Inapproximability of the Perimeter Defense Problem

Evangelos Kranakis* Danny Krizanc† Lata Narayanan‡ Kun Xu§

Abstract

We model the problem of detecting intruders using a set of infrared beams by the perimeter defense problem: given a polygon P, find a minimum set of edges $\mathcal{S}$ of the polygon such that any straight line segment crossing the polygon intersects at least one of the edges in $\mathcal{S}$. We observe that this problem is equivalent to a new hiding problem, the **M**ax-**H**idden-**E**dge-**S**et problem. We prove the APX-hardness of the **M**ax-**H**idden-**E**dge-**S**et problem for polygons without holes and rectilinear polygons without holes, by providing gap-preserving reductions from the **M**ax-**5**-**O**ccurrence-**2**-**S**at problem.

1 Introduction

Given a region of interest to be defended, we are interested in detecting the presence of an intruder inside the region who originated from outside the region. We model the region of interest by a polygon, and the trajectory of the intruder by a curve intersecting the interior of the polygon. For arbitrary curves, or for line segments that can terminate inside the polygon, there is no choice but to defend the entire perimeter of the polygon. Therefore, we consider the case when the path of the intruder is a straight line segment that *crosses* the polygon (intersects the perimeter of the polygon in at least two distinct edges) and require the intruder to be detected before exiting the polygon.

Infrared beam sensors are an increasingly popular way of achieving intruder detection. Such a device consists of a matched transmitter-receiver pair; the transmitter emits an infrared beam to a receiver module. Usually the beam distance can be adjusted. An intruder going across the beam would interrupt the circuit and be detected. In several applications, it may make sense to place the beams only on the perimeter of the polygon, as allowing beams to intersect either the interior or the exterior of the polygon may lead to false alarms. We are interested in minimizing the number of infrared beams to be placed on the perimeter that are required to ensure that any intruder L whose path crosses the polygon will be detected. This implies that the transmitter and receiver should be placed on adjacent vertices of the polygon, so that the beam is aligned with the edge between them. Our intruder detection problem can therefore be modeled as follows:

Definition 1 ***M**inimum-**E**dge-**P**erimeter-**D**efense : Given a polygon P, find a minimum-sized subset $\mathcal{S}$ of edges of P such that any straight line segment L crossing P intersects at least one edge in $\mathcal{S}$.*

It is not difficult to see that this problem can be reduced to a *hiding* problem, i.e. finding a maximum-sized subset of mutually invisible edges of the polygon [1]. Indeed $\mathcal{S}$ is a solution to the perimeter defense problem if and only if all elements in $\overline{\mathcal{S}}$ are mutually invisible. In what follows, we focus on the **M**ax-**H**idden-**E**dge-**S**et problem:

Definition 2 ***M**ax-**H**idden-**E**dge-**S**et : Given a polygon P, find a maximum-sized subset of mutually invisible edges of the polygon.*

Guarding and hiding problems have been studied extensively in the literature. The *Maximum Hidden Set* (MHS) problem introduced in [1] is to find a maximum-sized set of mutually invisible points in a polygon. In the *Maximum Hidden Vertex Set* ($MHVS$), the points are constrained to be vertices of the polygon. Hiding and guarding problems are combined in the *Minimum Hidden Guard Set* ($MHGS$) and the *Minimum Hidden Vertex Guard Set* ($MHVGS$) and *Hidden Vertex Guard Admissibility* problems. All these problems were shown to be NP-complete and lower and

*School of Computer Science, Carleton University, kranakis@scs.carleton.ca
†Department of Mathematics, Wesleyan University, dkrizanc@wesleyan.edu
‡Department of Computer Science and Software Engineering, Concordia University, lata@cse.concordia.ca
§Department of Computer Science and Software Engineering, Concordia University, ku_xu@encs.concordia.ca

[1] Two edges e_1 and e_2 are invisible from each other iff for every $p_1 \in e_1$ and $p_2 \in e_2$ such that the line connecting p_1 and p_2 lies entirely within the polygon, at least one of p_1 and p_2 is an endpoint of its edge.

upper bounds for their approximation ratios were given in [1]. The restriction of the problem instance to a terrain was proved to be NP-complete in [3].

In [3, 6] it was shown that for polygons with holes or terrains, the MHS and $MHVS$ problems cannot be approximated by a polynomial time algorithm with an approximation ratio of n^ϵ for some $\epsilon > 0$. For polygons without holes, these problems were shown to be APX-hard. Recently, Eidenbenz [7] presented an inapproximability result for the $MHGS$ problem. He proved that for input polygons with or without holes or terrains, the $MHGS$ problem is also APX-hard. Notice that the $MHVGS$ problem is a much harder problem. It is NP-hard to even determine whether a feasible solution exists [1].

To our knowledge, the complexity of the **M**ax-**H**idden-**E**dge-**S**et problem has not been studied. Since a set of mutually invisible edges in a polygon is an independent set of vertices in the visibility graph of the edges of the polygon, the **M**ax-**H**idden-**E**dge-**S**et problem can be reduced to the Maximum Independent Set problem, and therefore is approximable with an $O(n(\log\log n)^2/\log^3 n)$ approximation ratio [2]. Not every graph is a visibility graph of a polygon (for example, $K_{2,3}$), and therefore, the reduction does not go through in the other direction.

Our Results

In this paper, we prove that the **M**ax-**H**idden-**E**dge-**S**et problem is APX-hard for polygons without holes. The proof is using a reduction from **M**ax-**5**-**O**ccurrence-**2**-**S**at problem, which was shown to be APX-hard in [4, 5]. In fact, we show that the **M**ax-**H**idden-**E**dge-**S**et problem is APX-hard even when restricted to rectilinear polygons without holes. It follows that the **M**inimum-**E**dge-**P**erimeter-**D**efense problem is also APX-hard even for rectilinear polygons without holes.

2 APX-hardness of Max-Hidden-Edge-Set for an Arbitrary Polygon

In this section, we show that the **M**ax-**5**-**O**ccurrence-**2**-**S**at problem is transformable in polynomial time to **M**ax-**H**idden-**E**dge-**S**et by an approximation-preserving (gap-preserving) reduction [8].

Definition 3 *Let Φ be a boolean formula given in conjunctive normal form, with at most two literals in each clause and each variable appearing in at most 5 five clauses. The **M**ax-**5**-**O**ccurrence-**2**-**S**at problem consists of finding a truth assignment for the variables of Φ such that the number of satisfied clauses is maximum.*

Construction

The goal is to accept an instance of **M**ax-**5**-**O**ccurrence-**2**-**S**at as input and in polynomial time to construct a connected simple polygonal region P such that the difference in the number of hidden edges obtained by the optimal and approximation algorithms preserves the gap between the optimal and approximate results (the number of satisfied clauses) in **M**ax-**5**-**O**ccurrence-**2**-**S**at . The construction is similar to the one proposed in [6]. As shown in Figure 1, the main body is a convex polygon without holes inside; we refer to it as the *center polygon*. For each clause, a *clause pattern* is built on the top right of the center polygon, and for each variable, a *variable pattern* is built on the bottom left of the center polygon. Variable patterns are separated by the *cb*-edges and form a convex curve along the center polygon's bottom. A basic unit in both types of patterns is a *dent*: a set of continuous line segments that form a convex shape (see Figure 2). It is clear that at most one edge from any dent can be included in the **M**ax-**H**idden-**E**dge-**S**et . Now we show how to construct the clause and variable patterns.
Clause Patterns: The clause pattern is shown in Figure 3. Each pattern consists of 15 adjacent edges forming three dents. Each dent has exactly five edges. Without loss of generality, we assume that all clauses contain two literals. Then for each clause we use the left and right dents to represent the two literals and the middle dent to represent the satisfiability of the clause.
Variable Patterns: The variable pattern is shown in Figure 4. Each variable has a TRUE-leg and a FALSE-leg, each consisting of three components: L-dents, M-dents and E-dents. Observe that no two dents in the same component can see each other. Five L-dents are arranged along a line, separated by g-edges, and each L-dent (L_i) consists of one t-edge (t_i), one b-edge (b_i) and one l-edge (l_i). Each occurrence of a variable matches a pair of L-dents, one from the variable's TRUE-leg and the other from the variable's FALSE-leg. Since each variable appears in at most 5 clauses, we only need five L-dents for each leg. Opposite to the L-dents are four adjacent M-dents and six adjacent E-dents. Each M-dent consists of three edges. We denote the middle edge for a dent M_i by m_i. Finally, six E-dents form a simple zig-zag line under the M-dents. We pick up every alternate edge from this collection of dents and denote the edge chosen from E_i by e_i. By choosing the length and direction of each edge appropriately, each leg of the variable pattern can be constructed with the following properties.

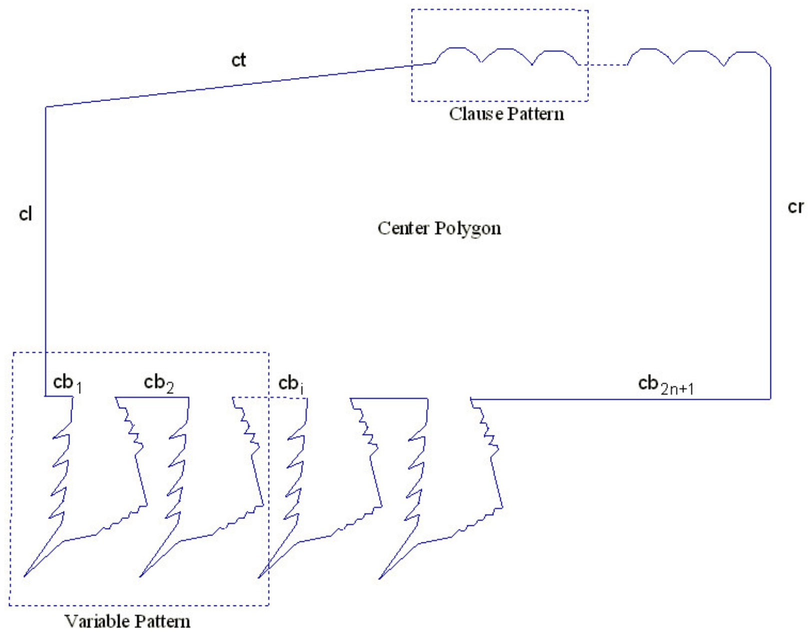

Figure 1: Overview of Construction (Arbitrary Polygon)

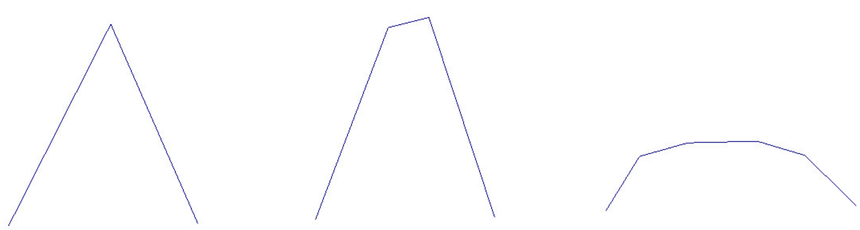

Figure 2: Dents

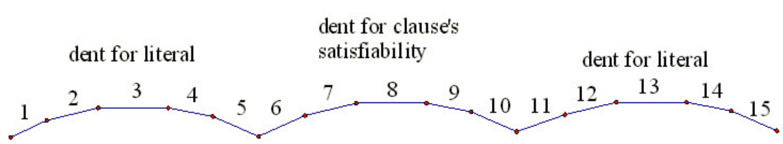

Figure 3: Clause Pattern (Arbitrary Polygon)

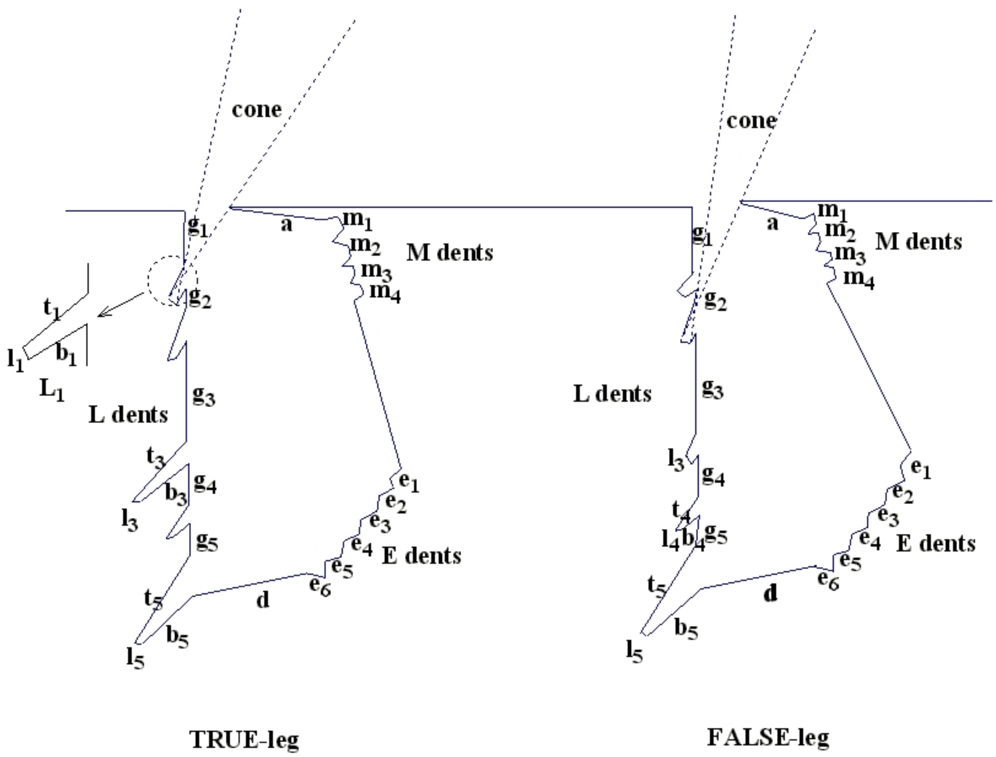

Figure 4: Variable Pattern (Arbitrary Polygon)

P1 For $1 \leq i \leq 5$, t_i, g_i, a and d can each see all the edges in component M and E, and edge g_i can see d and a.

P2 For $1 \leq i \leq 5$, b_i and l_i see all the edges in component M and edge a, but cannot see any edge in component E or edge d.

P3 For $1 \leq i \leq 4$, m_i can see all the edges in component L and g, but no edge in component E.

P4 For $1 \leq i \leq 5$, if l_i does not match any occurrence of a variable, l_i sees only the M-dent and edge a.

P5 Components M and E are angled so that they cannot be seen by any clause pattern.

The last step of the construction is to establish the relationship between the clause and variable patterns. As shown in Figure 5, we connect them by cones. Each cone starts at an l-edge and ends at a clause pattern and the clause pattern's edges inside the cone are visible to the l-edge. Consequently, if we add an l-edge (cone's bottom) to the hidden edge set, we cannot add any edge at the top of the cone. Further, the cones are overlapped in specific ways, as shown in Figure 5.

Analysis

Next we show the relationship between a satisfying assignment for a **M**ax-**5**-**O**ccurrence-**2**-**S**at instance and a hidden edge set for the corresponding polygon. Let I be an instance of the **M**ax-**5**-**O**ccurrence-**2**-**S**at problem with n variables and m clauses and I' be the instance of the corresponding **M**ax-**H**idden-**E**dge-**S**et problem. We assume without loss of generality that every variable has more than one occurrence in I (if not, then the unique clause containing the variable can definitely be satisfied).

Lemma 1 *If I has an assignment S that satisfies at least $(1-\epsilon)m$ clauses, then I' has a solution S' with at least $21n + 2m + (1-\epsilon)m$ edges.*

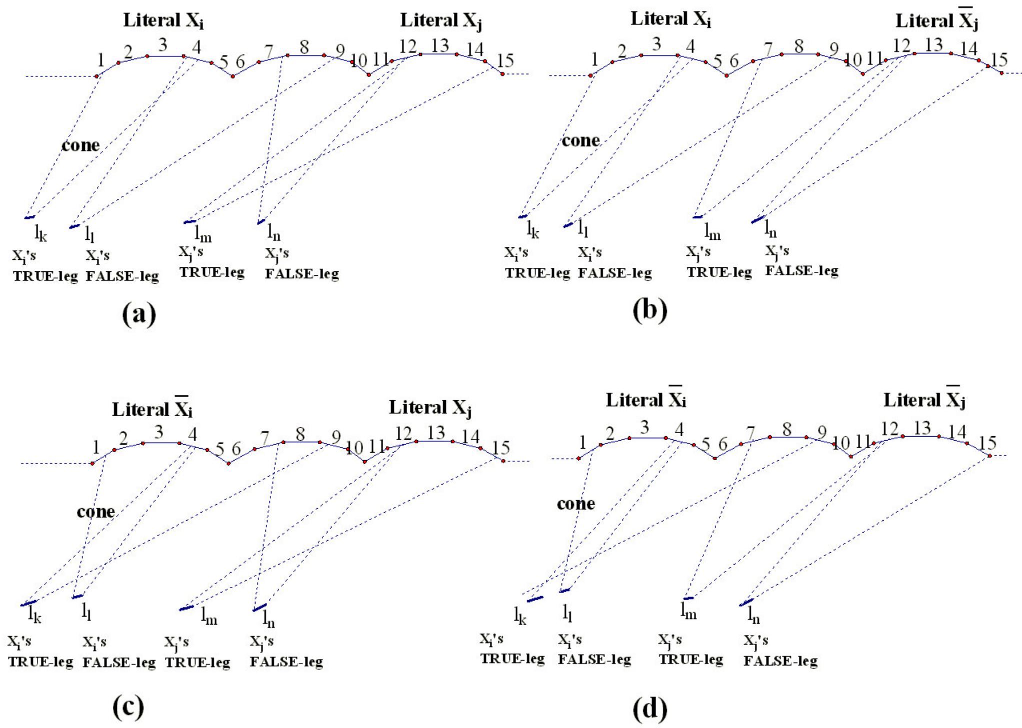

Figure 5: Relationship between Clause and Variable Patterns (Arbitrary Polygon)

Proof. For a variable with a TRUE assignment, we add all the i-edges and e-edges from its TRUE-leg and all the m-edges and e-edges from its FALSE-leg (and vice-versa for a variable with a FALSE assignment). So no matter what the truth value is, we can add 5 l-edges, 4 m-edges and 12 e-edges. Since there are n variable patterns, we add $21n$ edges to S'.

Next we add edges from the clause patterns. We show that we can always add 3 edges for each satisfied clause and 2 edges for each unsatisfied clause. Consider a clause with two literals (x_i, x_j), see Figure 5.(a). We examine all the possible assignments of x_i and x_j in S.

Suppose x_i and x_j are both TRUE. Then, all l-edges in the TRUE-legs of variable patterns corresponding to x_i and x_j have already been added to S'. Therefore we cannot add any of the edges 1-4 or 12-15 from the clause pattern to S', because all these edges are visible to the l-edges mentioned above. On the other hand, no l-edge of both variables' FALSE-legs belongs to S', thus any of edges 5-11 can be added to S'. Since for each dent at most one edge belongs to the hidden set, we can add edges 5, 11, and any of the edges 6-10 to S'. A similar analysis can be used for all other truth value combinations for x_i and x_j to show each unsatisfied clause contributes 2 edges and each satisfied clause contributes 3 edges to S'. Since S satisfies at least $(1-\epsilon)m$ clauses, we conclude that S' has at least $21n + 2(m - (1-\epsilon)m) + 3(1-\epsilon)m = 21n + 2m + (1-\epsilon)m$ edges. □

The next two lemmas give us the relationship in the other direction.

Lemma 2 *Given a solution S' to the **M**ax-**H**idden-**E**dge-**S**et instance I', without decreasing the number of the hidden edges, we can transform it such that the contribution from each variable pattern leg to S' is all its e-edges and either some subset of its l- or all of its m-edges.*

Proof. We consider two cases for each variable pattern leg: either an edge from L-dents belongs to S' or not. If a t_i-edge of an L-dent belongs to S', then replace it with the corresponding l_i-edge. By choosing the l_i edge to be small enough and the angle of the L-dent appropriately, we can ensure that the edges visible to l_i-edge and t_i-edge are exactly the same (though they see slightly different parts of these edges). Therefore, the edges in the solution S' belonging to the relevant clause pattern can remain in S'. If a b_i-edge of an L-dent belongs to S', we can apply a similar replacement to it. It is easy to see that in this case, no M-dent edge could have belonged to S'. It follows from property **P1** that at most five edges from the a-, d- and g- edges can be in the solution S', which can be replaced by six e-edges. In the second case, there was no edge in S' from any of the L-dents. It is easy to verify that the largest number of hidden edges that can be chosen from the variable pattern leg without choosing any L-dent edges is the set of all m- and e-edges, which do not see any clause pattern edges at all, and therefore maximize the number of clause pattern edges that can be included in S'. □

Lemma 3 *If the **M**ax-**H**idden-**E**dge-**S**et instance I' has a solution S' with at least $21n + 3m - (\epsilon + \gamma)m$ edges, then the corresponding **M**ax-**5**-**O**ccurrence-**2**-**S**at instance I has an assignment S which satisfies at least $(1 - \epsilon - \gamma)m$ clauses.*

Proof. By Lemma 2, we can assume that S' has the property that from any variable pattern leg, all e-edges and either some subset of l-edges or all m-edges are in S'. Next, we transform S' without decreasing its size so that every variable pattern contributes all l- and e-edges from one leg and all m and e-edges from the other leg. The transformation consists of the following two key steps. First, for each variable pattern, we always choose the leg which contains more l-edges in S'. If an l-edge in this leg does not belong to S', then we simply add it to S'; there can be at most one clause pattern edge in S' that can be seen by this l-edge, which now has to be removed from S'.

Second, for the other leg, we remove all its l-edges from S' and add all its m-edges to S'. If fewer than five l-edges were removed, then clearly, the size of the solution has not decreased. If instead we replaced five l-edges with 4 m-edges, the size of S' is decreased by one. But since there are at least two occurrences for each variable in the instance, at least two of the removed l-edges match the literal dents in the clause patterns, which combined with the fact that all l-edges from the other leg are in S' implies that one extra edge from each of the clause patterns matching the removed l-edges can be added to S'. These extra edges could not have been in S' before, because they were covered by the (removed) l-edges.

Now we can easily construct a truth assignment with the required properties: if a variable pattern contributes l-edges from its TRUE-leg, then it is assigned TRUE, and otherwise it is assigned FALSE. The maximum number of edges in S' contributed by variable patterns is $21n$. We claim that apart from these variable pattern edges, S' contains only clause pattern edges. This is because for $1 \le i \le 2n + 1$, edge cb_i, cl, ct and cr can see each other; at most one of these could be part of a hidden set. However, all these edges also see all edges in all the clause patterns, therefore a solution S' with the required size can only have clause pattern edges. It is known that 2 edges from each clause pattern, for a total of $2m$ edges, can be added to S' no matter it is satisfied or not. If S' has at least $21n + 2m + (1 - \epsilon - \gamma)m$ edges, we must have at least $(1 - \epsilon - \gamma)m$ satisfied clauses in the corresponding **M**ax-**5**-**O**ccurrence-**2**-**S**at instance I. □

Now we show the APX-hardness of the **M**ax-**H**idden-**E**dge-**S**et problem. Let I be an instance of the **M**ax-**5**-**O**ccurrence-**2**-**S**at problem with n variables and m clauses and I' be the instance of the corresponding **M**ax-**H**idden-**E**dge-**S**et problem. We denote the optimal solutions for these problems by $OPT(I)$ and $OPT(I')$ respectively. From Lemma 1 and Lemma 3 we have:

1. $|OPT(I)| \ge (1 - \epsilon)m \rightarrow |OPT(I')| \ge 21n + 2m + (1 - \epsilon)m$
2. $|OPT(I)| < (1 - \epsilon - \gamma)m \rightarrow |OPT(I')| < 21n + 3m - (\epsilon + \gamma)m$

It is known that **M**ax-**5**-**O**ccurrence-**2**-**S**at is APX-hard. Therefore, for an instance I such that either $|OPT(I)| \ge (1 - \epsilon)m$ or $|OPT(I)| < (1 - \epsilon - \gamma)m$ for some constants $\epsilon, \gamma > 0$, it is NP-hard to decide which case is true. We claim that unless P=NP, no polynomial time approximation algorithm for **M**ax-**H**idden-**E**dge-**S**et can achieve an approximation ratio better than $\frac{21n+2m+(1-\epsilon)m}{21n+3m-(\epsilon+\gamma)m}$. Suppose to the contrary, that a polynomial time approximation algorithm denoted by APO has a performance ratio $< \frac{21n+2m+(1-\epsilon)m}{21n+3m-(\epsilon+\gamma)m}$. Given an instance of **M**ax-**5**-**O**ccurrence-**2**-**S**at such that either $|OPT(I)| \ge (1 - \epsilon)m$ or $|OPT(I)| < (1 - \epsilon - \gamma)m$ for some constants $\epsilon, \gamma > 0$, we apply our reduction to obtain an instance I' of **M**ax-**H**idden-**E**dge-**S**et . If $|APO(I')| \ge 21n + 3m - (\epsilon + \gamma)m$ then $|OPT(I')| \ge 21n + 3m - (\epsilon + \gamma)m$, which further means $|OPT(I)| \ge (1 - \epsilon - \gamma)m$ (because of (2) above). Because I can belong only to one of the two categories, we know that $|OPT(I)| \ge (1 - \epsilon)m$. If instead $|APO(I')| < 21n + 3m - (\epsilon + \gamma)m$, then $\frac{|OPT(I')|}{|APO(I')|} < \frac{21n+2m+(1-\epsilon)m}{21n+3m-(\epsilon+\gamma)m}$ implies $|OPT(I')| < \frac{21n+2m+(1-\epsilon)m}{21n+3m-(\epsilon+\gamma)m}|APO(I')| < 21n + 2m + (1 - \epsilon)m$. Therefore $|OPT(I)| < (1 - \epsilon)m$ which implies $|OPT(I)| < (1 - \epsilon - \gamma)m$. Consequently, it will be possible to decide in polynomial time which category the instance I belongs to, which contradicts the APX-hardness of the **M**ax-**5**-**O**ccurrence-**2**-**S**at problem. We now calculate the ratio. Using the fact that $n < 2m$, we have

$$\frac{21n + 2m + (1-\epsilon)m}{21n + 3m - (\epsilon+\gamma)m} = \frac{1}{1 - \frac{\gamma m}{21n+3m-\epsilon m}} \geq \frac{1}{1 - \frac{\gamma m}{42m+3m-\epsilon m}} \geq 1 + \epsilon' \quad (1)$$

Theorem 4 *There exists a constant $\epsilon > 0$ such that no polynomial time approximation algorithm for the **M**ax-**H**idden-**E**dge-**S**et problem on polygons without holes can have an approximation ratio of $1+\epsilon$, unless $P = NP$.*

3 APX-hardness of Max-Hidden-Edge-Set for a Rectilinear Polygon

If we restrict the **M**ax-**H**idden-**E**dge-**S**et problem to a rectilinear polygon, we can apply a very similar reduction and analysis. The new overall construction, clause pattern, variable pattern and the relationship are shown in Figures 6-9. In Figure 6, we add two new components UC-dents and LC-dents. Due to the fact that the center polygon is rectilinear, *cb*- and *ct*-edges cannot form a convex curve as in the non-rectilinear case. Therefore we need UC- and LC-dents to make sure *cb*- and *ct*-edges are not included in the maximal hidden set. We respectively put $(m+2)$ UC-dents and $(2n+2)$ LC-dents on the center polygon's top right corner and bottom right corners. Notice that every *ct*-edge (*cb*-edge) can see the UC- (LC-) dents completely. In Figure 7, we modify the clause pattern to contain three steps on the right side. In each step's corner, there exists a small convex polygon. It is easy to see that from each step we can add two edges, and a total of six edges for a clause, to the hidden edge set. In Figure 8, we change the shape of the L-dent. As depicted in the figure, for each L-dent we can add at most two edges, l_3 and l_4 (or l_3 and l_5), to the solution. On the right side of each leg, we have 9 M-dents and 11 E-dents. By choosing the appropriate heights and lengths of edges, we make each L-dent's $\{l_2, l_3, l_4, l_5\}$ edges see M-dents completely and $\{l_1, l_7\}$ edges see E-dents completely. Figure 9 shows the relationship between the clause and variable pattern. We draw the figure for $(X_i, \overline{X_j})$. Similar to the earlier case, if the clause is satisfied, we can add at most 6 edges to the solution; otherwise, we can only add 5 edges. Observe that a similar construction works accordingly for other types of clauses.

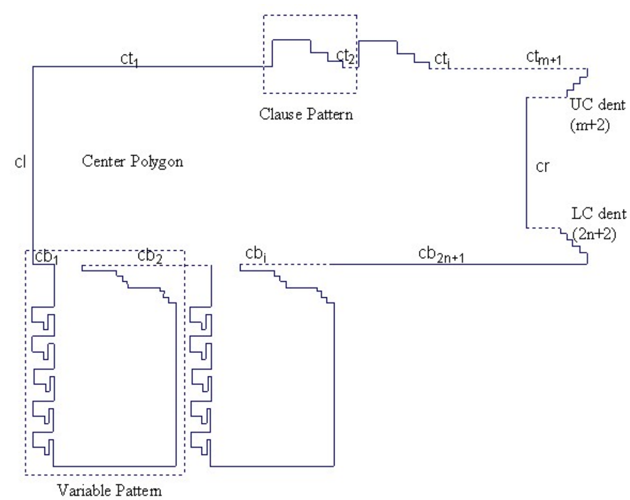

Figure 6: Overview of Construction (Rectilinear Polygon)

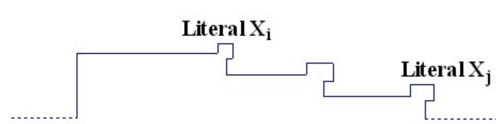

Figure 7: Clause Pattern (Rectilinear Polygon)

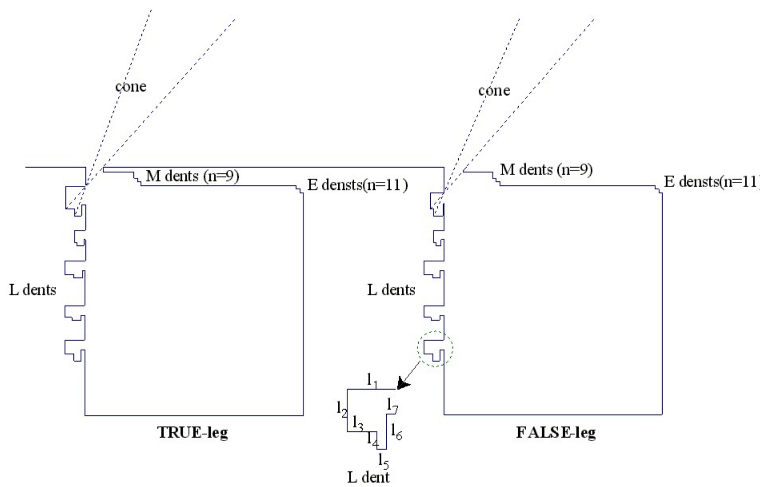

Figure 8: Variable Pattern (Rectilinear Polygon)

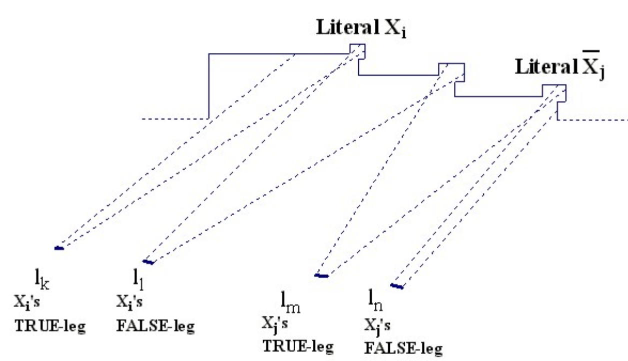

Figure 9: Relationship between Clause and Variable Patterns (Rectilinear Polygon)

Lemma 5 *If the **M**ax-**5**-**O**ccurrence-**2**-**S**at instance I with n variables and m clauses has an assignment S that satisfies at least $(1-\epsilon)m$ clauses, then the corresponding **M**ax-**H**idden-**E**dge-**S**et instance I' has a solution S' with at least $43n + 6m + 4 + (1-\epsilon)m$ edges.*

Proof. Similar to the proof in Lemma 1, for each variable pattern, we can include at most 41 edges (10 from the L-dents of one leg, 9 from the M-dents of the other leg and 11 from the E-dents of each leg) to the hidden edge set. For the center polygon we can include $2n+2$ edges from the LC-dents and $m+2$ edges from the UC-dents. Finally, 5 edges from each clause pattern and $(1-\epsilon)m$ extra edges corresponding to the satisfied clauses can be included. □

Lemma 6 *If the **M**ax-**H**idden-**E**dge-**S**et instance I' has a solution S' with at least $43n + 6m + 4 + (1-\epsilon-\gamma)m$ edges for a constructed polygon, then the corresponding **M**ax-**5**-**O**ccurrence-**2**-**S**at instance I with n variables and m clauses has an assignment S which satisfies at least $(1-\epsilon-\gamma)m$ clauses.*

Proof. Similar to the proof in Lemma 3, without decreasing the number of hidden edge, we can convert any given hidden edge set to one following the standard structure (one leg contains all L-dents and the other leg contains all M-dents). Then we conclude the proof with a similar argument used in Lemma 3. □

The following theorem follows immediately from Lemma 5 and Lemma 6.

Theorem 7 ***M**ax-**H**idden-**E**dge-**S**et is APX-hard even when restricted to a rectilinear polygon without holes.*

Corollary 8 ***M**inimum-**E**dge-**P**erimeter-**D**efense is APX-hard, even for rectilinear polygons without holes.*

The NP-hardness of **M**inimum-**E**dge-**P**erimeter-**D**efense problem follows immediately from Corollary 8.

4 Discussion

We proved the APX-hardness of the **M**ax-**H**idden-**E**dge-**S**et problem for rectilinear polygons, which implies the APX-hardness of the **M**inimum-**E**dge-**P**erimeter-**D**efense problem. In light of the lower bound of $n^{1-O(1/(\log n)^{\gamma})}$ (where γ is a constant) for the approximation ratio for the Maximum Independent Set problem, it would be interesting to know if a higher lower bound also applies for our problems. The complexity of **M**ax-**H**idden-**E**dge-**S**et for monotone rectilinear problems remains an interesting open problem.

References

[1] T. Shermer. Hiding people in polygons. *Computing*, 42(2-3):109–131, 1989.

[2] U. Feige. Approximating maximum clique by removing subgraphs. *SIAM J. Discret. Math.*, 18(2):219–225, 2005.

[3] Stephan Eidenbenz. How many people can hide in a terrain? In *ISAAC*, pages 184–194, 1999.

[4] C.H. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.

[5] Pierluigi Crescenzi and Viggo Kann. Approximation on the web: A compendium of NP optimization problems. In *RANDOM*, pages 111–118, 1997.

[6] Stephan Eidenbenz. Inapproximability of finding maximum hidden sets on polygons and terrains. *Comput. Geom.*, 21(3):139–153, 2002.

[7] Stephan Eidenbenz. Finding minimum hidden guard sets in polygons - tight approximability results. *Comput. Geom.*, 34(2):49–57, 2006.

[8] S. Arora and C. Lund. Hardness of approximations. In *Approximation algorithms for NP-hard problems*, pages 399–446. PWS Publishing Co., Boston, MA, USA, 1997.