On the optimality of planar and geometric approximation schemes

Dániel Marx
Institut für Informatik,
Humboldt-Universität zu Berlin,
dmarx@informatik.hu-berlin.de

Abstract

We show for several planar and geometric problems that the best known approximation schemes are essentially optimal with respect to the dependence on ϵ. For example, we show that the $2^{O(1/\epsilon)} \cdot n$ time approximation schemes for planar MAXIMUM INDEPENDENT SET and for TSP on a metric defined by a planar graph are essentially optimal: if there is a $\delta > 0$ such that any of these problems admits a $2^{O((1/\epsilon)^{1-\delta})}n^{O(1)}$ time PTAS, then the Exponential Time Hypothesis (ETH) fails. It is known that MAXIMUM INDEPENDENT SET on unit disk graphs and the planar logic problems MPSAT, TMIN, TMAX admit $n^{O(1/\epsilon)}$ time approximation schemes. We show that they are optimal in the sense that if there is a $\delta > 0$ such that any of these problems admits a $2^{(1/\epsilon)^{O(1)}}n^{O((1/\epsilon)^{1-\delta})}$ time PTAS, then ETH fails.

1 Introduction

Many classical graph-theoretic problems admit polynomial-time approximation schemes (PTAS) when restricted to planar graphs. For example, in planar graphs a $(1 + \epsilon)$-approximation for MAXIMUM INDEPENDENT SET, MINIMUM VERTEX COVER, MINIMUM DOMINATING SET [7], and TSP [19] can be computed in time $2^{O(1/\epsilon)} \cdot n$. Planar graph problems and planar geometric problems share some structural similarities, hence two-dimensional geometric problems often have PTAS's. For example, given a set of unit disks in the plane, a $(1 + \epsilon)$-approximation of the maximum independent set can be found in time $n^{O(1/\epsilon)}$ [15].

Improving the dependence of the running time on ϵ is an obvious goal and there is a history of such improvements in the literature. Arora [5] presented an $n^{O(1/\epsilon)}$ time PTAS for Euclidean TSP, which was improved to $n \cdot \log^{O(1/\epsilon)} n$ in the journal version of the paper [6]. Later, Rao and Smith [26] gave a PTAS with running time further improved to $2^{O((1/\epsilon) \log(1/\epsilon))}n + O(n \log n)$. Thus, given a PTAS, it

seems worth investigating whether improvements in the dependence on ϵ are possible or we have reached the optimum PTAS. For planar graphs, an exact solution of MAXIMUM INDEPENDENT SET can be found in time $2^{O(\sqrt{n})}$ [2] (instead of the trivial $2^{O(n)}$); this might raise the hope that a $2^{O(\sqrt{1/\epsilon})} \cdot n^{O(1)}$ PTAS exists. Finding a set of k independent unit disks can be done in time $n^{O(\sqrt{k})}$ [3] (instead of the trivial $n^{O(k)}$), hence an $n^{O(\sqrt{1/\epsilon})}$ time PTAS would not be completely surprising. The main contribution of the paper is proving almost-tight lower bounds for planar and geometric problems: we show that the known approximation schemes for these problems are essentially optimal with respect to the dependence on ϵ in the running time. To prove these lower bounds, we assume the Exponential Time Hypothesis (ETH): we assume that n-variable 3SAT cannot be solved in time $2^{o(n)}$. The Sparsification Lemma of Impagliazzo, Paturi, and Zane [16] states that this assumption is equivalent to the (seemingly stronger) assumption that there is no algorithm that solves 3SAT in $2^{o(n)}$, where n is the size of the instance. We remark that instead of assuming ETH, the results in this paper follow from the weaker assumption that m-clause 3SAT cannot be solved in time $2^{O(m^{1-\delta})}$ for any $\delta > 0$.

The following theorem states our lower bounds for five problems that (with the exception of the last) are known to admit approximation schemes with running time $2^{O(1/\epsilon)} \cdot n$.

Theorem 1.1. *Assuming ETH, there is no $\delta > 0$ such that a $2^{O((1/\epsilon)^{1-\delta})}n^{O(1)}$ time PTAS exists for*

- MAXIMUM INDEPENDENT SET *on planar graphs,*

- MINIMUM VERTEX COVER *on planar graphs,*

- MINIMUM DOMINATING SET *on planar graphs,*

- TSP *with a metric defined by an unweighted planar graph.*

- MINIMUM VERTEX COVER *for unit disks or unit squares.*

The problems considered in the following theorem admit approximation schemes with running time $n^{O(1/\epsilon)}$; we show that $1/\epsilon$ in the exponent of n cannot be replaced by $(1/\epsilon)^{1-\delta}$ for any $\delta > 0$. The statement of the theorem is slightly stronger: the lower bound on the exponent holds even if we allow an exponential function of $1/\epsilon$ as a multiplier:

Theorem 1.2. *Assuming ETH, there is no $\delta > 0$ such that a $2^{(1/\epsilon)^{O(1)}} \cdot n^{O((1/\epsilon)^{(1-\delta)})}$ time PTAS exists for*

- *MPSAT for planar formulas,*

- *TMIN for planar formulas,*

- *TMAX for planar formulas,*

- MAXIMUM INDEPENDENT SET *for unit disks or squares.*

- MINIMUM DOMINATING SET *for unit disks or squares.*

Ideally, for these problems we would like to have a tight result that rules out the possibility of a PTAS with running time $f(\epsilon)n^{o(1/\epsilon)}$, for *any* function $f(\epsilon)$. For example, Theorem 1.2 does not rule out the possibility of a PTAS with running time, say, $2^{2^{2^{1/\epsilon}}} n^{\log\log(1/\epsilon)}$—such a PTAS could be considered as a theoretical improvement over $n^{O(1/\epsilon)}$ time. Without going into details, we mention that it is possible to prove the weaker result that no $f(\epsilon)n^{o(\sqrt{1/\epsilon})}$ PTAS exists for any function $f(\epsilon)$ for these problems. Theorems 1.1 and 1.2 are not tight because we are using the almost-linear size PCP construction of Dinur [12]. Replacing this with a linear-size PCP would result in tight lower bounds, but currently such a construction is not in sight.

A simple observation gives us non-tight lower bounds on the dependence on ϵ. Since MAXIMUM INDEPENDENT SET is NP-hard for planar graphs, there is a polynomial-time algorithm that turns a 3SAT instance of size n into an equivalent instance of MAXIMUM INDEPENDENT SET on a planar graph of size n^c, for some constant c. By setting $\epsilon := 1/(n^c + 1)$, a $(1 + \epsilon)$-approximation algorithm can solve the constructed instance of MAXIMUM INDEPENDENT SET exactly. Therefore, no PTAS with running time $2^{o((1/\epsilon)^{1/c})}n^{O(1)}$ can exist: otherwise it would be able to solve 3SAT in time $2^{o(n)}$, contradicting ETH. By observing that $c = 2$ in the known reductions from 3SAT to planar MAXIMUM INDEPENDENT SET, we obtain a lower bound of $2^{\sqrt{1/\epsilon}} \cdot n^{O(1)}$. However, this argument cannot be used to prove stronger lower bounds: as planar MAXIMUM INDEPENDENT SET can be solved in time $2^{O(\sqrt{n})}$ [2], a reduction with $c < 2$ would mean that 3SAT can be solved in time $2^{o(n)}$. Therefore, new techniques are required for the almost-tight bounds of Theorems 1.1.

The situation is similar in the case of MAXIMUM INDEPENDENT SET for unit disks. The W[1]-hardness proof of [24] together with a result of [11] implies that there is a constant $c > 0$ such that there is no $f(k)n^{o(k^{1/c})}$ time algorithm for finding a set of k independent disks, unless ETH fails. This means (by a simple argument of Bazgan [8] and Cesati and Trevisan [10]) that no PTAS with running time $f(\epsilon)n^{o((1/\epsilon)^{1/c})}$ can exist. However, the problem can be solved in time $n^{O(\sqrt{k})}$ [3], thus $c \geq 2$, which means that we cannot prove a lower bound stronger than $f(\epsilon)n^{O(\sqrt{1/\epsilon})}$ with this argument.

We get around these difficulties by using the fact that, assuming ETH, not only 3SAT cannot be solved in time $2^{o(n)}$, but the optimization version MAX3SAT cannot be approximated to some constant factor in time $2^{O(n^{1-\delta})}$ for any $\delta > 0$. This is a consequence of the almost-linear size PCP of Dinur [12]. The lower bounds for the planar problems are obtained by a reduction that is approximation preserving in a weak way: a fast PTAS for the planar problem would imply a fast constant-factor approximation for MAX3SAT. Note that this reduction is somewhat unusual, since a problem without a PTAS is reduced to a problem that admits a PTAS.

Previously, the literature focused mostly on the PTAS vs. EPTAS question [10, 9, 13, 24, 23, 25]. An *efficient PTAS* (EPTAS) is an approximation scheme with running time $f(\epsilon)n^{O(1)}$ for some function f. These papers showed, for problems that were known to admit approximation schemes, that no EPTAS exists (under the standard parameterized complexity assumption W[1] $\neq$ FPT). These results can be turned into non-tight lower bounds on the exponent of n. In [4], almost-tight lower bounds were obtained for the string matching problem CLOSEST SUBSTRING; this is the only previous result that we are aware of where almost-tight bounds were established. Our result give almost-tight bounds for most of the problems considered in [10, 9, 13, 24]. Furthermore, we also give almost-tight lower bounds for problems that admit EPTASs (Theorem 1.1); to our knowledge, these are the first results of this type.

In Section 2, we introduce the planar problem MATRIX TILING. This problem admits an $n^{O(1/\epsilon)}$ time PTAS and a special case of the problem admits a $2^{O(1/\epsilon)}n$ time PTAS. We show that these approximation schemes are optimal in the sense of Theorem 1.1 and 1.2. The lower bounds of Theorem 1.1 and 1.2 are obtained by reducing MATRIX TILING (or its special case) to the various problems (Sections 3–6). MATRIX TILING was defined with such reductions in mind, thus the reductions are fairly straightforward if we can construct the problem-specific gadgets. It is likely that reduction from MATRIX TILING can be used to prove lower bounds for many other planar and geometric problems.

2 The MATRIX TILING problem

The problem MATRIX TILING plays a central role in the paper: we prove lower bounds on the efficiency of the approximation schemes of MATRIX TILING and these lower bounds are transferred to other problems by appropriate reductions. We denote by Z_D the set $\{0, 1, \ldots, D-1\}$ throughout the paper.

MATRIX TILING

Input:
Integers k, D, and k^2 nonempty sets $S_{i,j} \subseteq Z_D \times Z_D$ ($1 \leq i, j \leq k$).

Find:
For each $1 \leq i, j \leq k$, a value $s_{i,j} \in S_{i,j} \cup \{\bigstar\}$ such that

- If $s_{i,j} = (a_1, a_2)$ and $s_{i,j+1} = (b_1, b_2)$, then $a_1 = b_1$.
- If $s_{i,j} = (a_1, a_2)$ and $s_{i+1,j} = (b_1, b_2)$, then $a_2 = b_2$.

Goal:
Maximize the number of pairs (i, j) ($1 \leq i, j \leq k$) with $s_{i,j} \neq \bigstar$.

We think of the values $s_{i,j}$ of a solution as being elements of a matrix, hence we use the expressions row, column, and cell with the obvious meaning. Observe that the optimum is always at least $k^2/4$: if i and j are both odd, then let $s_{i,j}$ be an arbitrary element of $S_{i,j}$, otherwise let $s_{i,j} = \bigstar$. (Here we use that $S_{i,j}$ is required to be nonempty.) This observation will be useful when reducing MATRIX TILING to other problems. The size of the input instance can be bounded by $O(k^2 D^2)$.

The following lemma gives a reduction from MAX3SAT to MATRIX TILING. The reduction is approximation-preserving in a certain weak sense, which allows use to use a PTAS for MATRIX TILING to solve MAX3SAT. A formula with m clauses is α-*satisfiable* if there is an assignment that satisfies at least αm clauses.

Lemma 2.1. *There is an algorithm with running time polynomial in the size of the* output *that, given a 3SAT formula ϕ having m clauses and an integer k, constructs an instance of* MATRIX TILING *with parameters k and $D := 3^{\lceil m/k \rceil}$ such that for every $1 > \alpha > 0$,*

- *if ϕ is satisfiable, then the optimum of* MATRIX TILING *is k^2,*

- *if ϕ is not α-satisfiable, then the optimum of* MATRIX TILING *is at most $k^2 - k(1-\alpha)/2 + 1$.*

Proof. Let us associate a variable $y_i \in \{1, 2, 3\}$ with each clause of ϕ. The intended meaning of $y_i = \ell$ is that the i-th clause is satisfied by its ℓ-th literal. To enforce this interpretation, we construct a set of constraints: if the ℓ_1-th literal of the i_1-th clause is the negation of the ℓ_2-th literal of the i_2-th

clause, then the constraint $(y_{i_1} \neq \ell_1 \vee y_{i_2} \neq \ell_2)$ is added to the set. It is clear that a subset C_0 of clauses is satisfiable if and only if there is an assignment to the corresponding $|C_0|$ variables that satisfies all the constraints induced by C_0.

We want to partition the y_i's into k blocks $B_1, \ldots, B_k$ of equal size, but m/k is not necessarily integer. Let $m' := k\lceil m/k \rceil$ and let us add $m' - m \leq m'/k$ new dummy variables that do not appear in any of the constraints. Set $D := 3^{m'/k}$; for each block of m'/k variables, we can fix a bijection between Z_D and the possible assignments of the variables in the block. We construct an instance of MATRIX TILING with parameters k and D where the sets $S_{i,j}$ are defined as follows. We say that a partial assignment γ of the variables y_i is *compatible,* if there is no constraint $(y_{i_1} \neq \ell_1 \vee y_{i_2} \neq \ell_2)$ with $\gamma(y_{i_1}) = \ell_1$ and $\gamma(y_{i_2}) = \ell_2$. If $i = j$, then $S_{i,j}$ contains those pairs (t, t) where $t \in Z_D$ corresponds to a compatible assignment of block B_i. For $i \neq j$, consider a pair $(v_i, v_j) \in Z_D \times Z_D$. Let γ_i (resp., γ_j) be the assignment of block B_i (resp., B_j) that corresponds to v_i (resp., v_j). The pair (v_i, v_j) is in $S_{i,j}$ if and only if γ_i and γ_j together form a compatible assignment of $B_i \cup B_j$. If a set $S_{i,j}$ is empty, then ϕ is unsatisfiable, since there is no compatible assignment for some pair of blocks. In this case, we can output an arbitrary instance with optimum $k^2/4$. This completes the description of the constructed instance of MATRIX TILING.

If ϕ is satisfiable, then there is an assignment γ of the y_i's that satisfies all the constraints we defined. For each $1 \leq i, j \leq k$, let $v_i \in Z_D$ be the value corresponding to assignment γ restricted to block B_i. Set $s_{i,j} = (v_i, v_j)$. Assignment γ satisfies all the constraints; in particular, it satisfies all the constraints induced by the variables in $B_i \cup B_j$, implying $(v_i, v_j) \in S_{i,j}$. Thus we obtain a solution with value k^2.

For the second part, suppose that MATRIX TILING has a solution with value at least $k^2 - k(1-\alpha)/2 + 1$, i.e., the number of $\bigstar$'s is at most $k(1-\alpha)/2 - 1$. Let us call a block B_i *bad* if the i-th row or the i-th column contains at least one $\bigstar$. As a $\bigstar$ can make at most 2 blocks bad, there are at most $(1-\alpha)k - 2$ bad blocks. Let us call a variable y_t and (if y_t is not a dummy variable) the corresponding clause of ϕ *good,* if y_t does not belong to a bad block; clearly, there are at least $(m'/k)(k - (1-\alpha)k + 2) = \alpha m' + 2m'/k$ good variables. Less than m'/k of the good variables are dummy variables, hence we have at least αm good clauses. We claim that there is an assignment of ϕ satisfying all the good clauses, contradicting the assumption that ϕ is not α-satisfiable. If block B_r is not bad, then there is a value v_r such that the first component of each $s_{r,j}$ ($1 \leq j \leq k$) is v_r. Similarly, there is a value v'_r such that the second component of each $s_{i,r}$ ($1 \leq i \leq k$) is v'_r. Since $(v_r, v'_r) \in S_{r,r}$, we get that $v_r = v'_r$. The value v_r defines an assignment of the variables y_t in block v_r; this way, we obtain a value for

each good variable y_t. In a natural way, we can construct an assignment of ϕ corresponding to the values of the good variables: $y_t = \ell$ means that the t-th clause is satisfied by its ℓ-th literal, hence it defines the value of one variable of ϕ; if a variable x_i of ϕ is not assigned a value this way, then we can assign an arbitrary value to it. This assignment is well defined: if $y_{t_1} = \ell_1$ and $y_{t_2} = \ell_2$ force a variable to different values, then there is a constraint $(y_{t_1} \neq \ell_1 \vee y_{t_2} \neq \ell_2)$. Suppose that $y_{t_1} \in B_{r_1}$, $y_{t_2} \in B_{r_2}$; this means that the assignment of B_{r_1} corresponding to v_{r_1} is not compatible with the assignment of B_{r_2} corresponding to v_{r_2}, contradicting the fact that $s_{r_1,r_2} = (v_{r_1}, v_{r_2}) \in S_{r_1,r_2}$. Therefore, the constructed assignment satisfies every good clause, i.e., ϕ is α-satisfiable. □

Using standard layering techniques, it can be shown that MATRIX TILING admits a PTAS with running time $n^{O(1/\epsilon)}$. In Theorem 2.3, we show that the $1/\epsilon$ in the exponent of n is essentially optimal. The lower bound is obtained from known hardness results on MAX3SAT.

The Sparsification Lemma of Impagliazzo, Paturi, and Zane [16] implies that ETH is equivalent to the assumption that m-clause 3SAT cannot be solved in time $2^{o(m)}$. The almost-linear size PCP of Dinur can be used to turn a formula ϕ with m clauses into a formula ψ with $m' := m \log^{O(1)} m$ clauses such that if ϕ is satisfiable, then ψ is satisfiable and if ϕ is unsatisfiable, then ψ is not α-satisfiable for some constant $\alpha < 1$. Therefore, the running time of an algorithm distinguishing between these two cases cannot be $2^{o(m)}$, which means that it cannot be $2^{O(m'^{(1-\delta)})}$ for any $\delta > 0$.

Lemma 2.2. *There is a constant $1 > \alpha > 0$ such that if there is an algorithm that can distinguish between satisfiable and not α-satisfiable 3SAT formulas in time $2^{O(m^{1-\delta})}$ for some constant $\delta > 0$ (where m is the number of clauses), then ETH fails.*

Theorem 2.3. *If there are constants $\delta, d > 0$ such that MATRIX TILING has a PTAS with running time $2^{O(1/\epsilon)^d} \cdot n^{O(1/\epsilon)^{1-\delta}}$, then ETH fails.*

Proof. Let ϕ be a 3SAT formula with m clauses. Set $k := \lceil m^{1/(2d+1)} \rceil \leq m$. Let us use the algorithm of Lemma 2.1 to construct an instance of MATRIX TILING from ϕ with this value of k. Set $\epsilon := (1-\alpha)/(2k) - 1/k^2 = \Theta(1/k)$, where α is the universal constant in Theorem 2.2 (we assume that m, and hence k, is sufficiently large that ϵ is positive.) If ϕ is satisfiable, then the optimum of MATRIX TILING is k^2. On the other hand, if ϕ is not α-satisfiable, then the optimum is at most $k^2 - k(1-\alpha)/2 + 1$. Since $k^2/(1+\epsilon) > k^2(1-\epsilon) = k^2 - k(1-\alpha)/2 + 1$, a $(1+\epsilon)$-approximation algorithm can distinguish between the two cases. If there is a PTAS with running time

$2^{O(1/\epsilon)^d} \cdot n^{O(1/\epsilon)^{(1-\delta)}}$, then the two cases can be distinguished in time

$$2^{O(k^d)} \cdot (k^2 \cdot D^2)^{O(k)^{1-\delta}}$$
$$= \exp\left(O(k^d) + (O(\log k) + O(m/k)) \cdot O(k)^{1-\delta}\right)$$
$$= \exp\left(O(m^{d/(2d+1)}) + o(k) + O(m/k^\delta)\right)$$
$$= \exp\left(o(m) + O(m^{1-\delta/(2d+1)})\right) = 2^{o(m)},$$

which, by Lemma 2.2, contradicts ETH. □

Let us investigate the special case of MATRIX TILING with $D = 2$. It can be shown that this special case admits a PTAS with running time $2^{O(1/\epsilon)} n$ and, as we shall see in Theorem 2.6, this is essentially optimal. The lower bounds in Theorem 1.1 are obtained by reductions from this special case. When reducing from this special case, $|S_{i,j}| \leq 4$ implies that a "gadget" representing some $S_{i,j}$ has to have only a constant number of states. Therefore, the gadget construction can be simpler and the reduction can be done to a wider range of problems.

If $D = 2$, then we are not able to reduce MAX3SAT to MATRIX TILING as in Lemma 2.1. Instead, we reduce MAX2SAT. We say that a 2SAT formula is *simple,* if it does not contain unsatisfiable clauses or duplicated clauses.

Lemma 2.4. *There is a polynomial-time algorithm that, given a simple 2SAT formula ϕ with m clauses where each variables appears in at most d clauses, constructs an instance of MATRIX TILING with parameters $k := O(d^2m)$ and $D := 2$ such that if t is the minimum number of unsatisfied clauses in ϕ, then the optimum of the constructed instance is $k^2 - t$.*

Proof. Let $x_1, \ldots, x_p$ be an ordering of the variables of ϕ ($p \leq 2m$). By adding new variables, we obtain a longer sequence X of variables. Let $x_{i_1}, \ldots, x_{i_{d'}}$ be those variables that appear together with x_i in some clause ($d' \leq d$). Set $z := 4d$. We replace x_i with a sequence of $2zd' + 1$ variables, called the *segment* of x_i. This segment contains x_i and $2zd'$ new variables $x_{i_s,i,\ell}$ ($1 \leq s \leq d', 1 \leq \ell \leq 2z$). The variables in the segment are ordered in such a way that $x_{i_s,i,\ell}$ is before x_i for $1 \leq \ell \leq z$ and it is after x_i for $z + 1 \leq \ell \leq 2z$. Replacing every x_i with the corresponding segment of $2zd' + 1$ variables gives a sequence X of $k \leq p(2zd + 1) = O(d^2m)$ variables. For each new variable $x_{i_s,i,\ell}$, we say that it is a *copy* of x_{i_s}; each variable has at most $2zd$ copies (we do not consider a variable x_i to be a copy of itself). If a variable is a copy of x_i or it is x_i itself, then we say that the variable *represents* x_i. We construct an instance of MATRIX TILING with parameters k and $D = 2$ where the rows and the columns are indexed by the k variables in X. The sets $S_{i,j}$ are defined the following way. If there is a clause $(x_i \vee x_j)$, then the set in

row x_i and column x_j is $\{(0,1),(1,0),(1,1)\}$. We consider a clause as an ordered pair of literals, hence clause $(x_i \vee x_j)$ does not influence the set in row x_j of column x_i. We proceed similarly for clauses containing negations: in this case, the set $S_{i,j}$ contains the three pairs corresponding to the satisfying assignments of the clause. Every other set is $\{(0,0),(0,1),(1,0),(1,1)\}$, unless the row and the column indices represent the same variable x_i, in which case the set is $\{(0,0),(1,1)\}$.

Assume that ϕ has an assignment γ that satisfies all but t of the clauses. We define the value of $s_{i,j}$ the following way. If the i-th variable in sequence X represents x_{i_0} and the j-th variable represents x_{j_0}, then $s_{i,j} = (\gamma(x_{i_0}), \gamma(x_{j_0}))$. This pair is in $S_{i,j}$ unless $S_{i,j}$ corresponds to a clause not satisfied by γ. That is, the i-th (resp., j-th) variable of the sequence is x_{i_0} (resp., x_{j_0}) itself (not a copy) and there is an unsatisfied clause where the first (resp., second) variable is x_{i_0} (resp., x_{j_0}). If the pair is not in $S_{i,j}$, then we set $s_{i,j} = \bigstar$; clearly, the number of $\bigstar$'s is at most t.

Assume now that there is a solution with value at least $k^2 - t$. A variable of the sequence X is *bad* if the row or the column corresponding to the variable contains at least one $\bigstar$; otherwise let us call it *good*. If the i-th variable of the sequence X is good, then let us associate to this variable the value that appears as the first component of each pair in the i-th row. Because of the way the set $S_{i,i}$ is defined, this value is the same as the value that appears as the second component of each pair in the i-th column. If the i-th and the j-th variables of X are both good and they represent the same variable, then the same value is associated to them: set $S_{i,j} = \{(0,0),(1,1)\}$ ensures this. A variable of ϕ is *spoiled* if at least z of its copies are bad. We construct an assignment γ of ϕ the following way: if x_i is spoiled, then we set $\gamma(x_i)$ arbitrarily, otherwise let $\gamma(x_i)$ be the common value associated with the good copies of x_i.

We claim that at most t clauses of ϕ are not satisfied by γ. For each clause, we define a set of cells called the *cross* of the clause. Let x_i, x_j be the two variables of the clause. The cross of the clause contains those cells of row x_i whose column belongs to the segment of x_j and contains those cells of column x_j whose row belongs to the segment of x_i. It is easy to see that the crosses for different clauses are disjoint (here we use the assumption that there are no duplicated clauses in ϕ). We claim that for every unsatisfied clause, at least one of the following is true: (1) the cell in row x_i of column x_j contains a $\bigstar$, (2) the clause has a spoiled variable, (3) the cross of the clause contains at least two $\bigstar$'s. Consider a clause such that none of (1)–(3) hold. Since x_i is not spoiled, there is an $1 \le \ell_{i,1} \le z$ such that variable $x_{i,j,\ell_{i,1}}$ is good and there is a $z + 1 \le \ell_{i,2} \le 2z$ such that variable $x_{i,j,\ell_{i,2}}$ is good. If there is no $\bigstar$ in row x_i between columns $x_{i,j,\ell_{i,1}}$ and x_j, then the first component of row x_i of column x_j is $\gamma(x_i)$. Similarly, if there is no $\bigstar$ in row x_i

between columns x_i and $x_{i,j,\ell_{i,2}}$, then the first component of row x_i of column x_j is again $\gamma(x_i)$. As neither (1) nor (3) holds for the clause, at least one of these two statements has to be true. A similar argument shows that the second component of row x_i of column x_j is $\gamma(x_j)$. Therefore, the fact that the pair $(\gamma(x_i), \gamma(x_j))$ appears in the corresponding cell implies that the clause is satisfied by γ.

Let t_0 be the number of clauses for which (1) is true. Observe that the corresponding t_0 $\bigstar$'s do not influence whether (2) or (3) are true for the other clauses. Therefore, it is sufficient to investigate the effect of the remaining $t - t_0$ $\bigstar$'s. A $\bigstar$ can make at most 2 variables of the sequence X bad, hence there are at most $2(t - t_0)/z$ spoiled variables in ϕ. Therefore, (2) is true for at most $d \cdot 2(t - t_0)/z \le (t - t_0)/2$ clauses. As the crosses are disjoint, at most $(t - t_0)/2$ of them can contain at least two $\bigstar$'s. Therefore, the total number of unsatisfied clauses is at most $t_0 + (t - t_0)/2 + (t - t_0)/2 = t$. $\qquad\square$

Combining Lemma 2.2 with standard reductions, we get

Lemma 2.5. *There are constants $1 > \alpha_1 > \alpha_2 > 0$ such that if there is an algorithm that can distinguish between α_1-satisfiable and not α_2-satisfiable 2SAT formulas in time $2^{O(m^{1-\delta})}$ for some constant $\delta > 0$ (where m is the number of clauses), then ETH fails. Furthermore, we can assume that the formula is simple and a variable appears at most d times for some constant $d > 0$.*

Theorem 2.6. *If there is a $\delta > 0$ such that MATRIX TILING has a PTAS with running time $2^{O(1/\epsilon)^{1-\delta}} \cdot n^{O(1)}$ in the special case $D = 2$, then ETH fails.*

Proof. Let ϕ be a 3SAT formula with m clauses. Let us use the algorithm of Lemma 2.4 to construct an instance of MATRIX TILING with $k = O(d^2 m)$ and $D = 2$. If ϕ is α_1-satisfiable, then the optimum of the constructed instance of MATRIX TILING is at least $k^2 - (1 - \alpha_1)m$, while if ϕ is not α_2-satisfiable, then the optimum is less than $k^2 - (1 - \alpha_2)m$. This means that by setting $\epsilon := (\alpha_1 - \alpha_2)m/(4k^2) = O(1/k)$, a $(1 + \epsilon)$-approximation algorithm can distinguish between the two cases. Therefore, if the assumed PTAS exists, then this can be done in time

$$2^{O(1/\epsilon)^{1-\delta}} \cdot n^{O(1)} = 2^{O(k)^{1-\delta}} \cdot k^{O(1)} = 2^{O(m)^{1-\delta} + O(\log k)}$$
$$= 2^{O(m^{1-\delta})},$$

which, by Lemma 2.5, contradicts ETH. $\qquad\square$

Having proved the lower bounds of Theorems 2.3 and 2.6, we transfer these bounds to other optimization problems by means of an L-reduction.

Definition 2.7. *Let A and B be optimization problems and c_A and c_B their respective cost functions. A pair of*

logspace-computable functions R and S is an L-*reduction if all of the following conditions are met:*

- *if x is an instance of problem A, then $R(x)$ is an instance of problem B,*

- *if y is a solution to $R(x)$, then $S(y)$ is a solution to x,*

- *there exists a constant $\alpha > 0$ such that $\mathrm{OPT}(R(x)) \leq \alpha \mathrm{OPT}(x)$.*

- *there exists a constant $\beta > 0$ such that $|\mathrm{OPT}(x) - c_A(S(y))| \leq \beta |\mathrm{OPT}(R(x)) - c_B(y)|$.*

It is easy to see that the bounds of Theorems 2.3 and 2.6 remain valid under L-reductions:

Lemma 2.8. *(1) If there is an L-reduction from* MATRIX TILING *to Problem X, then there are no $d, \delta > 0$ such that Problem X admits a PTAS with running time $2^{O((1/\epsilon)^d)}n^{O((1/\epsilon)^{1-\delta})}$, unless ETH fails.*

(2) If there is an L-reduction from MATRIX TILING *with $D = 2$ to Problem X, then there is no $\delta > 0$ such that Problem X admits a PTAS with running time $2^{O((1/\epsilon)^{1-\delta})}n^{O(1)}$, unless ETH fails.* $\square$

3 Planar logic problems

Khanna and Motwani [18] defined classes of optimization problems that admit polynomial-time approximation schemes. The problems are formulated using Boolean logical expressions. A formula in *disjunctive normal form* (DNF) is a disjunction of terms. A DNF is *positive* (resp., *negative*), if every literal is positive (resp., negated). The *weight* of an assignment is the number of variables that are set to true.

MPSAT
Input: A collection $\mathcal{C} = \{\phi_1, \ldots, \phi_n\}$ of DNFs.
Find: An assignment γ.
Goal: Maximize the number of DNFs satisfied by γ.

TMIN
Input: A collection $\mathcal{C} = \{\phi_1, \ldots, \phi_n\}$ of positive DNFs.
Find: An assignment γ that satisfies every DNF in $\mathcal{C}$.
Goal: Minimize the weight of γ.

TMAX
Input: A collection $\mathcal{C} = \{\phi_1, \ldots, \phi_n\}$ of negative DNFs.
Find: An assignment γ that satisfies every DNF in $\mathcal{C}$.
Goal: Maximize the weight of γ.

These problems generalize many of the standard optimization problems: for example, MAX CUT can be reduced to MPSAT; MAXIMUM INDEPENDENT SET can be reduced to TMAX; and MINIMUM VERTEX COVER can be reduced to TMIN. (In all three reductions, we associate a
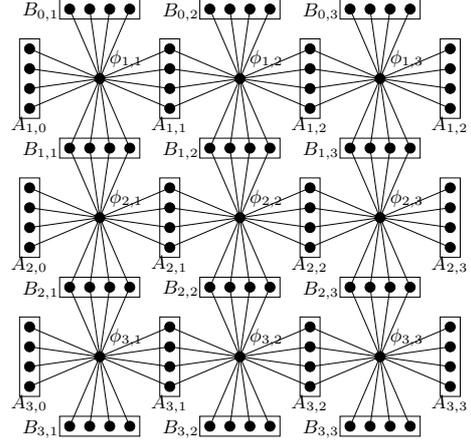
Figure 1. Structure of the instance constructed in Theorem 3.1 ($D = 4$, $k = 3$).

variable with each vertex and a DNF with each edge.) Given an instance of the above problems, the *incidence graph* is a bipartite graph defined by associating a vertex to each variable and to each DNF, and by connecting each variable to every DNF where it appears. Khanna and Motwani [17] show that MPSAT, TMIN, TMAX all admit $n^{O(1/\epsilon)}$ time PTAS's if the incidence graph is planar. Here we prove that these approximation schemes are essentially optimal:

Theorem 3.1. *If there is a $\delta > 0$ such that* PLANAR MP-SAT *has a PTAS with running time $2^{(1/\epsilon)^{O(1)}} \cdot n^{O(1/\epsilon)^{1-\delta}}$, then ETH fails.*

Proof. We present an L-reduction from MATRIX TILING to PLANAR MPSAT. The collection $\mathcal{C}$ consists of k^2 DNFs $\phi_{i,j}$ ($1 \leq i, j \leq k$) on $2k(k+1)D$ variables. The variables are arranged into blocks $A_{i,j}$ ($1 \leq i \leq k$, $0 \leq j \leq k$) and $B_{i,j}$ ($0 \leq i \leq k$, $1 \leq j \leq k$), where each block contains D variables. The variables in block $A_{i,j}$ (resp., $B_{i,j}$) will be denoted by $a_{i,j,s}$ (resp., $b_{i,j,s}$) for $s \in Z_D$. The DNF $\phi_{i,j}$ ($1 \leq i, j \leq t$) contains variables only from blocks $A_{i,j-1}$, $A_{i,j}$, $B_{i-1,j}$, $B_{i,j}$. As shown in Figure 1, the incidence graph is planar. The formulas are defined as

$$\phi_{i,j} = \bigvee_{(x,y) \in S_{i,j}} (a_{i,j-1,x} \wedge a_{i,j,x} \wedge b_{i-1,j,y} \wedge b_{i,j,y}$$
$$\wedge \bigwedge_{x' \neq x} \bar{a}_{i,j-1,x'} \wedge \bigwedge_{x' \neq x} \bar{a}_{i,j,x'} \wedge \bigwedge_{y' \neq y} \bar{b}_{i-1,j,y'} \wedge \bigwedge_{y' \neq y} \bar{b}_{i,j,y'}).$$

This completes the description of the instance. We claim that the optimum of the constructed instance is the same as the optimum of MATRIX TILING. Assume that MATRIX TILING has a solution where the number of $\bigstar$'s is t. If $s_{i,j} = (x, y) \neq \bigstar$, then set the variables in $A_{i,j-1}$, $A_{i,j}$,

$B_{i-1,j}$, $B_{i,j}$ such that the term corresponding to (x,y) is satisfied in $\phi_{i,j}$, i.e., $a_{i,j-1,x'} = a_{i,j,x'}$ is true if and only if $x' = x$, and $b_{i-1,j,y'} = b_{i,j,y'}$ is true if and only if $y = y'$. Observe that this assignment is well-defined: for example, if $s_{i,j}, s_{i,j+1} \neq \bigstar$, then they assign the same values to the variables in $A_{i,j}$ (since $s_{i,j}, s_{i,j+1}$ agree in the first component). Assign values to the remaining variables arbitrarily. It is clear that if $s_{i,j} \neq \bigstar$, then the corresponding $\phi_{i,j}$ is satisfied.

For the other direction, assume that t of the $\phi_{i,j}$'s are satisfied in an assignment. If $\phi_{i,j}$ is satisfied, then there is a pair $(x,y) \in S_{i,j}$ such that the term corresponding to (x,y) is satisfied; set $s_{i,j} = (x,y)$ in this case. Let $s_{i,j} = \bigstar$ if $\phi_{i,j}$ is not satisfied. It is easy to verify that the $s_{i,j}$'s form a valid solution of MATRIX TILING. □

In a very similar way (details omitted), we can reduce MATRIX TILING to TMIN and TMAX, hence

Theorem 3.2. *If there is a $\delta > 0$ such that* PLANAR TMIN *or* PLANAR TMAX *has a PTAS with running time* $2^{(1/\epsilon)^{O(1)}} \cdot n^{O(1/\epsilon)^{1-\delta}}$*, then ETH fails.* □

4 Intersection graphs

Given a set of geometric objects in the plane, the *intersection graph* has one vertex for each object, and two vertices are connected by an edge if and only if they have nonempty intersection. The intersection graphs of squares, rectangles, disks, segments, and other geometric objects play an important role in many applications such as facility location [28], frequency assignment [22], and map labeling [1]. The MAXIMUM INDEPENDENT SET problem for the intersection graph of unit squares has a $2^{O(1/\epsilon)}n$ time PTAS [15]. Here we show that this PTAS is essentially optimal:

Theorem 4.1. *If there is a $\delta > 0$ such that* MAXIMUM INDEPENDENT SET *for unit squares admits a PTAS with running time* $2^{(1/\epsilon)^{O(1)}} \cdot n^{O(1/\epsilon)^{1-\delta}}$*, then ETH fails.*

Proof. The proof is by an L-reduction from MATRIX TILING. We assume that the squares are open, i.e., two squares that only touch each other do not intersect. We represent each $S_{i,j}$ by a gadget $G_{i,j}$. A gadget consists of 16 blocks of squares, where each block induces a clique. Consider the 16 squares $A_1, \ldots, A_{16}$ of Fig. 2a, and let $(x_1,y_1), \ldots, (x_{16},y_{16})$ be the coordinates of the lower left corner of these squares. (These squares are not part of the set of squares that we construct, but they will be useful in the definition of the set.) Set $\epsilon := 1/(4D^2)$ and let $\iota : Z_D \times Z_D \to Z_{D^2}$ be an arbitrary bijection. If $(x,y) \in S_{i,j}$ and $\iota(x,y) = r$, then we add 16 squares $B_{i,r}$ ($1 \leq i \leq 16$) to gadget $G_{i,j}$ with coordinates

(a)   (b)
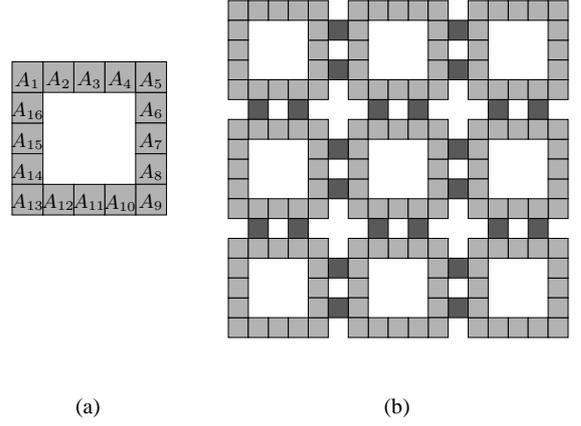
Figure 2. The gadget and the way the gadgets are arranged for $k = 3$ (Theorem 4.1).

$$
\begin{array}{ll}
(x_1 + r\epsilon, y_1 + r\epsilon) & (x_9 - r\epsilon, y_9 - r\epsilon) \\
(x_2 + r\epsilon, y_2 - y\epsilon) & (x_{10} - r\epsilon, y_{10} + y\epsilon) \\
(x_3 + r\epsilon, y_3 - D\epsilon) & (x_{11} - r\epsilon, y_{11} + D\epsilon) \\
(x_4 + r\epsilon, y_4 + y\epsilon) & (x_{12} - r\epsilon, y_{12} - y\epsilon) \\
(x_5 + r\epsilon, y_5 - r\epsilon) & (x_{13} - r\epsilon, y_{13} + r\epsilon) \\
(x_6 - x\epsilon, y_6 - r\epsilon) & (x_{14} + x\epsilon, y_{14} + r\epsilon) \\
(x_7 - D\epsilon, y_7 - r\epsilon) & (x_{15} + D\epsilon, y_{15} + r\epsilon) \\
(x_8 + x\epsilon, y_8 - r\epsilon) & (x_{16} - x\epsilon, y_{16} + r\epsilon)
\end{array}
$$

Furthermore, we add 15 (!) squares $B_{1,D^2}, \ldots, B_{15,D^2}$ with the following coordinates:

$$
\begin{array}{ll}
(x_1 + D^2\epsilon, y_1 + D^2\epsilon) & (x_9 - D^2\epsilon, y_9 - D^2\epsilon) \\
(x_2 + D^2\epsilon, y_2 - D\epsilon) & (x_{10} - D^2\epsilon, y_{10} + D\epsilon) \\
(x_3 + D^2\epsilon, y_3 - D\epsilon) & (x_{11} - D^2\epsilon, y_{11} + D\epsilon) \\
(x_4 + D^2\epsilon, y_4 - D\epsilon) & (x_{12} - D^2\epsilon, y_{12} + D\epsilon) \\
(x_5 + D^2\epsilon, y_5 - D^2\epsilon) & (x_{13} - D^2\epsilon, y_{13} + D^2\epsilon) \\
(x_6 - D\epsilon, y_6 - D^2\epsilon) & (x_{14} + D\epsilon, y_{14} + D^2\epsilon) \\
(x_7 - D\epsilon, y_7 - D^2\epsilon) & (x_{15} + D\epsilon, y_{15} + D^2\epsilon) \\
(x_8 - D\epsilon, y_8 - D^2\epsilon)
\end{array}
$$

It is clear that at most 16 independent squares can be selected from a gadget and an independent set of size 16 contains exactly one square B_{i,v_i} for each $1 \leq i \leq 16$. It can be verified that for every $0 \leq j \leq D^2$, the squares $B_{1,j}$, $\ldots$, $B_{16,j}$ are independent (if they exist). Furthermore, we show that every independent set I of size 16 is of this form. Observe that if $v_i > v_{i+1}$, then B_{i,v_i} and $B_{i+1,v_{i+1}}$ intersect (and similarly for $B_{16,v_{16}}$ and B_{1,v_1}). Therefore, we get $v_1 \leq v_2 \leq \cdots \leq v_{16} \leq v_1$, which means that there is a j such that $v_i = j$ for every $1 \leq i \leq 16$. In particular, this means that if I is an independent set of size 16, then a square B_{i,D^2} cannot appear in I, as there is no square B_{16,D^2}.

The k^2 gadgets are arranged as in Fig. 2b, i.e., there is an offset of 6 between the rows and columns. Adjacent gad-

gets are connected by two blocks of additional squares as follows. If $j < k$, then in gadget $G_{i,j}$ we add the squares with coordinates $H_r^1 = (x_6 + 1 - r\epsilon, y_6 - D^2\epsilon)$, $H_r^2 = (x_8 + 1 + r\epsilon, y_8 + D^2\epsilon)$ for $0 \leq r < D$. These squares form the *horizontal connection* between $G_{i,j}$ and $G_{i,j+1}$. Similarly, if $i < k$, then we add $V_r^1 = (x_{10} - D^2\epsilon, y_{10} - 1 + r\epsilon)$, $V_r^2 = (x_{12} + D^2\epsilon, y_{12} - 1 - r\epsilon)$ for $0 \leq r < D$ (the *vertical connection* between $G_{i,j}$ and $G_{i+1,j}$). This completes the description of the constructed set of squares.

We claim that if the optimum of MATRIX TILING is $k^2 - t$, then the maximum number of independent squares is $16k^2 + 4k(k-1) - t$. Recall that the optimum of MATRIX TILING is always at least $k^2/4$, hence the reduction increases the optimum by at most a factor of $\alpha = 80$. Assume that MATRIX TILING has a solution with value $k^2 - t$. If $s_{i,j} = (x,y) \neq \bigstar$, then select the 16 squares $B_{1,\iota(x,y)}$, $\ldots$, $B_{16,\iota(x,y)}$ from gadget $G_{i,j}$. If $s_{i,j} = \bigstar$, then select the 15 squares $B_{1,D^2}, \ldots, B_{15,D^2}$ from $G_{i,j}$. Note that the $16k^2 - t$ squares selected so far are independent. We show that this independent set can be extended by $4k(k-1)$ further squares by selecting two squares from each of the $2k(k-1)$ connections. If $s_{i,j} = \bigstar$, then the 15 squares selected from $G_{i,j}$ do not intersect *any* of the squares in the horizontal or vertical connections of $G_{i,j}$. Therefore, if $s_{i,j}$ or $s_{i,j+1}$ is $\bigstar$, then it is easy to select two squares from the connection between $G_{i,j}$ and $G_{i,j+1}$. If $s_{i,j} = (x,y)$ and $s_{i,j+1} = (x,y')$, then we select H_x^1 and H_x^2 from the connection between $G_{i,j}$ and $G_{i,j+1}$. Observe that these two squares do not conflict with squares $B_{6,\iota(x,y)}$ and $B_{8,\iota(x,y)}$ of $G_{i,j}$ and with squares $B_{14,\iota(x,y)}$ and $B_{16,\iota(x,y)}$ of $G_{i,j+1}$. For example, $B_{6,\iota(x,y)}$ at $(x_6 - x\epsilon, y_6 - \iota(x,y)\epsilon)$ does not conflict with H_x^1 at $(x_6 + 1 - x\epsilon, y_6 - D^2\epsilon)$, since the difference in the first coordinate is exactly 1. In a similar way, we can select two squares from each vertical connection, hence we obtain an independent set of size $16k^2 + 4k(k-1) - t$, as required.

Assume now that we have an independent set I of size $16k^2 + 4k(k-1) - t$. If $|G_{i,j} \cap I| < 16$, then set $s_{i,j} = \bigstar$. We set $s_{i,j}$ to $\bigstar$ also in the case if I contains less than two squares from the horizontal connection between $G_{i,j}$ and $G_{i,j+1}$ or from the vertical connection between $G_{i,j}$ and $G_{i+1,j}$. For every remaining $G_{i,j}$, we have $|G_{i,j} \cap I| = 16$, hence (as we have argued above), there is an r such that $G_{i,j} \cap I = \{B_{1,r}, \ldots, B_{16,r}\}$. Set $s_{i,j} = \iota^{-1}(r)$ in this case. From $|I| = 16k^2 + 4k(k-1) - t$ it follows that we set at most t cells to $\bigstar$. Assume that $s_{i,j} = (x,y)$ and $s_{i,j+1} = (x',y')$ in the constructed solution; we have to show that $x = x'$. There are two squares $H_{x_1}^1$, $H_{x_2}^2$ selected from the horizontal connection between $G_{i,j}$ and $G_{i,j+1}$ (otherwise $s_{i,j}$ would be $\bigstar$). Now we have that $x \geq x_1 \geq x'$, otherwise $H_{x_1}^1$ would intersect either $B_{6,\iota(x,y)}$ of $G_{i,j}$ or $B_{16,\iota(x',y')}$ of $G_{i,j+1}$. Similarly, by observing $H_{x_2}^2$, $B_{8,\iota(x,y)}$ of $G_{i,j}$, and $B_{14,\iota(x',y')}$ of $G_{i,j+1}$,

we get $x \leq x_2 \leq x'$, hence $x = x'$ follows. With an analogous argument, we can show that if $s_{i,j} = (x,y)$ and $s_{i+1,j} = (x',y')$, then $y = y'$ holds. $\square$

By an argument of [24], the reduction can be made to work for unit disks as well. A similar gadget construction can be used in the case of the MINIMUM DOMINATING SET problem.

Theorem 4.2. *If there is a $\delta > 0$ such that* MAXIMUM INDEPENDENT SET *or* MINIMUM DOMINATING SET *for unit squares or unit disks has a PTAS with running time $2^{(1/\epsilon)^{O(1)}} \cdot n^{O(1/\epsilon)^{1-\delta}}$, then ETH fails.* $\square$

There is a $2^{O(1/\epsilon^2)}n$ time PTAS for MINIMUM VERTEX COVER on unit disk graphs [24, 27], hence the analog of Theorem 4.2 is not true for this problem. However, we can prove a lower bound with the following argument. In the case $D = 2$, the reduction of Theorem 4.1 constructs an intersection graph having bounded degree. For bounded-degree graphs, there is an L-reduction between MAXIMUM INDEPENDENT SET and MINIMUM VERTEX COVER. Hence by (2) of Lemma 2.8,

Theorem 4.3. *If there is a $\delta > 0$ such that* MINIMUM VERTEX COVER *for unit squares or unit disks has a PTAS with running time $2^{(1/\epsilon)^{1-\delta}} \cdot n^{O(1)}$, then ETH fails.* $\square$

Note that the lower bound of Theorem 4.3 does not match the best known approximation scheme. This might suggest that the $2^{O(1/\epsilon^2)}$ PTAS of [24, 27] can be improved to $2^{O(1/\epsilon)}$. We leave this as an open question.

5 Independent Set, Vertex Cover, Dominating Set

Lipton and Tarjan [21] used the planar separator theorem of [20] to show that MAXIMUM INDEPENDENT SET on planar graphs admits a PTAS with running time $2^{O(1/\epsilon)^2}n + O(n \log n)$. Using a completely different approach, Baker [7] improved the dependence on ϵ to $2^{O(1/\epsilon)} \cdot n$. Here we show that Baker's algorithm is essentially optimal:

Theorem 5.1. *If there is a $\delta > 0$ such that* MAXIMUM INDEPENDENT SET *for planar graphs has a PTAS with running time $2^{O(1/\epsilon)^{1-\delta}} \cdot n^{O(1)}$, then ETH fails.*

Proof. The proof is by presenting an L-reduction from MATRIX TILING with $D = 2$ to PLANAR MAXIMUM INDEPENDENT SET. Each of the k^2 sets $S_{i,j}$ will be represented by an appropriate gadget. A gadget is a planar graph with distinguished vertices a_0, a_1, b_0, b_1, c_0, c_1, d_0, d_1 on its boundary (in this order) such that a_0a_1, b_0b_1, c_0c_1, d_0d_1 are edges and there are no other edges between these vertices. Let B be the set of these 8 vertices. We say that a

subset $B' \subseteq B$ *represents* the pair $(x, y) \in Z_2 \times Z_2$ if $B' = \{a_x, c_x, b_y, d_y\}$. The purpose of the gadget representing $S_{i,j}$ is to ensure that the vertices selected from B represent a pair from $S_{i,j}$, otherwise the gadget incurs a penalty. The following lemma states the requirements formally:

Lemma 5.2. *There is a constant $c > 0$ such that for every nonempty set $S \subseteq Z_2 \times Z_2$, there is a gadget G_S such that*

1. *The size of the maximum independent set is $\alpha(G_S) = c + 1$.*

2. *If $B' \subseteq B$ represents a pair in S, then B' can be extended to an independent set of size $c + 1$.*

3. *If $B' \subseteq B$ is an independent set of size 4, then B' can be extended to an independent set of size c.*

4. *If I is an independent set of size $c + 1$, then $B \cap I$ represents a set in S.*

The proof the lemma uses standard techniques: we can argue that every set $S \subseteq Z_2 \times Z_2$ can be represented by a formula, which can be turned into a planar formula, which can be turned into a planar graph using the standard reduction from 3SAT to MAXIMUM INDEPENDENT SET. Details omitted.

We construct a planar graph G the following way. For every $1 \leq i, j \leq k$, we introduce a gadget $G_{i,j}$ that is the copy of $G_{S_{i,j}}$. For every $1 \leq i \leq k$, $1 \leq j < k$, vertex c_0 (resp., c_1) of $G_{i,j}$ is connected with vertex a_1 (resp., a_0) of $G_{i,j+1}$; and for every $1 \leq i < k$, $1 \leq j \leq k$, vertex d_0 (resp., d_1) of $G_{i,j}$ is connected with vertex b_1 (resp., b_0) of $G_{i+1,j}$.

We claim that if the optimum of MATRIX TILING is $k^2 - t$, then $\alpha(G) = (c + 1)k^2 - t$. Assume that there is a solution of MATRIX TILING with value $k^2 - t$. For each $s_{i,j} \neq \star$, let us select a subset $B' \subseteq B$ that represents $s_{i,j}$, and extend it to an independent set of size $c+1$ (Prop. 3 of Lemma 5.2). Observe that the independent sets selected from neighboring gadgets do not conflict. For example, if $s_{i,j}, s_{i,j+1} \neq \star$, then they agree in the first component, hence either c_0 is selected from $G_{i,j}$ and a_0 is selected from $G_{i,j+1}$, or c_1 is selected from $G_{i,j}$ and a_1 is selected from $G_{i,j+1}$. Consider the cells with $s_{i,j} = \star$ in arbitrary order, and in each gadget, select 4 vertices of B that do not conflict with the vertices already selected from the neighboring gadgets (this can be done, since it is not possible that, say, both a_0 and a_1 have a selected neighbor). Extend these 4 vertices into an independent set of size c (Prop. 2 of Lemma 5.2). If t is the number of $\star$'s in the solution of MATRIX TILING, then we obtain an independent set of size $ct + (c+1)(k^2 - t)$, hence $\alpha(G) \geq (c+1)k^2 - t$ follows.

For the other direction, suppose that G has an independent set I of size $(c + 1)k^2 - t$. Since $|I \cap G_{i,j}| \leq c + 1$ (Prop. 1 of Lemma 5.2), $|G_{i,j} \cap I| \leq c$ for at most t of the $G_{i,j}$'s. If $|G_{i,j} \cap I| \leq c$, then set $s_{i,j} = \star$. If $|G_{i,j} \cap I| = c + 1$, then by Prop. 4 of Lemma 5.2, $B \cap I$ in $G_{i,j}$ represents a pair (x, y) from $S_{i,j}$; in this case, set $s_{i,j} = (x, y)$. We can verify that this is a valid solution of MATRIX TILING. For example, if $s_{i,j} = (x_1, y_1)$ and $s_{i,j+1} = (x_2, y_2)$, then $x_1 = x_2$: otherwise $c_{x_1} \in I$ of $G_{i,j}$ and $a_{x_2} \in I$ of $G_{i,j+1}$ would be neighbors. As the number of $\star$'s is t, we get that the optimum of MATRIX TILING is at least $k^2 - t$. $\qquad\square$

Observe that the reduction constructs bounded-degree graphs. For bounded degree graphs, there is an L-reduction from MAXIMUM INDEPENDENT SET to MINIMUM VERTEX COVER. Furthermore, there is an L-reduction from MINIMUM VERTEX COVER to MINIMUM DOMINATING SET, hence we have

Theorem 5.3. *If there is a $\delta > 0$ such that MINIMUM VERTEX COVER or MINIMUM DOMINATING SET for planar graphs has a PTAS with running time $2^{O(1/\epsilon)^{1-\delta}} \cdot n^{O(1)}$, then ETH fails.*

6 TSP on planar graphs

The Traveling Salesperson Problem (TSP) is one of the most studied optimization problems. Here we consider the special case where the distance metric is the shortest-path metric of an unweighted planar graph. That is, given a planar graph, the task is to find a tour (closed walk) of minimum length that visits every vertex at least once. The first PTAS for the problem was given by Grigni et al. [14], with running time $n^{O(1/\epsilon)}$. Recently, this was improved to $2^{O(1/\epsilon)}n$ by Klein [19], which is essentially optimal:

Theorem 6.1. *If there is a $\delta > 0$ such that TSP on unweighted planar graphs has a PTAS with running time $2^{O(1/\epsilon)^{1-\delta}} \cdot n^{O(1)}$, then ETH fails.*

Proof. The proof is by presenting an L-reduction from MATRIX TILING with $D = 2$ to the problem. We say that a tour is *efficient* if after removing any loop (i.e., a subtour that starts and ends on the same vertex) there is at least one vertex that is not visited by the resulting shorter tour. Clearly, the optimum tour is efficient. The purpose of the *exclusive-or line* (Fig. 3) is to force the tour to use exactly one of the two edges e_1 and e_2. Assume that in a graph G edges e_1 and e_2 are connected as in Fig. 3b, and there is an efficient TSP tour T that visits each of the 16 internal vertices of the exclusive-or line exactly once. It can be verified by inspection that tour T enters the exclusive-or line only once, and it either uses both e_1' and e_1'' (Fig. 3c), or both e_2' and e_2'' (Fig. 3d). In fact, we can make the assumption slightly weaker: the same conclusion holds even if we only assume that 15 of the 16 internal vertices are visited exactly once.
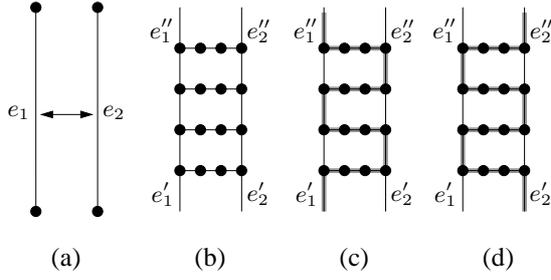
Figure 3. The exclusive-or line.

(a) (b)

Figure 4. The gadget and the way the gadgets are arranged for $k = 3$ (Theorem 6.1).

(Here we use the assumption that T is efficient: otherwise it would be possible that T visits the exclusive-or line twice, the first time visiting every vertex and the second time visiting only one vertex.)

We represent each set $S_{i,j}$ of MATRIX TILING by a gadget $G_{i,j}$. The gadget is a planar graph with 4 distinguished vertices v_1, v_2, v_3, v_4 and 4 distinguished edges a_0, a_1, b_0, b_1 on its boundary (see Fig. 4a). We say that a set $\mathcal{W}$ of walks is a *traversal* of the gadget if both endpoints of each walk is in $\{v_1, v_2, v_3, v_4\}$ and each vertex of the gadget is visited by at least one of the walks. A traversal *uses* an edge e if at least one of the walks go through e. A traversal $\mathcal{W}$ *represents* a pair $(x, y) \in Z_2 \times Z_2$ if a_x and b_y are the only distinguished edges that are used by $\mathcal{W}$. The *weight* of a traversal is the number of vertices in the gadget that are visited more than once. The properties of the gadget are summarized in the following lemma:

Lemma 6.2. *There is a constant $c > 0$ such that for every nonempty set $S \subseteq Z_2 \times Z_2$, there is a gadget G_S such that*

1. *There are c vertices in G_S.*

2. *For every $(x, y) \in S$, there is a traversal $\mathcal{W} = \{P_1, P_2\}$ of weight 0 where simple path P_1 connects v_1 and v_3, simple path P_2 connects v_2 and v_4, and $\mathcal{W}$ represents (x, y).*

3. *For every $Z \subseteq \{a_0, a_1, b_0, b_1\}$, there is a traversal $\mathcal{W} = \{P_1, P_2\}$ of weight 1 where walk P_1 connects v_1 and v_3, walk P_2 connects v_2 and v_4, and $\mathcal{W}$ uses the edges in Z, but not those in $\{a_0, a_1, b_0, b_1\} \setminus Z$.*

4. *If a traversal $\mathcal{W}$ has weight 0, then it represents a set $(x, y) \in S$.*

The proof of this lemma is tedious, but uses only the standard gadgeteering techniques developed for proving the NP-hardness of Hamiltonian cycle. Details will appear in the full version.

Each set $S_{i,j}$ is represented by a gadget $G_{i,j}$ that is a copy of $G_{S_{i,j}}$. The gadgets are connected as shown in

Fig. 4b. Distinguished edges of adjacent gadgets are connected with exclusive-or lines. For $1 \leq i < k$, $1 \leq j \leq k$, vertex v_3 (resp., v_4) of $G_{i,j}$ is connected with vertex v_1 (resp., v_2) of $G_{i+1,j}$. Vertices v_3 and v_4 of gadget $G_{k,j}$ are connected with each other (for $1 \leq j \leq k$), while vertex v_2 of $G_{1,j}$ is connected with vertex v_1 of $G_{1,j+1}$ (for $1 \leq j < k$). Finally, vertex v_1 of $G_{1,1}$ is connected with vertex v_2 of $G_{1,k}$. Observe that the number of vertices of the constructed graph is $c \cdot k^2 + 32k(k-1)$: there are k^2 gadgets with c vertices each, and each of the $2k(k-1)$ exclusive-or lines add 16 new vertices.

We show that if the optimum of MATRIX TILING is $k^2 - t$, then the optimum of the constructed instance of TSP is $c \cdot k^2 + 32k(k-1) + t$. Assume first that MATRIX TILING has a solution where the number of ★'s is t. We construct a tour that starts at vertex v_1 of $G_{1,1}$, goes to vertex v_3 of $G_{1,1}$, goes to vertex v_1 of $G_{2,1}$, goes to vertex v_3 of $G_{2,1}$, etc., until it reaches vertex v_3 of $G_{k,1}$. Using the edge v_3v_4, the tour goes to v_4 of $G_{k,1}$ and revisits the first column by going to v_2 of $G_{k,1}$, to v_4 of $G_{k-1,1}$, to v_2 of $G_{k-1,1}$, etc., until it reaches vertex v_2 of $G_{1,1}$. At this point the tour goes to vertex v_1 of $G_{1,2}$ and visits the second column in a similar way. Finally, when the tour reaches vertex v_2 of $G_{1,k}$, it returns to v_1 of $G_{1,1}$. Each gadget is visited twice: once by a walk from v_1 to v_3, and once by a walk from v_2 to v_4. To complete the construction of the tour, we have to find an appropriate traversal for each gadget. If $s_{i,j} = (x, y) \neq$ ★, then we use the traversal of $G_{i,j}$ given by Prop. 2 of Lemma 6.2. If $s_{i,j}, s_{i,j+1} \neq$ ★, then $s_{i,j} = (x, y_1)$, $s_{i,j+1} = (x, y_2)$ implies that the traversal of $G_{i,j}$ uses edge a_1 if and only if the traversal of $G_{i,j+1}$ does not use edge a_0. Therefore, the tour can be extended to the exclusive-or lines in such a way that the tour visits the vertices of these lines exactly once (Fig. 3c or Fig. 3d). If

$s_{i,j} = \bigstar$, then we choose an appropriate subset of the distinguished edges, and by Prop. 3 of Lemma 6.2, we obtain a traversal of weight 1 that is compatible with the traversals of the neighboring gadgets, i.e., there is no conflict on the exclusive-or lines. If the weight of a traversal is 1, then there is exactly one vertex in the gadget that is visited twice. Thus there are exactly t vertices in the graph that are visited twice, all the other vertices are visited exactly once. Therefore, the total length of the tour is $c \cdot k^2 + 32k(k-1) + t$.

For the other direction, assume that there is an efficient tour T with length $c \cdot k^2 + 32k(k-1) + t$. This means that there are at most t vertices that are visited more than once; let X be the set of these vertices. If $G_{i,j} \cap X \neq \emptyset$, then set $s_{i,j} = \bigstar$. If at least two vertices of X are in the exclusive-or line connecting $G_{i,j}$ and $G_{i,j+1}$, then set $s_{i,j} = s_{i,j+1} = \bigstar$. The exclusive-or line connecting $G_{i,j}$ and $G_{i+1,j}$ is handled similarly. Let $G_{i,j}$ be a gadget such that $s_{i,j}$ was not set to $\bigstar$. Each of the exclusive-or lines connected to $G_{i,j}$ contains at most one vertex of X, thus (as we have observed at the beginning of the proof) the line is used in a "proper way," i.e., as in Fig. 3c or Fig. 3d. Therefore, if we remove the exclusive-or lines, then T induces a traversal $\mathcal{W}$ of $G_{i,j}$ of weight 0 (since $G_{i,j} \cap X = \emptyset$). By Prop. 4 of Lemma 6.2, this means that $\mathcal{W}$ represents a pair $(x, y) \in S_{i,j}$; set $s_{i,j} := (x, y)$. It easy to see that if $s_{i,j} = (x_1, y_1)$ and $s_{i,j+1} = (x_2, y_2)$, then $x_1 = x_2$, otherwise it would not be true that the exclusive-or line connecting a_1 of $G_{i,j}$ and a_0 of $G_{i,j+1}$ contains at most one vertex of X. Similarly, if $s_{i,j}, s_{i+1,j} \neq \bigstar$, then they agree in the second component. The number of $\bigstar$'s is at most the size of X, hence we obtain a solution with value at least $k^2 - t$. □

For the weighted version of the problem, Klein [19] gives a $2^{O(1/\epsilon^2)}n$ time PTAS. We leave it as an open question whether this PTAS is optimal for the weighted version or (as in the unweighted case) it can be improved to $2^{O(1/\epsilon)}n$ to match Theorem 6.1.

References

[1] P. K. Agarwal, M. van Kreveld, and S. Suri. Label placement by maximum independent set in rectangles. *Comput. Geom.*, 11(3-4):209–218, 1998.

[2] J. Alber, H. Fernau, and R. Niedermeier. Parameterized complexity: exponential speed-up for planar graph problems. *J. Algorithms*, 52(1):26–56, 2004.

[3] J. Alber and J. Fiala. Geometric separation and exact solutions for the parameterized independent set problem on disk graphs. *J. Algorithms*, 52(2):134–151, 2004.

[4] A. Andoni, P. Indyk, and M. Patrascu. On the optimality of the dimensionality reduction method. In *FOCS '06: Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*, pages 449–458, Washington, DC, USA, 2006. IEEE Computer Society.

[5] S. Arora. Polynomial time approximation schemes for Euclidean TSP and other geometric problems. In *FOCS 1996*, pages 2–11. IEEE Comput. Soc. Press, 1996.

[6] S. Arora. Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems. *J. ACM*, 45(5):753–782, 1998.

[7] B. S. Baker. Approximation algorithms for NP-complete problems on planar graphs. *J. ACM*, 41(1):153–180, 1994.

[8] C. Bazgan. Schémas d'approximation et complexité paramétrée. Technical report, Université Paris Sud, 1995.

[9] L. Cai, M. Fellows, D. Juedes, and F. Rosamond. The complexity of polynomial-time approximation, 2006. To appear in *Theory of Computing Systems*.

[10] M. Cesati and L. Trevisan. On the efficiency of polynomial time approximation schemes. *Inform. Process. Lett.*, 64(4):165–171, 1997.

[11] J. Chen, X. Huang, I. A. Kanj, and G. Xia. Linear FPT reductions and computational lower bounds. In *STOC 2004*, pages 212–221, New York, 2004. ACM.

[12] I. Dinur. The PCP theorem by gap amplification. In *STOC 2006*, pages 241–250, 2006.

[13] R. G. Downey. Parameterized complexity for the skeptic. In *Proceedings of the 18th IEEE Annual Conference on Computational Complexity*, pages 147–169, 2003.

[14] M. Grigni, E. Koutsoupias, and C. Papadimitriou. An approximation scheme for planar graph TSP. In *FOCS 1995*, pages 640–645. IEEE, Los Alamitos, CA, 1995.

[15] H. B. Hunt, III, M. V. Marathe, V. Radhakrishnan, S. S. Ravi, D. J. Rosenkrantz, and R. E. Stearns. NC-approximation schemes for NP- and PSPACE-hard problems for geometric graphs. *J. Algorithms*, 26(2):238–274, 1998.

[16] R. Impagliazzo, R. Paturi, and F. Zane. Which problems have strongly exponential complexity? *J. Comput. System Sci.*, 63(4):512–530, 2001.

[17] S. Khanna and R. Motwani. Towards a syntactic characterization of PTAS. In *STOC 1996*, pages 329–337.

[18] S. Khanna, M. Sudan, L. Trevisan, and D. P. Williamson. The approximability of constraint satisfaction problems. *SIAM J. Comput.*, 30(6):1863–1920, 2001.

[19] P. Klein. A linear-time approximation scheme for planar weighted TSP. In *FOCS 2005*, pages 647–656. IEEE, 2005.

[20] R. J. Lipton and R. E. Tarjan. A separator theorem for planar graphs. *SIAM J. Appl. Math.*, 36(2):177–189, 1979.

[21] R. J. Lipton and R. E. Tarjan. Applications of a planar separator theorem. *SIAM J. Comput.*, 9(3):615–627, 1980.

[22] E. Malesińska. *Graph-Theoretical Models for Frequency Assignment Problems*. PhD thesis, TU Berlin, 1997.

[23] D. Marx. The closest substring problem with small distances. In *FOCS 2005*, pages 63–72, 2005.

[24] D. Marx. Efficient approximation schemes for geometric problems? In *ESA 2005*, pages 448–459, 2005.

[25] D. Marx. Parameterized complexity and approximation algorithms, 2006. To appear in *The Computer Journal*.

[26] S. B. Rao and W. D. Smith. Approximating geometrical graphs via "spanners" and "banyans". In *STOC '98 (Dallas, TX)*, pages 540–550. ACM, New York, 1999.

[27] E. J. van Leeuwen. Better approximation schemes for disk graphs. In *SWAT 2006*, pages 316–327, 2006.

[28] D. W. Wang and Y.-S. Kuo. A study on two geometric location problems. *Inform. Process. Lett.*, 28(6):281–286, 1988.