somewhat straightforward result concerning parameterized tractability found in the work of Cai and Juedes [CJ01].

Proposition 4.2 ([CJ01]) *If a parameterized problem Π is solvable in time $O(2^{O(s(n)k})p(n))$ for some unbounded and nondecreasing function $s(n) = o(\log n)$ and some polynomial p , then it is parameterized tractable.*

Theorem 4.2 $\text{PLANAR TMIN}^{\text{polylog}} \subseteq \text{FPT}$.

Proof: The proof is similar to the proof of Theorem 4.1. Given a problem Π in $\text{PLANAR TMIN}^{\text{polylog}}$, an instance I of Π can be represented as a collection of FOF C , where the width of each FOF in C is bounded by $(\log n)^c$, for some constant c .

As in the proof of the previous theorem, we can show that the treewidth of the incidence graph for C is $O(\sqrt{k})$. Since the width of each minterm is bounded by $(\log n)^c$, we can construct a $(\log n)^c O(\sqrt{k})n$ -time algorithm to determine if there is a weight k truth assignment that satisfies every formula in C by using the standard dynamic programming approach [Bod88]. For each node X_i in the tree-decomposition of the incidence graph, build a table with one entry for every possible setting of the variables and FOFs in X_i . The table stores the value of the minimum weighted satisfying assignment in the boolean formula constructed from the subtree rooted at X_i that is consistent with the given table entry. Note that since the width of each minterm is bounded by $(\log n)^c$, we can use the name of the minterm that satisfies it as a possible setting for each FOF. This gives a table of size at most $(\log n)^c O(\sqrt{k})$. The minimum weighted satisfying assignment can be found by building the tables for each node in the tree decomposition in a bottom-up fashion. This approach is very similar to the algorithms given in the work of Khanna and Motwani [KM96].

The running-time of this algorithm is $2^{\log(\log n)^c O(\sqrt{k})}N$, where N is the total input size. Since $\log \log n = o(\log n)$, it follows from Proposition 4.2 that Π is in FPT. \square

Similar results are given for PLANAR TMAX and PLANAR MPSAT.

Theorem 4.3 PLANAR TMAX₁ \subseteq FPT and PLANAR TMAX ^{polylog} \subseteq FPT.

Proof: The proof follows the proofs of the previous theorems. Let Π be an optimization problem in PLANAR TMAX₁ or PLANAR TMAX ^{polylog}. It suffices to show that the incidence graph for each instance of this problem has small treewidth. Let I be an instance of Π and assume that $OPT(I) = k$. Then I can be expressed by a collection C of m first-order formula, each a disjunction of negative minterms over n variables.

We first show that when the maximum weight of satisfying assignments is k , the incidence graph for C must be an $O(k)$ -outerplanar graph. Without loss of generality, assume that in each formula f no variable occurs in every minterm for some FOF – otherwise, such a variable must be assigned the value “false” and the collection C can be simplified by eliminating such a variable. Now, consider the incidence graph of C . It must be r -outerplanar for some $r \leq n + m$. Let L_1, \dots, L_r be the layers in this r -outerplanar graph. Notice that, because of the planarity of the graph, the first order formulae in layers L_{i-1} and L_{i+2} must be separated in the sense that no variable can appear at the same time in both a FOF in L_{i+2} and in a set of variables in L_{i-1} or vice versa. Also notice that, since no layer is empty and the incidence graph is bipartite, each layer, except possibly the last layer, contains at least one variable and at least one FOF.

The layers of the r -outerplanar incidence graph can thus be partitioned into groups $\{L_1, L_2, L_3\}$, $\{L_4, L_5, L_6\}$, \dots , $\{L_{3t-2}, L_{3t-1}, L_{3t}\}$, where $t = \lceil r/3 \rceil$, such that layer the FOF in L_{3j+i} are separated from the FOF in layer $L_{3(j-1)+i}$ and layer $L_{3(j+1)+i}$ for each $i = 1, 2, 3$ and $j = 0, \dots, t - 1$. Consider some FOF f in layer L_{3j+i} . Since no variable occurs in every minterm in f , at least one variable x that appears in f can be assigned the value “true” such that all f is satisfied. (For example, we can simply select one variable from each layer L_{3j+1} for $j = 0, \dots, t - 1$, set them to “true” and set all other variables to “false”. In this way, every FOF has at most 1 variable in it set to “true”, and thus it is satisfied.) The value of x has no effect on FOFs in layers $L_{3(j+1)+i}$ and $L_{3(j-1)+i}$. Now, if we set all other variables to be false, this is a satisfying assignment to the collection C of first order formula. Therefore, at least t variables can be assigned the value “true.” This implies that $r \leq 3t \leq 3k$ since k is the maximum number of variables assigned the value “true.” Therefore the incidence graph can be at most $3k$ -outerplanar.

According to Proposition 4.1, the incidence graph has treewidth $9k - 1$ when the maximum weight for satisfying assignments is k . Using the associated tree decomposition in the standard dynamic programming algorithm gives the result. \square

Theorem 4.4 $\text{PLANAR MPSAT}_1 \subseteq \text{FPT}$ and $\text{Planar MPSAT}^{\text{polylog}} \subseteq \text{FPT}$.

Proof: The proof is similar to the proof for Theorem 4.3, except that we are trying to maximize the number of FOF that are satisfied. Let C be a collection of FOF, and assume that the maximum number of FOFs that can be satisfied is k for any the truth assignment to the variables in C . We will show that the incidence graph for C is at most $3k$ outerplanar.

As in the proof of Theorem 4.3, we can break the incidence graph into r layers L_1, \dots, L_r . The first order formulae in layer L_{i-1} must be separated from the first order formula in layer L_{i+2} in the sense that they do not share variables. The layers of the r -outerplanar incidence graph can thus be partitioned into groups $\{L_1, L_2, L_3\}, \{L_4, L_5, L_6\}, \dots, \{L_{3t-2}, L_{3t-1}, L_{3t}\}$, where $t = \lceil r/3 \rceil$, such that the FOF in the layer L_{3j+i} are separated from the FOF in layer $L_{3(j-1)+i}$ and layer $L_{3(j+1)+i}$ for each $i = 1, 2, 3$ and $j = 0, \dots, t-1$. Since each layer L_{3j+i} contains at least one FOF, pick one of them. Call this FOF f . Now, pick one minterm m in f and set the variables from m appropriately so that m evaluates to true. This assignment to the variables satisfies f . Now, since L_{3j+i} and $L_{3(j+1)+i}$ are separated, we can do this assignment for all layers L_{3j+i} . Thus we have created an assignment to the variables that satisfies at least t FOF.

This gives the bound $r \leq 3k$ on the outerplanarity of the incidence graph when the maximum number of formulae satisfied is k . According to Proposition 4.1, the incidence graph has treewidth $9k - 1$ when the maximum weight for satisfying assignments is k . Using the associated tree decomposition in the standard dynamic programming algorithm gives the result. \square

For the above theorems and proofs, we have the following results concerning EPTAS.

Corollary 4.1 *All problems in the classes PLANAR TMIN_1 , Planar TMAX_1 , and PLANAR MPSAT_1 admit time $O(2^{O(1/\epsilon)n})$ EPTAS.*

Proof: As shown in the proof of Theorem 4.1, for each problem Π in PLANAR TMIN_1 , Π^* can be solved in time $O(2^t n)$ via the standard tree decomposition algorithmic schema, where t is the treewidth of the incidence graph of each collection C of FOFs. Therefore, by Lemma 4.1, there is an EPTAS for Π running in time

$$O\left(\frac{1}{\epsilon} 2^{O(1/\epsilon)n}\right) = O(2^{O(1/\epsilon)n}).$$

Similarly, for each problem Π in PLANAR TMAX_1 , Π^* can be solved in time $O(2^t n)$ steps, where t is the treewidth of the given incidence graph. So, by Lemma 4.1, there is an EPTAS for Π running in time $O(2^{O(1/\epsilon)n})$. The proof for PLANAR MPSAT_1 is similar. \square

Corollary 4.2 *All problems in the classes $\text{PLANAR TMIN}^{\text{polylog}}$, $\text{Planar TMAX}^{\text{polylog}}$, and $\text{PLANAR MPSAT}^{\text{polylog}}$ admit EPTAS.*

Proof: The proof employs Theorems 4.2, 4.3, and 4.4 and Lemma 4.1, and is similar to the proof of Corollary 4.1. \square

5 Conclusion

The problems that are the main focus of our investigation in this paper, PLANAR TMIN , PLANAR TMAX , and PLANAR MPSAT , were introduced by Khanna and Motwani as central to their ambitious program to address the question of, “What is it about the structure of an optimization

problem that allows it to have a PTAS?” They proposed that the answer is, roughly, hidden planar logical structure [KM96].

Such a program would not have to capture *all* PTASs in order to be considered successful. A skeptic might consider the program implausible on the grounds that “planar structure” is much too special to play such a general role. On the other hand, many structural and algorithmic results that hold for planar graphs actually generalize to any class of graphs that excludes a fixed graph H as a minor (see [AST90, DH04, DH05, DH06]). In this form, the original program of Khanna and Motwani must still be considered interesting and worthy of further investigation.

One can parameterize any given classical problem in a variety of interesting and potentially relevant ways (the parameter, in a parameterized problem, doesn’t have to be the solution size). In particular, parameterizing by the goodness of approximation, with $k = 1/\epsilon$, is one of the most fundamental ways to parameterize. A parameterized problem Π , parameterized in this way, is in the class XP (“solvable in polynomial time for every fixed parameter value”) if and only if the optimization problem has a PTAS, and it is fixed-parameter tractable if and only if the optimization problem has an EPTAS. The difference between a PTAS and an EPTAS is basic, since between the two classes of algorithms lies the distinction that animates the entire field of parameterized complexity. To put it differently, the largely unexplored question of what optimization problems admitting PTASs admit EPTASs, is a premiere application area for parameterized complexity theory. It is not necessary in this context to *introduce* a parameter, since the natural parameter $k = 1/\epsilon$ is already fully engaged.

As additional motivation for such explorations, of which there have been only a few so far (see the recent results in [Ma05a, Ma05b]), it has become clear that many of the PTASs developed in recent years are unlikely to be useful in practice because of the heavy dependence of the exponents of the polynomials bounding the running times of the PTASs on the parameter $k = 1/\epsilon$ [Fe02, Dow03, Fe03, Ma05a].

The centrality of the three planar logic optimization problems to the PTAS-explanatory program of Khanna and Motwani has motivated our attention to the “*PTAS versus EPTAS*” question for these problems. Our results have been both positive and negative. We have shown that none of the three proposed fundamental planar logic problems admit EPTASs unless $FPT = W[1]$.

We have also explored restrictions based on the maximum size of minterms. We have shown that if an optimization problem can be described by planar collections of FOFs with minterms of size 1, then it does have an EPTAS. But there is also a limit to such “good news” — we have shown that there are problems that can be described by a collection of FOFs with minterms of size at most 4 that do not have an EPTAS unless $FPT = W[1]$. This connection between the syntactic description of planar problems and the existence of EPTASs also deserves further investigation. For instance, what is the complexity of problems in PLANAR TMIN₂ and PLANAR TMIN₃?

References

- [ABFKN00] J. Alber, H. Bodlaender, H. Fernau, T. Kloks and R. Niedermeier, “Fixed Parameter Algorithms for Dominating Set and Related Problems on Planar Graphs,” *Algorithmica* 33 (2002), 461–493.
- [ADF95] K. Abrahamson, R. Downey and M. Fellows, “Fixed Parameter Tractability and Completeness IV: On Completeness for $W[P]$ and $PSPACE$ Analogs,” *Annals of Pure and Applied Logic* 73 (1995), 235–276.

- [Ar97] S. Arora, “Nearly Linear Time Approximation Schemes for Euclidean TSP and Other Geometric Problems,” *Proc. 38th Annual IEEE Symposium on the Foundations of Computing (FOCS’97)*, IEEE Press (1997), 554–563.
- [AST90] N. Alon, P. Seymour and R. Thomas, “A Separator Theorem for Nonplanar Graphs,” *J. Amer. Math. Soc.* 3 (1990), 801–808.
- [Bak94] B. Baker, “Approximation Algorithms for NP-Complete Problems on Planar Graphs,” *Journal of the ACM* 41 (1994), 153–180.
- [Baz95] C. Bazgan, “Schémas d’approximation et complexité paramétrée,” Rapport de stage de DEA d’Informatique à Orsay, 1995.
- [BFR98] R. Balasubramanian, M. Fellows and V. Raman, “An Improved Fixed-Parameter Algorithm for Vertex Cover,” *Information Processing Letters* 65 (1998), 163–168.
- [Bod88] H. Bodlaender, “Dynamic Programming on Graphs with Bounded Treewidth,” *Proceedings of ICALP 1988*, Springer-Verlag, *Lecture Notes in Computer Science* 317 (1988), 105–118.
- [Bod97] H. Bodlaender, “Treewidth: Algorithmic Techniques and Results,” *Proceedings of MFCS 1997*, Springer-Verlag, *Lecture Notes in Computer Science* 1295 (1997), 19–36.
- [CC97] L. Cai and J. Chen, “On Fixed-Parameter Tractability and Approximability of NP Optimization Problems,” *J. Computer and System Sciences* 54 (1997), 465–474.
- [CJ01] L. Cai and D. Juedes, “Subexponential Parameterized Algorithms Collapse the W-Hierarchy,” *Proceedings of ICALP 2001*, Springer-Verlag, *Lecture Notes in Computer Science* 2076 (2001), 273–284.
- [CT97] M. Cesati and L. Trevisan, “On the Efficiency of Polynomial Time Approximation Schemes,” *Information Processing Letters* 64 (1997), 165–171.
- [DF95a] R. Downey and M. Fellows, “Fixed-Parameterized Tractability and Completeness II: On Completeness for $W[1]$,” *Journal of Theoretical Computer Science* 141 (1995), 109–131.
- [DF95b] R. Downey and M. Fellows, “Parameterized computational feasibility,” In: P. Clote and J. Remmel (eds.): *Feasible Mathematics II*. Birkhäuser, Boston (1995), 219–244.
- [DF99] R. Downey and M. Fellows. *Parameterized Complexity*. Springer-Verlag, 1999.
- [DFS99] R. Downey, M. Fellows and U. Stege. Parameterized complexity: a framework for systematically confronting computational intractability. In: *Contemporary Trends in Discrete Mathematics* (R. Graham, J. Kratochvil, J. Nešetřil and F. Roberts, eds.), Proceedings of the DIMACS-DIMATIA Workshop on the Future of Discrete Mathematics, Prague, 1997, *AMS-DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, vol. 49 (1999), 49–99.
- [DH04] E. Demaine and M. Hajiaghayi, “Fast Algorithms for Hard Graph Problems: Bidimensionality, Minors and Local Treewidth,” *Proc. 12th International Symposium on Graph Drawing (GD 2004)*, Springer-Verlag, *Lecture Notes in Computer Science* ??? (2004), 517–533.

- [DH05] E. Demaine and M. Hajiaghayi, “Graphs Excluding a Fixed Minor have Grids as Large as Treewidth, with Combinatorial and Algorithmic Applications through Bidimensionality,” In: *Proc. 16th ACM-SIAM Symposium on Discrete Algorithms (SODA) 2005*, ACM Press (2005), 682–689.
- [DH06] E. Demaine and M. Hajiaghayi, “Linearity of Grid Minors in Treewidth with Applications through Bidimensionality,” to appear.
- [Dow03] R. G. Downey, “Parameterized Complexity for the Skeptic,” *Proc. 18th IEEE Annual Conf. on Computational Complexity*, IEEE Press (2003), 147–169.
- [Fe02] M. Fellows, “Parameterized Complexity: The Main Ideas and Connections to Practical Computing.” In *Experimental Algorithmics*, Springer-Verlag, *Lecture Notes in Computer Science* 2547 (2002), 51–77.
- [Fe03] M. Fellows, “New Directions and New Challenges in Algorithm Design and Complexity, Parameterized,” *Proc. of the Workshop on Algorithms and Data Structures (WADS 2003)*, Springer-Verlag, LNCS 2748 (2003), 505–520.
- [GGR96] O. Goldreich, S. Goldwasser and D. Ron, *Proceedings of the 37th IEEE Conf. on Foundations of Computer Science (FOCS)*, IEEE Press (1996), 339–348.
- [HMRRRS98] H. B. Hunt III, M. Marathe, V. Radhakrishnan, S. Ravi, D. Rosenkrantz and R. Stearns, “NC-Approximation Schemes for NP- and PSPACE-hard Problems on Geometric Graphs,” *Journal of Algorithms* 26 (1998), 238–274.
- [GJ79] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness*. W.H. Freeman, San Francisco, 1979.
- [KM96] S. Khanna and R. Motwani, “Towards a Syntactic Characterization of PTAS,” in: *Proc. STOC 1996*, ACM Press (1996), 329–337.
- [Ma05a] D. Marx, “Efficient Approximation Schemes for Geometric Structures?” *Proc. European Symposium on Algorithms* (2005), 448–459.
- [Ma05b] D. Marx, “The Closest Substring and Consensus Patterns problems with small distances,” *Proc. FOCS 2005*, IEEE Press (2005), 63–72.