Computing Conforming Partitions with Low Stabbing Number for Rectilinear Polygons*

Therese Biedl[1], Stephane Durocher[2], Debajyoti Mondal[3], Rahnuma Islam Nishat[4], and Bastien Rivier[4]

[1]University of Waterloo, Waterloo, Canada, biedl@uwaterloo.ca
[2]University of Manitoba, Winnipeg, Canada, durocher@cs.umanitoba.ca
[3]University of Saskatchewan, Saskatoon, Canada, dmondal@cs.usask.ca
[4]Brock University, Saint Catharines, Canada, rnishat@brocku.ca, brivier@brocku.ca

Abstract

A *conforming partition* of a rectilinear n-gon P is a partition of P into rectangles without using Steiner points (i.e., all corners of all rectangles must lie on P). The stabbing number of such a partition is the maximum number of rectangles intersected by an axis-aligned segment lying in the interior of P. In this paper, we examine the problem of computing conforming partitions with low stabbing number. We show that computing a conforming partition with stabbing number at most 4 is $\mathcal{NP}$-hard, which strengthens a previously known hardness result [Durocher & Mehrabi, Theor. Comput. Sci. 689: 157-168 (2017)] and eliminates the possibility for fixed-parameter-tractable algorithms parameterized by the stabbing number unless $\mathcal{P} = \mathcal{NP}$. In contrast, we give (i) an $O(n \log n)$-time algorithm to decide whether a conforming partition with stabbing number 2 exists, (ii) a fixed-parameter-tractable algorithm parameterized by both the stabbing number and treewidth of the pixelation of the polygon, and (iii) a fixed-parameter-tractable algorithm parameterized by the stabbing number for simple polygons in general position.

1 Introduction

Partitioning an n-gon P with nice properties is a fundamental paradigm in computational geometry. We are interested in the *stabbing number* of a partition, i.e., the maximum number of elements of the partition that are intersected by a straight line segment that lies interior to the polygon. Consider a partition of a polygon into triangles. Such a partition yields a data structure to efficiently process a ray shooting query inside the polygon: a ray is traced by traversing the sequence of triangles that are stabbed by the ray. Since the running time is proportional to the number of stabbed triangles, it is desirable to find a triangular partition such that no ray intersects too many triangles, or in other words, to minimize the stabbing number. Hershberger and Suri [15] showed that every simple polygon has a triangular partition with stabbing number $O(\log n)$ and there exist polygons where any triangular partition has stabbing number $\Omega(\log n)$. There is also an $O(1)$-approximation algorithm for minimizing the stabbing number of triangular partitions [1].

In this paper, we restrict the attention to rectilinear polygons, partition them into rectangles, and for the stabbing number only consider line segments that are in the interior of the polygon and axis-aligned (we call these *stabbing segments*). More precisely, we study the following problem for a rectilinear n-gon P, possibly with holes: partition P into rectangles while minimizing the *stabbing number of the partition*, that is, the maximum over all stabbing segments $\mathbf{s}$ of the number of partition rectangles intersected by $\mathbf{s}$. We

*This paper is dedicated to the memory of our friend Saeed, whose work inspired the project. This work is funded in part by the Natural Sciences and Engineering Research Council of Canada (NSERC).

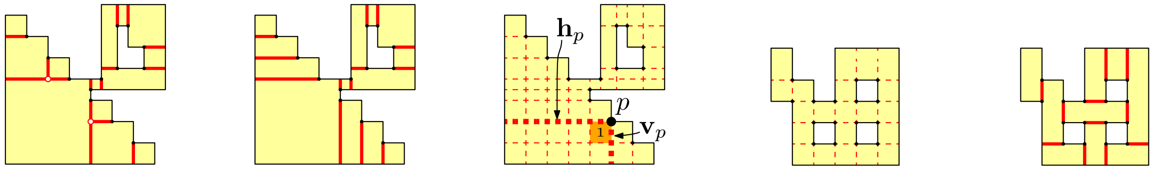


Figure 1: (a) An optimal rectangular partition of P_1 using two Steiner points (tiny hollow circles) with stabbing number 3. The portion of the edges of partition rectangles which are not on the boundary of P_1 are plain bold (red). (b) An optimal conforming partition of P_1 with stabbing number 4. (c) The pixelation of a polygon P_1 in general position with one hole. The reflex vertices are tiny (black) discs and the reflex segments are dotted (red). The horizontal and vertical reflex segments $\mathbf{h}_p$ and $\mathbf{v}_p$ from the reflex vertex p are bold. The wedge-pixel of p is shaded (in orange) and labeled 1.(d) The pixelation of a thin polygon P_2 with three holes (not in general position). (e) An optimal conforming partition of P_2 with stabbing number 3.

often describe such a partition via the inserted segments. A *Steiner point* of a partition is an endpoint of a segment that does not lie on P. We say that such a rectangular partition is *optimal*, and call its stabbing number the *(minimum) stabbing number of* P. Figure 1(a) shows an example of an optimal partition. Similar to triangular partitions, every rectilinear polygon has stabbing number $O(\log n)$, and there exist polygons of arbitrary size with stabbing number $\Omega(\log n)$ [9]. However, there also exist arbitrary-size polygons with stabbing number $O(1)$. To this end, Abam et al. [1] gave a 3-approximation algorithm for computing the stabbing number of simple rectilinear polygons. An interesting open problem in this context is to determine the computational complexity of computing the stabbing number for simple polygons. Although this question remains open in general, there has been some progress on a variant of rectangular partition called conforming partition.

A *conforming partition* of a rectilinear polygon P is a rectangular partition without Steiner points. Put differently, the partition is obtained by using internally disjoint axis-aligned segments that are *maximal* (i.e., both endpoints are in P). To minimize the stabbing number, it suffices to restrict the attention to partitions that use only *reflex segments*, i.e., maximal axis-aligned open segments where one endpoint is a reflex vertex of P (Figure 1(b)). Again, we say that a conforming partition is *optimal* if its stabbing number is minimum among all the conforming partitions, and we call this stabbing number the *conforming stabbing number of* P. Durocher and Mehhrabi [12, 11] showed that computing an optimal conforming partition is $\mathcal{NP}$-hard for polygons with holes, and gave a 2-approximation algorithm for computing the conforming stabbing number (see also [16] for experimental results). However, the complexity of the problem remains open for simple polygons without holes.

Contributions. In this paper, we investigate the problem of computing an optimal conforming partition of rectilinear polygons (possibly with holes) from the perspective of designing *fixed-parameter tractable (FPT) algorithms*, i.e., algorithms with a running time of the form $f(k)n^{O(1)}$ for some chosen parameter k and some computable function $f(\cdot)$ that is independent of n. A natural question in the context of asking for (conforming) partitions with stabbing number at most k is to search for an FPT algorithm parameterized by k. We show that such an algorithm does not exist unless $\mathcal{P} = \mathcal{NP}$. Specifically, deciding whether the conforming stabbing number (and in fact the stabbing number) of a polygon is at most 4 remains $\mathcal{NP}$-hard (Section 3). This strengthens the $\mathcal{NP}$-hardness result of Durocher and Mehhrabi [12], who show it is $\mathcal{NP}$-hard to determine whether the conforming stabbing number is $\Theta(\sqrt{n})$.

Our hardness result puts forward two interesting questions. First, is it decidable whether a rectilinear polygon admits a conforming partition with stabbing number at most 2 or 3 in polynomial time? Second, are there other natural parameters for FPT algorithms to compute optimal conforming partitions? For the former, we give an $O(n \log n)$-time algorithm to decide whether a conforming partition with stabbing number 2 exists (Section 4); this leaves the case of stabbing number 3 open. For the latter, we give two

FPT algorithms to test whether a polygon P has conforming stabbing number at most k (Section 5). One is parameterized by the sum of k plus the *treewidth* of P, the other is specific to simple polygons in *general position* and is parameterized by k alone.

2 Preliminaries

Throughout the article, the polygons we consider are all rectilinear (i.e., the edges are axis-aligned) and may contain holes. A polygon is in *general position* if no three vertices lie on one axis-aligned line (Figure 1(c)). A polygon is *thin* if no pair of its reflex segments intersect (Figure 1(d)-(e)).

The *pixelation* of a polygon P (possibly with holes) is the partition of P obtained by adding for each reflex vertex p its horizontal and vertical reflex segments; these segments are denoted $\mathbf{h}_p$ and $\mathbf{v}_p$ (Figure 1(c)-(d)). A *pixel* is a maximal region of P that does not intersect a reflex segment. For a reflex vertex p of P, the *wedge-pixel of* p is the pixel incident to the wedge defined by the reflex segments of p, i.e., the pixel that is incident to p and to $\mathbf{h}_p$ and $\mathbf{v}_p$ (Figure 1(c)).

Recall that a *stabbing segment* of a rectilinear polygon P is an axis-aligned line segment that lies in the interior of P; for purposes of the stabbing number we only need to consider segments of maximal length, and we consider them to be open segments. We say that two stabbing segments are *equivalent* if they intersect the same set of pixels; there are $O(n)$ equivalence classes of stabbing segments. For instance, in Figure 1(c), there are 26 equivalence classes. Given a rectilinear polygon P, by k-STAB (k-CSTAB) we denote the problem of deciding whether P admits a partition (conforming partition) into rectangles such that all stabbing segments intersect at most k rectangles.

We will reduce from an $\mathcal{NP}$-hard problem called *rectilinear planar monotone* 3-SAT (RPM-3-SAT) [8] to prove the hardness results. The RPM-3-SAT problem is a variant of 3-SAT where every clause is either negative or positive, i.e., consists of either three negated or three non-negated variables. Furthermore, the bipartite graph constructed from the variable-clause incidences admits a planar drawing such that all vertices are drawn as rectangles, the *variable rectangles* (i.e., rectangles of vertices corresponding to variables) lie along the x-axis, the *positive (negative) clause rectangles* (i.e., rectangles of vertices corresponding to such clauses) lie above (below) the x-axis, and edges are represented by vertical lines of visibility between the rectangles of their endpoints. Figure 2(a) illustrates such an instance where the rectangles are shaded in gray.

3 Intractability of Stabbing Number 4 or More

In this section, we sketch a proof of $\mathcal{NP}$-completeness; details are in Appendix A.

Theorem 3.1. *For all integer $k \geq 4$, the decision problems k-STAB and k-CSTAB are $\mathcal{NP}$-complete. Moreover, k-CSTAB remains $\mathcal{NP}$-complete even for thin polygons and for polygons in general position.*

Proof structure. It is straightforward to verify that k-STAB and k-CSTAB are in $\mathcal{NP}$ for any integer k. We thus concentrate on proving $\mathcal{NP}$-hardness. First, we prove that 4-CSTAB is $\mathcal{NP}$-hard, even if only thin polygons are considered. In a thin polygon, any optimal partition is conforming, so in consequence 4-STAB is also $\mathcal{NP}$-hard. However, the gadgets take advantage of not being in general position. Second, we provide an alternative version of this proof, this time for polygons in general position (but the gadgets take advantage of not being thin). As an aside, we note here that it is not possible to make the gadgets both thin and in general position; the problem is actually polynomial in this case, see Theorem 4.1.

As a third step, using a similar approach and with a similar alternative version, we prove that 5-CSTAB, and thus 5-STAB, are $\mathcal{NP}$-hard. Finally, we show how to modify our constructions for ℓ-CSTAB ($\ell \in \{4, 5\}$) to work for $\ell + 2m$-CSTAB for any $m \geq 1$. Therefore k-CSTAB is $\mathcal{NP}$-hard for thin polygons for all $k \geq 4$, which implies hardness for k-STAB.

Proof sketch of the $\mathcal{NP}$-hardness of 4-CSTAB for thin polygons. We reduce RPM-3-SAT (defined in the preliminaries) to 4-STAB in polynomial time. We transform an instance ϕ of RPM-3-SAT (shaded in the background of Figure 2(a)) into an instance $P(\phi)$ of 4-STAB (the polygon in Figure 2(a)).

The polygon $P(\phi)$ consists of *variable gadgets* (drawn inside the variable rectangles), *split gadgets* (drawn above and/or below the variable gadgets and still inside the variable rectangles), and *clause gadgets* (drawn inside the clause rectangles). Crucial to our construction are *forcer gadgets*, indicated by a square labeled F in Figure 2(a) and shown in detail in Figure 2(b). A forcer gadget is designed to force the presence (in any conforming partition with stabbing number at most 4) of a certain pair of reflex segments in the pixel to which it is attached.

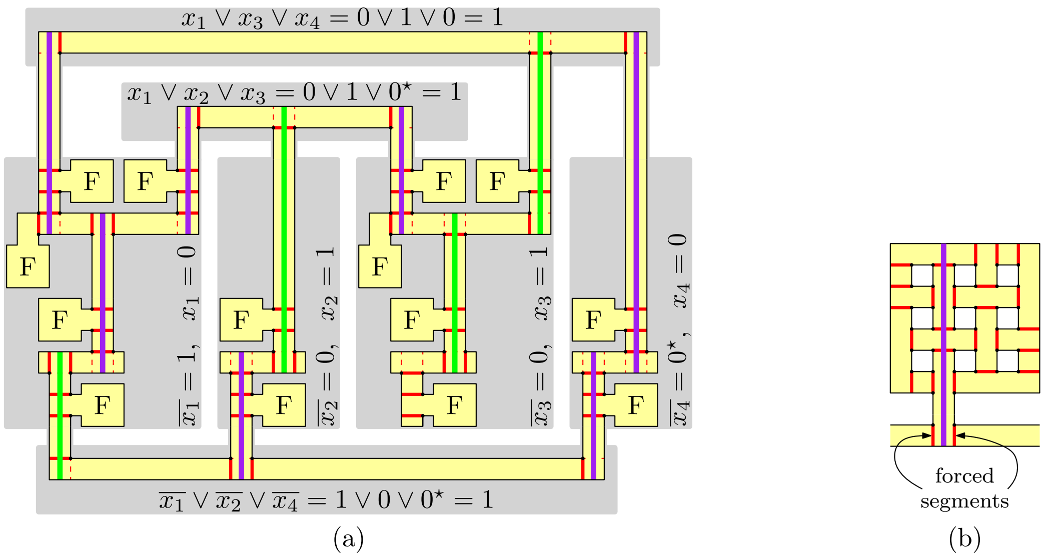


Figure 2: Gadgets used in the proof of Theorem 3.1 for $k = 4$ (using thin polygons). (a) The polygon $P(\phi)$ (not to scale) of the RPM-3-SAT drawing of $\phi = (x_1 \vee x_3 \vee x_4) \wedge (x_1 \vee x_2 \vee x_3) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_4})$. Forcer gadgets are represented by squares labeled F. The reflex segments of a partition with stabbing number 4 are solid bold (in red), the other reflex segments are dotted (in red). Vertical stabbing segments propagating 0 (respectively 1) are thick purple (respectively thick green). We use $0^\star$ for a value that is 1 in the variable assignment but that has been decreased by a variable gadget or by a split gadget and is propagated as 0. (b) The forcer gadget.

We now describe the properties of the gadgets and show at the same time that any conforming partition $\mathcal{R}$ of $P(\phi)$ implies a satisfying assignment for ϕ. Without loss of generality, $\mathcal{R}$ is *minimal*, i.e., no reflex segment can be removed while retaining a conforming partition.

We propagate information between gadgets along certain vertical stabbing segments that each intersect two gadgets. We say that such a stabbing segment $\mathbf{s}$ *propagates* 0 (standing for 'false') if it intersects three segments of $\mathcal{R}$ within the gadget that lies closer to the x-axis, and that $\mathbf{s}$ propagates 1 (standing for 'true') if it intersects only two segments of $\mathcal{R}$.

In the gadget of a variable x, there are two *out-stabs*, i.e., two vertical stabbing segments, which are assigned to the literals x and $\overline{x}$. We design variable gadgets such that not both out-stabs propagates 1, but all other combinations of propagated values are possible. We then read from the partition a value for x: x is assigned the value propagated by the out-stab of the literal x. Note that we set $x = 0$ (by convention) if both out-stabs propagate 0 (the convention $x = 1$ would have worked as well). In the partition of $P(\phi)$ in Figure 2(a), $x_1, x_2, x_3, x_4 = 0, 1, 1, 0$ (even though the out-stab of the literal $\overline{x_4}$ propagates 0).

A split gadget has an *in-stab* and two *out-stabs*. The in-stab is an out-stab of a variable gadget or of another split gadget. A split gadget "splits the propagation" in the sense that the value propagated by the two out-stabs is at most the value propagated by the in-stab. In Figure 2(a), the split gadget in x_1 splits the in-stab's value 0 into two out-stabs propagating 0 as well, whereas the split gadget in x_3 splits the in-stab's value 1 into a left out-stab propagating 0 and a right one propagating 1.

A clause gadget has three *in-stabs* each of which is an out-stab (of a variable gadget or a split gadget) propagating the value (possibly decreased) from a variable gadget. We design the clause gadget such that there exists a conforming partition where the horizontal stabbing segment within a clause gadget intersects at most four rectangles if and only if at least one of the three in-stabs of the clause gadget propagates 1. This in turn is possible only if one of the literals of the clause corresponds to an out-stab propagating 1, which in turn implies that we have assigned 1 to this literal, since propagated values do not increase. Therefore, a solution to 4-CSTAB implies a satisfying assignment for ϕ.

The other direction (i.e., proving that a satisfying assignment to ϕ gives a solution to 4-CSTAB) is similar and even easier and the reduction is hence complete. We prove that 5-CSTAB is $\mathcal{NP}$-hard using the exact same reduction idea with slightly modified gadgets.

Proof sketch for polygons in general position. In the previous reduction, we use aligned reflex vertices in two places: within the forcer gadget, and where the forcer gadget attaches at some pixel. To achieve the reduction for polygons in general position, we design a completely different forcer gadget based on a staircase, and we shift the attachment points of forcer gadgets slightly so that they are no longer aligned.

Proof sketch for $k > 4$. For the case when k is even, i.e. $k = 4 + 2m$ where m is a positive integer, we generalize the forcer gadget for stabbing number 4, by adding m rows and m columns. We then generalize the polygon $P(\phi)$ by attaching m forcer gadgets for stabbing number k to the middle of each row or column of adjacent pixels of $P(\phi)$. The hardness reduction for k-CSTAB now follows the same technique that we used to prove the hardness of 4-CSTAB. The case when k is odd is handled similarly by starting with the hardness of 5-CSTAB.

4 Tractability of Conforming Stabbing Number 2

The tractability of 2-CSTAB is very easy to show by phrasing the problem as a 2-SAT problem. The running time depends on the maximum number of reflex segments intersected by a stabbing segment.

Lemma 4.1. *There exists an algorithm that, for a rectilinear n-gon P where every stabbing segment intersects at most ℓ reflex segments, decides* 2-CSTAB *and provides a solution (if any) in $O(\ell n)$ time.*

Proof. Declare a boolean variable $x(\mathbf{s})$ for every reflex segment $\mathbf{s}$, with the intent that $\mathbf{s}$ is used in the solution if and only if $x(\mathbf{s})$ is true. To ensure that we have a conforming partition, we hence require

- $x(\mathbf{h}_p) \vee x(\mathbf{v}_p)$ for every reflex vertex p, as well as
- $\neg x(\mathbf{h}_p) \vee \neg x(\mathbf{v}_q)$ for any two intersecting reflex segments $\mathbf{h}_p, \mathbf{v}_q$.

To ensure that the conforming stabbing number is at most 2, we force that every stabbing segment intersects at most one chosen reflex segment. In other words, we require $\neg x(\mathbf{s}_1) \vee \neg x(\mathbf{s}_2)$ for any two reflex segments $\mathbf{s}_1, \mathbf{s}_2$ intersected by a common stabbing segment.

All these restrictions only involve two variables, so this gives a 2-SAT instance that has $O(n)$ variables. For every reflex segment $\mathbf{s}$, variable $x(\mathbf{s})$ belongs to at most two clauses of the first kind, and at most ℓ clauses each of the second and the third kind. So the number of clauses is $O(\ell n)$. Since 2-SAT can be solved in linear time [2], the result follows. □

In an arbitrary polygon there could be stabbing segments that intersect $\Theta(n)$ reflex segments, so the running time of the 2-SAT approach is $O(n^2)$ in the worst case. Our main contribution in this section is to give a faster algorithm, with $O(n \log n)$ running time.

We call a reflex segment *impossible* if no conforming partition with stabbing number 2 contains it, and *fixed* if any such conforming partition must contain it. The idea of our algorithm is to determine via some rules that some segments are impossible or fixed, from which we deduce other segments to be impossible or fixed. Repeated applications either provide an answer to 2-CSTAB, or end with a situation where the *undecided* segments (i.e., the ones where we did not derive that they are fixed or impossible) are in very restricted positions; we then find a conforming partition easily. We start with three obvious rules:

(R1) If, at some reflex vertex p, both reflex segments are impossible, then there is no conforming partition.

(R2) If, at some reflex vertex p, one reflex segment is impossible, then the other one is fixed.

(R3) If a stabbing segment $\mathbf{s}$ intersects a fixed segment, then all other reflex segments intersected by $\mathbf{s}$ are impossible.

Two non-trivial rules, (R4) and (R5), which trigger the entire process, are in the following lemmas:

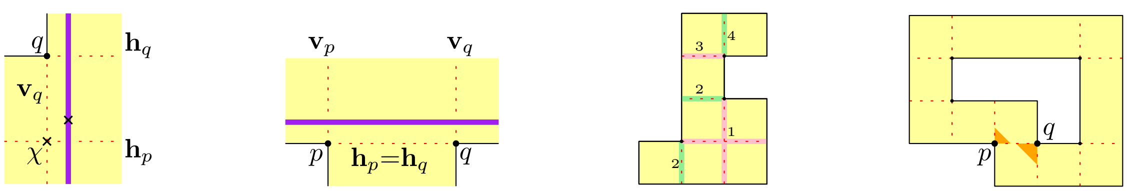
Lemma 4.2 (R4)**.** *Let $\mathbf{h}_p$ and $\mathbf{v}_q$ be a pair of horizontal and vertical reflex segments that intersect at a point interior to both. Then $\mathbf{h}_p$ and $\mathbf{v}_q$ are impossible.*

Proof. Assume for contradiction that some conforming partition used $\mathbf{h}_p$ (the argument is similar for $\mathbf{v}_q$). Then we cannot use $\mathbf{v}_q$ (since partition segments must not intersect), so must use $\mathbf{h}_q$. Let χ be the common point of $\mathbf{h}_p$ and $\mathbf{v}_q$. Up to symmetry, we may assume that the wedge-pixel of q lies to the right of $\mathbf{v}_q$. Then for small enough ε, the vertical stabbing segment through $\chi + (\varepsilon, \varepsilon)$ intersects both $\mathbf{h}_p$ and $\mathbf{h}_q$ (see also Figure 3(a)) and the conforming partition has stabbing number 3 or more. □

To explain (R5) we need a definition. A *gate* of a polygon is an axis-aligned segment $\overline{pq}$ that connects two reflex vertices p, q such that the wedge-pixels of p and q lie on the same side of $\overline{pq}$. Figure 3(b) shows a gate, while segment $\overline{pq}$ in Figure 3(d) is not a gate since the wedge-pixels are not on the same side of $\overline{pq}$.

Lemma 4.3 (R5)**.** *Any gate is fixed.*

Proof. Up to symmetry we may assume that gate $\overline{pq}$ is horizontal, so $\overline{pq} = \mathbf{h}_p = \mathbf{h}_q$. Since the wedge-pixels lie on the same side of $\overline{pq}$, we may assume up to symmetry that $\mathbf{v}_p$ and $\mathbf{v}_q$ both go upward from p and q. The horizontal stabbing segment through $p + (0, \varepsilon)$ (for a small enough ε) then intersects both $\mathbf{v}_p$ and $\mathbf{v}_q$ since it runs parallel to $\mathbf{h}_p = \mathbf{h}_q$ (see also Figure 3(b)). Thus, by (R3), any conforming partition with stabbing number 2 does not include both $\mathbf{v}_p$ and $\mathbf{v}_q$, which means by (R2) that the segment $\overline{pq}$ is included instead. □



(a) (b) (c) (d)

Figure 3: (a) (R4) illustrated. (b) (R5) illustrated. (c) An example of the propagation. Both segments at 1 are impossible by (R4). This fixes the two segments labeled 2 by (R2). This makes the segment labeled 3 impossible by (R3), which in turn fixes the segment labeled 4 by (R2). (d) An example where no reflex segments get fixed. There is a horizontal segment connecting two reflex vertices p, q, but it is not a gate.

Recall that our approach is to apply the above rules, and to keep track (by storing them in two lists L_{fixed} and $L_{\text{impossible}}$) of all reflex segments that we determine to be fixed or impossible. (There may be other fixed or impossible segments that we do not find.) This clearly can be done in polynomial time; we show in Appendix B how to implement it in $O(n \log n)$ time by applying line-sweep and ray-shooting techniques. If some segment $\mathbf{s}$ belongs to both L_{fixed} and $L_{\text{impossible}}$, then we conclude that there is no conforming partition with stabbing number 2.

We are left with three possible outcomes: We find that there is no conforming partition with stabbing number at most 2, or L_{fixed} defines a conforming partition, or neither. We are done in the first outcome. We are also done in the second outcome: If L_{fixed} defines a conforming partition, then by (R3) (and since L_{fixed} is disjoint from $L_{\text{impossible}}$) every stabbing segment intersects at most one segment of L_{fixed}, and so the stabbing number is 2. In the third outcome, we provide an algorithm to test in linear time whether there exists a solution. (In fact, there *always* is a solution, but for space reasons we do not prove this.)

Add the segments of L_{fixed} into P to obtain a partition of P into rectilinear polygons $P_1, \dots, P_\ell$ that we call the *pieces* of P. The idea is now to solve the problem for each piece of P and to put the solutions together. Next, we make two useful observations.

Observation 1. *For every piece P_i, every reflex segment $\mathbf{s}$ of P_i is a reflex segment of P that was undecided (i.e., neither in L_{fixed} nor in $L_{\text{impossible}}$).*

Proof. Since $\mathbf{s}$ is a reflex segment of P_i, one endpoint of $\mathbf{s}$ is a reflex vertex of P_i, hence also a reflex vertex of P. The other endpoint of $\mathbf{s}$ lies on the boundary of P_i. If this other endpoint were not on P, then it would be on the interior of a segment $\mathbf{s}' \in L_{\text{fixed}}$. But then rule (R4) would have been applied to $\mathbf{s}'$ and the reflex segment of P containing $\mathbf{s}$. This would have added $\mathbf{s}'$ to $L_{\text{impossible}}$, contradicting that L_{fixed} and $L_{\text{impossible}}$ are disjoint. Thus the other endpoint of $\mathbf{s}$ also lies on the boundary of P, and $\mathbf{s}$ is a reflex segment of P.

To see that $\mathbf{s}$ is undecided, observe first that p does not have an incident reflex segment in L_{fixed} since it is reflex in the piece P_i. Thus $\mathbf{s} \notin L_{\text{fixed}}$, and also $\mathbf{s} \notin L_{\text{impossible}}$, since otherwise rule (R2) would have added the other reflex segment at p to L_{fixed}. Therefore $\mathbf{s}$ is undecided. □

Observation 2. *P has a solution to* 2-CSTAB *if and only if each of the pieces $P_1, \dots, P_\ell$ of P has a solution to* 2-CSTAB.

Proof. Any solution for P includes all segments of L_{fixed}, thereby yielding a solution for each piece. Vice versa, assume that each piece P_i of P admits a solution $\mathcal{R}_i$ of reflex segments to 2-CSTAB. We show that $\mathcal{R} := L_{\text{fixed}} \cup \bigcup_i \mathcal{R}_i$ is a solution for P. To see that $\mathcal{R}$ is a conforming partition, observe that it only contains reflex segments of P by Observation 1, and assigns at least one reflex segment to each reflex vertex of P. Since the pieces $P_1, \dots, P_\ell$ are interior-disjoint, the reflex segments in $\bigcup_i \mathcal{R}_i$ do not intersect each other. They do not intersect a segment of L_{fixed} either, by Observation 1, so $\mathcal{R}$ yields a conforming partition.

To show that $\mathcal{R}$ has stabbing number at most 2, consider any stabbing segment $\mathbf{s}$ of P. If $\mathbf{s}$ intersects no segment of L_{fixed}, then it is also a stabbing segment for one piece P_i, and so will intersect at most one segment of $\mathcal{R}$. Now assume that $\mathbf{s}$ intersects a segment of L_{fixed}. Since rule (R3) was applied, all other reflex segments of P intersected by $\mathbf{s}$ were added to $L_{\text{impossible}}$, so were not reflex segments of any pieces, and hence are not used by $\mathcal{R}$. Therefore, stabbing segment $\mathbf{s}$ intersects at most one segment of $\mathcal{R}$. □

It remains to show how to solve the problem for each piece efficiently. Here, our previous 2-SAT approach comes to the rescue since the pieces are not arbitrary polygons. Specifically, since rule (R4) does not apply to piece P_i (for $i \in \{1, \dots, \ell\}$), it has no intersecting reflex segments, so it is thin. Since rule (R5) does not apply to P_i, it has no gate. We now prove a statement that holds for any thin gate-free polygon.

Lemma 4.4. *Let P be a thin rectilinear polygon that has no gate. Then every stabbing segment $\mathbf{s}$ intersects at most two reflex segments.*

Proof. Assume for contradiction that $\mathbf{s}$ intersects three reflex segments, say $\mathbf{s}$ intersects $\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3$, in this order and with no other reflex segments in between. See also Figure 4. Up to symmetry $\mathbf{s}$ is horizontal, so $\mathbf{s_1}, \mathbf{s}_2, \mathbf{s_3}$ are vertical, and up to renaming $\mathbf{s_1}$ is on the left of $\mathbf{s_2}$. Let p be the reflex vertex of P with $\mathbf{s}_2 = \mathbf{v}_p$.

Up to symmetry, the wedge-pixel ξ of p is to the left of $\mathbf{v}_p$ and below $\mathbf{h}_p$. Since P is thin, pixel ξ extends the entire length of $\mathbf{v}_p$, and in particular includes the point common to $\mathbf{s}$ and $\mathbf{v}_p$. Since there are no vertical reflex segments between $\mathbf{s}_1$ and $\mathbf{s}_2$ along $\mathbf{s}$, pixel ξ extends to the point common to $\mathbf{s}_1$ and $\mathbf{s}$, and therefore the entire length of $\mathbf{s}_1$. It also includes the entire length of $\mathbf{h}_p$. Thus the top left corner of ξ is a point q common to $\mathbf{s}_1$ and $\mathbf{h}_p$, hence q is a reflex vertex that lies on a horizontal line with p. Since both $\mathbf{v}_q$ and $\mathbf{v}_q$ bound sides of ξ, this makes $\overline{pq}$ a gate. □

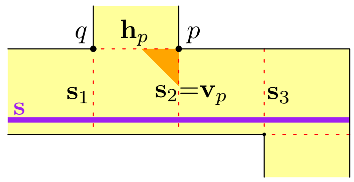


Figure 4: If stabbing segment **s** intersects three reflex segments, and no two reflex segments intersect, then the polygon is not in general position. The upper left corner of the wedge-pixel ξ of p (and q) is shaded (in orange).

Lemma 4.4 has two consequences:

Theorem 4.1. *There exists an algorithm that, for a rectilinear n-gon P that is thin and in general position, computes the stabbing number in $O(n)$ time.*

Proof. The stabbing number of P is 1 if P is a rectangle, and at least 2 otherwise. A polygon in general position has no gates, hence any stabbing segment of P intersects at most three reflex segments by Lemma 4.4. So the stabbing number of P (which is equal to the conforming stabbing number since P is thin) is either 2 or 3. Lemma 4.1 gives an algorithm to test whether it is 2 in $O(n)$ time. □

Theorem 4.2. *There exists an algorithm that, for any rectilinear n-gon P, decides* 2-CSTAB *and provides a solution (if any) in $O(n \log n)$ time.*

Proof. Apply all rules; this takes $O(n \log n)$ time. Then compute the pieces $P_1, \dots, P_\ell$ in $O(n)$ time. Using Lemma 4.4 and Theorem 4.1, test in $O(|P_i|)$ time whether a piece P_i has a solution for 2-CSTAB. By Observation 2, this information is enough to decide 2-CSTAB for P and compute the solution in case of an affirmative answer. Since $\sum_i |P_i| \in O(n)$, the result follows. □

5 Polygons with Small Treewidth

We now turn towards FPT algorithms, and in particular, study polygons with bounded treewidth. We recall first a few definitions. A *tree decomposition* of a graph $G = (V, E)$ is a tree $\mathcal{T}$ and an assignment β from the nodes of $\mathcal{T}$ to subsets of V (called *bags*) with the following properties: (a) For every vertex v of G, the bags containing v form a non-empty connected subtree of $\mathcal{T}$. (b) For every edge e of G, there exists a bag that contains both endpoints of e. The *width* of a tree decomposition is the maximum bag-size minus one, and the *treewidth* $tw(G)$ of G is the minimum width of a tree decomposition of G.

The treewidth has frequently been used for FPT algorithms for graph problems, but can also be used for solving problems on polygons, see e.g. [3]. Recall that the *pixelation* of a polygon P is obtained by inserting all reflex segments.

This gives rise to a planar graph (the *pixelation graph* G_P by replacing every crossing with a new vertex of degree 4 and every endpoint of a reflex segment on an edge of P with a vertex of degree 3 (see Figure 5(a)). The *treewidth* of P is the treewidth $tw(G_P)$ of the pixelation graph.

Our algorithm for polygons with small treewidth uses not only the pixelation graph, but also its *radial graph* R_P and defined as follows. The vertices of R_P are the vertices of G_P (we denote them by V_P), as well as one vertex for every pixel (we denote these by Ξ_P). We add an edge (ξ, v) between $\xi \in \Xi_P$ and $v \in V_P$ if and only if vertex v is incident to pixel ξ. See Figure 5(b). Using the techniques of Borradaile et al. [6], one can easily show that R_P has treewidth $O(tw(G_P))$, since pixels are incident to four vertices of G_P.

So we now show how to exploit small treewidth of R_P to find the conforming stabbing number of P. To this end, we use Courcelle's theorem [7], which states that if a graph property can be expressed in monadic second-order logic (MSOL) as a formula ϕ, then testing whether a graph G with a tree decomposition of width w satisfies the property can be done in time that is linear in the number of vertices and fixed-parameter

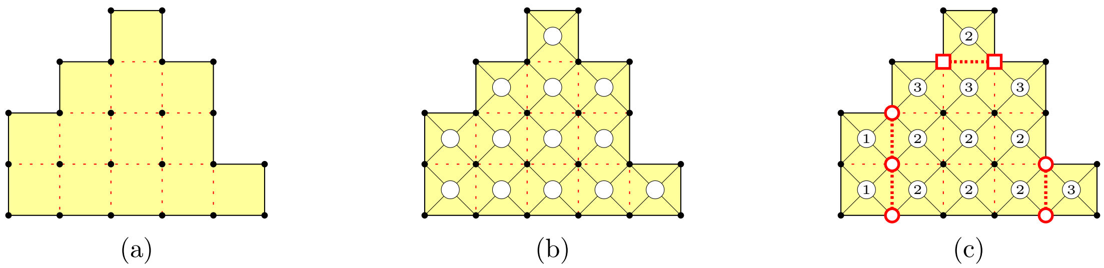


Figure 5: (a) A polygon P with its pixelation graph G_P. (b) The radial graph R_P contains the thin solid edges. (c) One possible solution to the MSOL formula ϕ for $k = 3$. We only show parts of this solution: Partition segments are dotted, bold, and red. Vertices in $\mathcal{V}$ ($\mathcal{H}$) are bold hollow red circles (squares). Vertices in Ξ_i^{hor} are indicated by ⓘ.

tractable in $|\phi| + tw(G)$. To express the conforming stabbing number of P via MSOL, we use the following ideas illustrated in Figure 5(c):

- We have vertex sets $\mathcal{V}$ and $\mathcal{H}$, with the intended meaning that these are the vertices of G_P that lie on vertical/horizontal reflex segments used by a conforming partition.

  With easy formulas that only rely on adjacencies of graph R_P, we can express that $\mathcal{V}$ and $\mathcal{H}$ indeed correspond to reflex segments of the appropriate orientation, and that we have a conforming partition: every reflex vertex of P belongs to at least one of these vertex sets, and no reflex segments intersect. (See Appendix C for details.)

- We partition Ξ_P into $\Xi_1^{\text{hor}} \cup \cdots \cup \Xi_k^{\text{hor}}$, with the intended meaning that if $\xi \in \Xi_P$ belongs to Ξ_i^{hor}, then the horizontal stabbing segment through ξ, when traversed left-to-right, has encountered at most i rectangles when it reaches ξ. Since we require $i \leq k$, this enforces that all horizontal stabbing segments hit at most k rectangles.

  With easy formulas that only rely on adjacencies of graph R_P, we can express that indices of the sets Ξ_i^{hor} indeed express rectangle-counts. Namely, if ξ, ξ' are two pixels that share a vertical edge (say with ξ left), and $\xi \in \Xi_i^{\text{hor}}$, then we require $\xi' \in \Xi_{i+1}^{\text{hor}}$ or $\xi' \in \Xi_i^{\text{hor}}$ depending to whether the segment corresponding to the shared edge is in the partition or not.

- Symmetrically we can force that all vertical stabbing segments hit at most k rectangles.

The length of the resulting formula is linear in k and independent of the size of graph R_P. If polygon P has n vertices, then G_P has $O(n^2)$ vertices, and so does R_P. Therefore, with Courcelle's theorem, we obtain the first FPT algorithm.

Theorem 5.1. *There exists an algorithm that, for a rectilinear n-gon P with treewidth ℓ, decides k-CSTAB in $O(f(k,\ell)n^2)$ time, for some function $f(\cdot)$ that does not depend on n.*

The function $f(\cdot)$ that falls out of Courcelle's theorem is rather large (it could be a tower of exponents). It is possible to decide k-CSTAB directly by doing bottom-up dynamic programming in a tree decomposition of R_P of minimum width ℓ' (which we know to be in $O(\ell)$). Each pixel needs to keep track of which of the sets $\Xi_1^{\text{hor}}, \ldots, \Xi_k^{\text{hor}}, \Xi_1^{\text{ver}}, \ldots, \Xi_k^{\text{ver}}$ it belongs to, and each vertex of V_P needs to keep track whether it is in $\mathcal{V}$ or $\mathcal{H}$. Since bags contain up to $\ell' + 1$ vertices, this gives at most $k^{2\ell'+2}$ possible configurations per bag, and with (not difficult but tedious to write) update-formulas one can therefore show how to solve k-CSTAB in $O(k^{2\ell'+2}n^2)$ time. We leave the details as an exercise. It is also not hard to modify the MSOL formulations so that it permits arbitrary partitions, rather than restricting to conforming ones. In other words, k-STAB is also fixed-parameter tractable in $k + tw(G_P)$. Details are also left as an exercise.

Now we give a second FPT algorithm, which makes a different assumption on the polygon P. We require P to be simple and to have no gates (the latter holds in particular if P is in general position), but in exchange we no longer need to bound the treewidth. The idea for this theorem is to distinguish by the maximum number of reflex segments intersected by a stabbing segment; if it is small then the treewidth is small and Theorem 5.1 applies, and if it is large enough then (as one shows) the conforming stabbing number is bigger than k. Details are in Appendix D.

Theorem 5.2. *There exists an algorithm that, for a simple gate-free rectilinear n-gon P , decides k-CSTAB in $O(f'(k)n^2)$ time, for some function $f'(\cdot)$ that does not depend on n.*

6 Conclusion

In this paper, we show that computing a conforming partition of a rectilinear polygon with stabbing number k is $\mathcal{NP}$-hard for all $k \geq 4$. Since the reduction uses only thin polygons, the hardness result follows even if we omit the conforming constraint. The polygons used in our reduction have holes. Therefore, determining the time complexity of computing an optimal (conforming) partition for simple polygons (i.e., without holes) remains open.

On the positive side, we provide an $O(n \log n)$-time algorithm to decide whether a polygon admits a conforming partition with stabbing number 2. Since the problem is $\mathcal{NP}$-hard already for conforming stabbing number 4, only the case of conforming stabbing number 3 remains open. For polygons (possibly with holes) with bounded treewidth and bounded conforming stabbing number, we give a quadratic-time algorithm to compute the minimum stabbing number. An exciting direction would be to design fixed-parameter tractable algorithms for simple polygons parameterized by the conforming stabbing number, which would complement the hardness result for polygons with holes. Interestingly, for simple polygons that are in general position, we already gave such a fixed-parameter-tractable algorithm. But we also proved that general position does not help the case of polygons with holes: computing a conforming partition with stabbing number at most k (for $k \geq 4$) remains $\mathcal{NP}$-hard for polygons in general position.

Extending all these results to higher dimensions would be interesting, even for the restricted class of orthogonal 3D-histograms where previous results focus on minimizing the number of partitions into rectangular boxes [4, 13].

References

[1] Mohammad Ali Abam, Boris Aronov, Mark de Berg, and Amirali Khosravi. Approximation algorithms for computing partitions with minimum stabbing number of rectilinear and simple polygons. In *Proceedings of the 27th ACM Symposium on Computational Geometry (SCG)*, pages 407–416, 2011. doi:10.1145/1998196.1998263.

[2] Bengt Aspvall, Michael F Plass, and Robert Endre Tarjan. A linear-time algorithm for testing the truth of certain quantified boolean formulas. *Information processing letters*, 8(3):121–123, 1979.

[3] T. Biedl and S. Mehrabi. On orthogonally guarding orthogonal polygons with bounded treewidth. *Algorithmica*, 83:641–666, 2021. doi:10.1007/s00453-020-00769-5.

[4] Therese Biedl, Martin Derka, Veronika Irvine, Anna Lubiw, Debajyoti Mondal, and Alexi Turcotte. Partitioning orthogonal histograms into rectangular boxes. In *Proceedings of the 13th Latin American Symposium on Theoretical Informatics (LATIN)*, pages 146–160. Springer, 2018.

[5] Hans L Bodlaender. Dynamic programming on graphs with bounded treewidth. In *Automata, Languages and Programming*, pages 105–118, 1988. doi:10.1007/3-540-19488-6_110.

[6] Glencora Borradaile, Jeff Erickson, Hung Le, and Robbie Weber. Embedded-width: A variant of treewidth for plane graphs. *CoRR*, abs/1703.07532, 2017. URL: http://arxiv.org/abs/1703.07532, arXiv:1703.07532.

[7] Bruno Courcelle. The monadic second-order logic of graphs. I. Recognizable sets of finite graphs. *Information and computation*, 85(1):12–75, 1990.

[8] Mark de Berg and Amirali Khosravi. Optimal binary space partitions for segments in the plane. *International Journal of Computational Geometry and Applications*, 22(03):187–205, 2012. `doi:10.1142/S0218195912500045`.

[9] Mark de Berg and Marc van Kreveld. Rectilinear decompositions with low stabbing number. *Information Processing Letters*, 52(4):215–221, 1994. `doi:10.1016/0020-0190(94)90129-5`.

[10] E. Demaine, F. Fomin, M. Hajiaghayi, and D. Thilikos. Subexponential parameterized algorithms on bounded-genus graphs and H-minor-free graphs. *J. ACM*, 52(6):866–893, 2005. `doi:10.1145/1101821.1101823`.

[11] Stephane Durocher and Saeed Mehrabi. Erratum to: Computing partitions of rectilinear polygons with minimum stabbing number. URL: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=0093663d76ea6411084216000c1c61b7c0b0f49d.

[12] Stephane Durocher and Saeed Mehrabi. Computing conforming partitions of orthogonal polygons with minimum stabbing number. *Theoretical Computer Science*, 689:157–168, 2017. `doi:10.1016/j.tcs.2017.05.035`.

[13] Peter Floderus, Jesper Jansson, Christos Levcopoulos, Andrzej Lingas, and Dzmitry Sledneu. 3D rectangulations and geometric matrix multiplication. *Algorithmica*, 80:136–154, 2018.

[14] Y. Giyora and H. Kaplan. Optimal dynamic vertical ray shooting in rectilinear planar subdivisions. *ACM Trans. Algorithms*, 5(3):28:1–28:51, 2009. `doi:10.1145/1541885.1541889`.

[15] John Hershberger and Subhash Suri. A pedestrian approach to ray shooting: Shoot a ray, take a walk. *Journal of Algorithms*, 18(3):403–431, 1995. `doi:10.1006/jagm.1995.1017`.

[16] Breno Piva and Cid C de Souza. Minimum stabbing rectangular partitions of rectilinear polygons. *Computers & Operations Research*, 80:184–197, 2017.

A Proof of Theorem 3.1

In this section, we provide a detailed proof of Theorem 3.1.

The following definition is used to shorten the writing of some frequent terms in the proofs. The vertical (respectively horizontal) reflex segments of P on the boundary of a pixel ξ are called the *verticals* (respectively *horizontals*) of ξ. We also use the following lemmas.

Lemma A.1. *The stabbing number of a thin polygon (possibly with holes) P is equal to the conforming stabbing number of P.*

Proof. Let $\mathcal{R}$ be a rectangular partition of P with a Steiner point q. We show that removing q and merging some adjacent rectangles yields a rectangular partition of P.

By definition of a thin polygon, q is not the intersection of two reflex segments of P. Thus, q is adjacent to at least one segment r of $\mathcal{R}$ which is not a reflex segment of P. Now, r is necessarily a full edge (and not only portions of an edge) shared by two rectangles of $\mathcal{R}$, because otherwise, P cannot be thin. Therefore, merging these two rectangles yields a rectangle. Repeating this process for all segments r eventually removes the q from the partition but does not increase the stabbing number of the partition. □

Lemma A.2. *Let P be a polygon and ξ be a pixel of P such that ξ is the wedge-pixel of any of its corners that is a reflex segment of P. Then a minimal conforming partition of P either includes the vertical reflex segments of ξ (and excludes the horizontal reflex segments of ξ), or includes the horizontal reflex segments of p (and excludes the vertical reflex segments of ξ).*

Proof. If $k = 2$, then there is only one reflex vertex. Hence the lemma is a direct consequence of the definition of a minimal conforming partition.

If $k = 3$, then ξ has two reflex vertices r, r' on the boundary that are adjacent. If we take the reflex segment between r, r', then we do not need the remaining two reflex segments as the partition is minimal. If we do not take the reflex segment between r, r', then to cover these reflex vertices, we must take the two other parallel reflex segments.

If $k = 4$, then there are four reflex vertices on the boundary of ξ. There are two ways to cover the four reflex vertices using two reflex segments: either to take the horizontals or the verticals. If we use three reflex segments, then at least one of them would be unnecessary for a minimal conforming partition. □

A.1 The Problem RPM-3-SAT to Be Reduced

Before defining problem RPM-3-SAT, we recall some terminology. The *graph* of a 3-CNF formula ϕ is the undirected bipartite graph G_ϕ defined as follows. The vertex set is the disjoint union of the set of variables of ϕ and of the set of clauses of ϕ. There is an edge between a variable x and a clause c if x is a variable of c in ϕ.

If a 3-CNF formula ϕ is *monotone*, that is to say, if any clause of ϕ contains either three positive literals or three negative literals, then a drawing D of the graph of ϕ is *rectilinear planar monotone* (or *RPM*) if the following holds (see e.g., Figure 6).

- The variable vertices are drawn with axis-aligned rectangles centered on the x-axis.
- The positive clauses are drawn with axis-aligned rectangles above the x-axis.
- The negative clauses are drawn with axis-aligned rectangles below the x-axis.
- The edges are drawn with open axis-aligned rectangles between the corresponding variable and clause rectangles.
- All of the variable rectangles, clause rectangles, and edge rectangles are pairwise disjoint.

Problem 1 (RPM-3-SAT)**.** Input: *A rectilinear planar monotone drawing of a 3-CNF formula ϕ.* Output: *Accept if ϕ is satisfiable. Reject otherwise.*

Problem 1 is known to be $\mathcal{NP}$-hard [8].

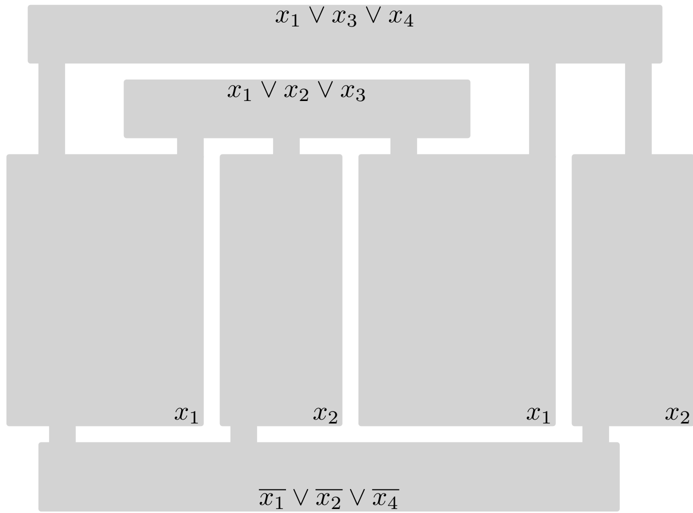


Figure 6: A RPM-3-SAT drawing of $\phi = (x_1 \vee x_3 \vee x_4) \wedge (x_1 \vee x_2 \vee x_3) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_4})$.

A.2 The Forcer Gadget for $k = 4$ Using Thin Polygons

Overview. A *forcer gadget* (Figure 7(a)) is a thin polygon similar to a 4×4 grid, with an extension to attach it to the rest of the polygon through an edge called *connection edge*. This is the only edge that is not drawn in the bootmmost row of Figure 7(a). By an *out-stab* of the force gadget we denote a segment that starts at a boundary point and leaves the forcer gadget after perpendicularly intersecting the connection edge, e.g., the thick segment of Figure 7(c) and (d). We now give the formal details.

Details of a Forcer gadget. Let F_0 be the polygon with holes defined as follows (see also Figure 7(a)). The coordinates of the vertices of F_0 in counterclockwise order along the outer boundary are:

$$((3,0),(3,1),(7,1),(7,8),(0,8),(0,1),(2,1),(2,0)).$$

The set of holes of F_0 is composed of 9 squares and is described next:

$$\{((a,b),(a+1,b),(a+1,b+1),(a,b+1)) : (a,b) \in \{1,3,5\} \times \{2,4,6\}\}.$$

An rectilinear polygon F is a *forcer gadget* if there exists a transformation τ such that τ is the composition of a translation with a rotation with angle in $\{0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}\}$ and such that $F = \tau(F_0)$. Next, we give names to some segments of interest of F.

- The edge $\tau((2,0)(3,0))$ is the *connection edge* of F (the only segment of the outer boundary of F_0 which is not drawn in Figure 7).
- The stabbing segment $\tau((2.5,8)(2.5,0))$ is the *out-stab* of F (the thick segment drawn with an arrow pointing outside F_0 in Figure 7(c) and (d)).

Lemma A.3. *Let F be an arbitrary forcer gadget. Then the following holds.*

(a) The out-stab of F intersects at least 3 reflex segments in any conforming partition with stabbing number at most 4 of F.

(b) The forcer gadget F admits a partition with stabbing number at most 4.

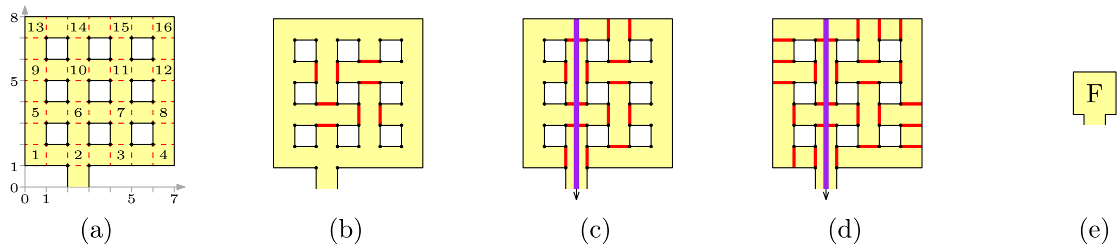


Figure 7: (a) The generic forcer gadget F_0 is shaded (in yellow). The connection segment is the only segment of the boundary of F_0 which is not solid (black). The reflex vertices are small (black) disks and the reflex segments are dotted (in red). The pixels of the reflex vertices are numbered from 1 to 16 as defined in the proof of Lemma A.3.
(b) The horizontals of pixels $6, 11$ and verticals of pixels $7, 10$ are solid (red) segments. They are one of the two possibilities for a minimal conforming partition of F_0 with stabbing number at most 4 (proof of Lemma A.3(a)).
(c) The out-stab of F_0 (thick purple) ends with an arrow pointing outside F_0. The solid (red) verticals of pixels $2, 15$ and horizontals of pixels $3, 14$ are in any minimal conforming partition of F_0 with stabbing number at most 4 (proof of Lemma A.3(a)).
(d) The solid (red) reflex segments form a minimal conforming partition of F_0 with stabbing number at most 4 (proof of Lemma A.3(b)).
(e) A schematic drawing of a forcer gadget used in the following figures.

Proof. It is enough to prove that (a) and (b) hold for F_0 considering only minimal conforming partitions. Let $\mathcal{R}$ be an arbitrary minimal conforming partition of F_0 with stabbing number at most 4.

We start by naming more parts of F_0. The polygon F_0 forms four rows and four columns that we number starting at row 1 for the bottom row and at column 1 is the left-most column. For each $(a, b) \in \{0, 2, 4, 6\} \times \{1, 3, 5, 7\}$, the pixel (which is a wedge-pixel of some reflex vertex of F_0) consisting of a unit square with (a, b) as its lower left corner is numbered $\frac{a+8b-8}{2} + 1$ (Figure 7(a)). Each of these pixels is the wedge-pixel of any of its corners that is a reflex vertex of $\mathcal{R}$, justifying the implicit use of Lemma A.2 in the rest of the proof.

(a): Pixels $6, 7, 10, 11$ have 2 verticals and 2 horizontals each. Thus, among pixels $6, 7, 10, 11$, $\mathcal{R}$ includes at most one pair of verticals per row (among rows 2, 3 of F_0) and one pair of horizontals per column (among columns 2, 3 of F_0). This leaves only two options: either $\mathcal{R}$ includes the horizontals of pixels $6, 11$ and the verticals of pixels $7, 10$ (Figure 7(b)), or $\mathcal{R}$ includes the horizontals of pixels $7, 10$ and the verticals of pixels $6, 11$.

Thus, the out-stab of F_0 intersects either the horizontals of pixel 6 or of pixel 10; regardless we can *not* use the horizontals of pixel 2 and therefore must use its verticals. This in turn means that we must use the horizontals of pixel 3, which (since one of pixels 7 and 11 uses the horizontals) means that we must use the verticals of pixel 15 and the horizontals of pixel 14 (Figure 7(c)). Therefore, column 2 of F_0 has at least 3 horizontal reflex segments included in $\mathcal{R}$, which proves (a).

(b): Let $\mathcal{R}_0$ be the minimal conforming partition of F_0 whose verticals are at pixels $1, 2, 5, 7, 10, 12, 13, 14$ (Figure 7(d)). There are 3 vertical (respectively horizontal) reflex segments of $\mathcal{R}_0$ in each row (respectively column) of F_0. Thus, $\mathcal{R}_0$ has stabbing number 4, thereby proving the existence of a conforming partition of F_0 with stabbing number at most 4, hence (b). □

A.3 The Variable Gadget for $k = 4$ Using Thin Polygons

Overview. A *variable gadget* consists of two forcer gadgets that are connected to a polygon Figure 8(a) such that there exist only three possible minimal conforming partitions of the gadget with stabbing number 4, e.g., Figure 8(d)-(f). The variable gadget connects to the rest of the polygon with two connection edges, i.e., the edges omitted from the polygon boundary in Figure 8(d). The connection edge at the top is called the *positive connection edge* and the one at the bottom is called the *negative connection edge*. We define a *positive out-stab* (*negative out-stab*), which is a maximal stabbing segment that perpendicularly intersects the positive connection edge (negative connection edge). There are two conforming partitions of the gadget that will determine the truth values of the variable (Figure 8(d)-(e)). In a false (true) configuration, exactly 3 (exactly 2) reflex segments intersect the positive out-stab, and exactly 2 (exactly 3) reflex segments intersect the negative out-stab. We now describe the details.

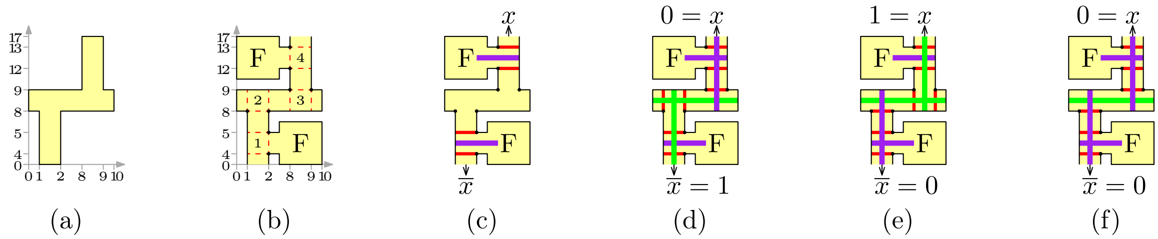


Figure 8: (a) The polygon V_0 used in the construction of the variable gadget. (b) The generic variable gadget V_1 is shaded (in yellow). The squares labeled F are forcer gadgets. The connection segments are the only segments of the boundary of V_1 which are not solid (black). The reflex vertices are small (black) disks and the reflex segments are dotted (red). The pixels are numbered from 1 to 4 as defined in the proof of Lemma A.4.
(c) The horizontals of pixels $1, 4$ are solid (red) segments. They are included in any minimal conforming partition of V_1 with stabbing number at most 4.
(d) The negative (respectively positive) out-stab of V_1ends with an arrow pointing downwards (respectively upwards) outside V_1. (A stabbing segments is green if it intersects 2 reflex segments, purple if it intersects 3 reflex segments.) The variable gadget V_1 is set to false: $\mathcal{R}_0$ includes the solid (red) horizontals of pixel 2 and verticals of pixel 3.
(e) The variable gadget V_1 is set to true: $\mathcal{R}_1$ includes the solid (red) verticals of pixel 2 and horizontals of pixel 3.
(f) The variable gadget V_1 is undetermined: $\mathcal{R}_2$ includes the solid (red) horizontals of both pixel 2 and pixel 3.

Details of a Variable Gadget. Let V_0 be the polygon without holes defined as follows (see Figure 8(a)). The coordinates of the vertices of V_0 in counterclockwise order along the boundary are:

$$((1,0), (2,0), (2,8), (10,8), (10,9), (9,9), \\ (9,17), (8,17), (8,9), (0,9), (0,8), (1,8)).$$

Let V_1 be a simple polygon defined as the union of V_0 with two forcer gadgets whose connection edges are $(2,4)(2,5)$ and $(8,12)(8,13)$. An rectilinear polygon V is a *variable gadget* if there exists a horizontal translation τ such that $V = \tau(V_1)$. Next, we give names to some segments of interest of V.

- The edge $\tau((1,0)(2,0))$ is the *negative connection edge* of V (the bottom segment of the outer boundary of V_1 which is not drawn in Figure 8).
- The edge $\tau((9,17)(8,17))$ is the *positive connection edge* of V (the top segment of the outer boundary of V_1 which is not drawn in Figure 8).
- The stabbing segment $\tau((1.5,9)(1.5,0))$ is the *negative out-stab* of V (the thick segment drawn with an arrow pointing downwards outside V_1 in Figure 8(d) and (e)).
- The stabbing segment $\tau((8.5,8)(8.5,17))$ is the *positive out-stab* of V (the thick segment drawn with an arrow pointing upwards outside V_1 in Figure 8(d) and (e)).

Lemma A.4. *Any variable gadget V admits exactly three minimal conforming partitions $\mathcal{R}_0$, $\mathcal{R}_1$, and $\mathcal{R}_2$ with stabbing number at most 4 such that the following holds (up to relabeling of $\mathcal{R}_0$, $\mathcal{R}_1$ and $\mathcal{R}_2$).*

- *Exactly* 3 *reflex segments of* $\mathcal{R}_0$ *intersect the positive out-stab of* V*, and exactly* 2 *reflex segments of* $\mathcal{R}_0$ *intersect the negative out-stab of* V*. In this case, we say that* V *is* set to false *(Figure 8(d)).*
- *Exactly* 2 *reflex segments of* $\mathcal{R}_1$ *intersect the positive out-stab of* V*, and exactly* 3 *reflex segments of* $\mathcal{R}_1$ *intersect the negative out-stab of* V*. In this case, we say that* V *is* set to true *(Figure 8(e)).*
- *Exactly* 3 *reflex segments of* $\mathcal{R}_2$ *intersect the positive out-stab of* V*, and exactly* 3 *reflex segments of* $\mathcal{R}_2$ *intersect the negative out-stab of* V*. In this case, we say that* V *is* undetermined *(Figure 8(f)).*

Proof. It is enough to prove Lemma A.4 for V_1. We start by naming more parts of V_1. The segment $(0, 8.5)(10, 8.5)$ is the *inner stab* of V_1 (the horizontal thick (green) segment in Figure 8(d) and (e)). The pixel consisting of a unit square is numbered k if its lower left corner is (Figure 8(b)):

- $(1, 4)$ and $k = 1$,
- $(1, 8)$ and $k = 2$,
- $(8, 8)$ and $k = 3$, or
- $(8, 12)$ and $k = 4$.

Let $\mathcal{R}$ be an arbitrary minimal conforming partition of V_1 with stabbing number at most 4. By Lemma A.3, $\mathcal{R}$ includes the horizontals of pixels $1, 4$ (Figure 8(c)).

The two remaining pixels are both intersected by the inner stab of V_1. Thus, there are three cases.

Case 1: $\mathcal{R}$ includes the horizontals of pixel 2 and the verticals of pixel 3 which corresponds to $\mathcal{R} = \mathcal{R}_0$ (Figure 8(d)).

Case 2: $\mathcal{R}$ includes the verticals of pixel 2 and the horizontals of pixel 3 which corresponds to $\mathcal{R} = \mathcal{R}_1$ (Figure 8(e)).

Case 3: $\mathcal{R}$ includes the horizontals of both pixel 2 and pixel 3 which corresponds to $\mathcal{R} = \mathcal{R}_2$ (Figure 8(f)).

□

A.4 The Split Gadget for $k = 4$ Using Thin Polygons

Overview. We design a split gadget to propagate the information of a variable gadget to other parts of the polygon. A split gadget consists of three forcer gadgets which are arranged such that the value of the maximal stabbing segment entering from a variable gadget into the split gadget can be propagated (either as it is, or with a decreased value) to the two stabbing segments leaving the split gadget. Figure 9 illustrates a *positive split gadget* that connects to a positive connection edge of a variable gadget. Symmetrically, we use a vertically reflected configuration for a negative connection edge, which is referred to as a *negative split gadget*. From the perspective of a split gadget, we can define *in connection edge*, *left connection edge*, and *right connection edge* that connect the split gadget to the rest of the polygon, and the corresponding perpendicular maximal stabbing segments as *in-stab*, *left out-stab* and *right out-stab*. We now discuss the details.

Details of a Split gadget. Let S_0 be the polygon without holes defined as follows (Figure 9(a)). The coordinates of the vertices of S_0 in counterclockwise order along the boundary are:

$$((13, 0), (14, 0), (14, 9), (24, 9), (24, 20), (23, 20), (23, 10),$$
$$(6, 10), (6, 20), (5, 20), (5, 10), (4, 10), (4, 9), (13, 9)).$$

Let S_1 be a simple polygon defined as the union of S_0 with three forcer gadgets whose connection edges are $(4, 9)(5, 9)$, $(6, 15)(6, 16)$, and $(23, 15)(23, 16)$. An rectilinear polygon S is a *split gadget* in the two following cases.

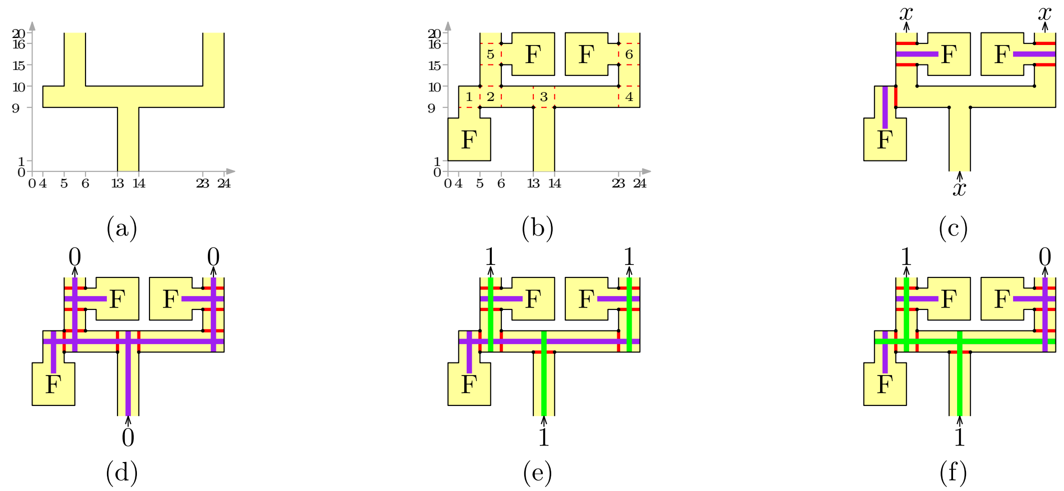


Figure 9: The figure is not to scale. (a) The polygon S_0. (b) The generic split gadget S_1 is shaded (in yellow). The squares labeled F are forcer gadgets. The connection segments are the only segments of the boundary of S_1 which are not solid (black). The reflex vertices are small (black) disks and the reflex segments are dotted (in red). The pixels are numbered from 1 to 6 as defined in the proof of Lemma A.5.
(c) The verticals of pixel 1 and the horizontals of pixels $5, 6$ are solid (red) segments. They are included in any minimal conforming partition of S_1 with stabbing number at most 4.
(d) The out-stabs (respectively in-stabs) of S_1 ends with an arrow pointing outside (respectively inside) S_1. (A stabbing segments is green if it intersects 2 reflex segments, purple if it intersects 3 reflex segments.) The split gadget S_1 "propagates false": $\mathcal{R}_0$ includes the solid (red) horizontals of pixels $2, 4$ and verticals of pixel 3.
(e) The split gadget S_1 "propagates true": $\mathcal{R}_1$ includes the solid (red) verticals of pixel $2, 4$ and horizontals of pixel 3.
(f) One of the three cases where the value propagated by some of the out-stabs (here the right out-stab) is decreased compared to the value propagated by the in-stab.

- If there exists a translation τ such that $S = \tau(S_1)$, in which case S is called a *positive* split gadget.
- If there exists a transformation τ such that τ is the composition of a horizontal reflection with a translation and such that $S = \tau(S_1)$, in which case S is called a *negative* split gadget.

Next, we give names to some segments of interest of S.

- The edge $\tau((13, 0)(14, 0))$ is the *in connection edge* of S (the bottom segment of the outer boundary of S_1 which is not drawn in Figure 9(a)).
- The edge $\tau((5, 20)(6, 20))$ is the *left connection edge* of S (the top left segment of the outer boundary of S_1 which is not drawn in Figure 9(a)).
- The edge $\tau((23, 20)(24, 20))$ is the *right connection edge* of S (the top right segment of the outer boundary of S_1 which is not drawn in Figure 9(a)).
- The stabbing segment $\tau((13.5, 0)(13.5, 10))$ is the *in-stab* of S (the thick segment drawn with an arrow pointing upwards inside S_1 in Figure 9(d)-(f)).
- The stabbing segment $\tau((5.5, 9)(5.5, 20))$ is the *left out-stab* of S (the leftmost thick segment drawn with an arrow pointing upwards outside S_1 in Figure 9(d)-(f)).
- The stabbing segment $\tau((23.5, 9)(23.5, 20))$ is the *right out-stab* of S (the rightmost thick segment drawn with an arrow pointing upwards outside S_1 in Figure 9(d)-(f)).

Lemma A.5. *Let S be an arbitrary split gadget. The following holds.*

(a) For every minimal conforming partition $\mathcal{R}_0$ of S with stabbing number at most 4 such that exactly 0 reflex segments of $\mathcal{R}$ intersect the in-stab of V, the following holds. Exactly 3 reflex segments of $\mathcal{R}_0$ intersect the left out-stab of V, and exactly 3 reflex segments of $\mathcal{R}_0$ intersect the right out-stab of V (Figure 9(d)).

(b) There exists a minimal conforming partition $\mathcal{R}_1$ of S with stabbing number at most 4 such that exactly 1 reflex segment of $\mathcal{R}_1$ intersect the in-stab of V, and such that the following holds. Exactly 2 reflex segments of $\mathcal{R}_1$ intersect the left out-stab of V and exactly 2 reflex segments of $\mathcal{R}_1$ intersect the right out-stab of V (Figure 9(e)).

Proof. It is enough to prove Lemma A.5 for S_1. We start by naming more parts of S_1. The segment $(4, 9.5)(24, 9.5)$ is the *inner stab* of S_1 (the horizontal thick (purple) segment in Figure 9(d) and (e)). The pixel consisting of a unit square is numbered k if its lower left corner is (Figure 9(b)):

- $(4, 9)$ and $k = 1$,
- $(5, 9)$ and $k = 2$,
- $(13, 9)$ and $k = 3$,
- $(23, 9)$ and $k = 4$,
- $(5, 15)$ and $k = 5$, or
- $(23, 15)$ and $k = 6$.

Let $\mathcal{R}$ be an arbitrary minimal conforming partition of S_1 with stabbing number at most 4. By Lemma A.3, $\mathcal{R}$ includes the verticals of pixel 1 and the horizontals of pixels 5, 6 (Figure 9(c)).

Pixels 2, 3, 4 are all intersected by the inner stab of S_1. Thus, we have the following two cases.

Case 1: If $\mathcal{R}$ includes the verticals of pixel 3, then $\mathcal{R}$ includes the horizontals of pixel 2, 4. Thus, in this case, the partition $\mathcal{R} = \mathcal{R}_0$ satisfies the assertion (a) of Lemma A.5 (Figure 9(d)).

Case 2: If not, $\mathcal{R}$ includes the horizontals of pixel 3. We then have four sub-cases corresponding to $x_{\text{left}}, x_{\text{right}} \in \{0, 1\}$, where $x_{\text{left}} = 0$ (respectively $x_{\text{right}} = 0$) if $\mathcal{R}$ includes the horizontals of pixel 2 (respectively pixel 4) and $x_{\text{left}} = 1$ (respectively $x_{\text{right}} = 1$) if $\mathcal{R}$ includes the verticals of pixel 2 (respectively pixel 4). In the case where $x_{\text{left}}, x_{\text{right}} = 1, 1$, the partition $\mathcal{R} = \mathcal{R}_1$ satisfies the assertion (b) of Lemma A.5 (Figure 9(e)). (The case where $x_{\text{left}}, x_{\text{right}} = 1, 0$ is show in Figure 9(f) as an example of one of the other three cases.)

□

A.5 The Clause Gadget for $k = 4$ Using Thin Polygons

Overview. A clause gadget consists of an axis-aligned rectangle that connects to its corresponding split gadgets or variable gadgets through three connection edges either at the bottom boundary (Figure 10(a)) or at the top boundary. The former case gives a *positive clause gadget* and the latter case gives a *negative clause gadget*. These connection edges are referred to as *left*, *center*, and *right* connection edges. The gadget is designed so that there are exactly 8 minimal conforming partitions and only one has stabbing number 4. We now describe the details.

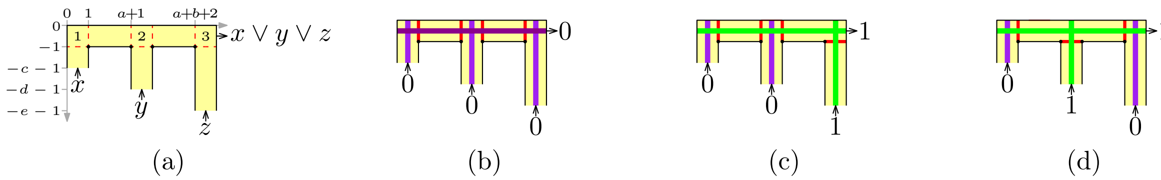


Figure 10: The figure is not to scale. (a) The generic clause gadget $C_0(a,b,c,d,e)$ is shaded (in yellow). The connection segments are the only segments of the boundary of $C_0(a,b,c,d,e)$ which are not solid (black). The reflex vertices are small (black) disks and the reflex segments are dotted (in red). The pixels are numbered from 1 to 3 as defined in the proof of Lemma A.6.
(b) The minimal conforming partition $\mathcal{R}_{000}$ with stabbing number 5. The verticals of pixels $1,2,3$ are solid (red) segments. The in-stabs (respectively inner stab) of $C_0(a,b,c,d,e)$ends with an arrow pointing inside (respectively outside) $C_0(a,b,c,d,e)$. (An in-stab is green if it intersects 2 reflex segments, purple if it intersects 3 reflex segments. Yet, the inner stab is green if it intersects 3 reflex segments or less, and purple otherwise.) The clause gadget $C_0(a,b,c,d,e)$ "propagates false": $\mathcal{R}_0$ includes the solid (red) horizontals of pixels $2,4$ and verticals of pixel 3.
(c) The minimal conforming partition $\mathcal{R}_{001}$ with stabbing number 4.
(d) The minimal conforming partition $\mathcal{R}_{010}$ with stabbing number 3.

Details of a Clause Gadget. Let a,b,c,d,e be positive integers, and $C_0(a,b,c,d,e)$ be the polygon without holes defined as follows (Figure 10). The coordinates of the vertices of $C_0(a,b,c,d,e)$ in counterclockwise order along the boundary are:

$$((0,0),(0,-c-1),(1,-c-1),(1,-1),(a+1,-1),(a+1,-d-1),$$
$$(a+2,-d-1),(a+2,-1),(a+b+2,-1),(a+b+2,-e-1),$$
$$(a+b+3,-e-1),(a+b+3,0)).$$

An rectilinear polygon $C(a,b,c,d,e)$ is a *clause gadget* in the following two cases.

- If there exists a translation τ such that $C(a,b,c,d,e)=\tau(C_0(a,b,c,d,e))$, then $C(a,b,c,d,e)$ is a called a *positive* clause gadget.
- If there exists a transformation τ such that τ is the composition of a horizontal reflection with a translation and such that $C(a,b,c,d,e)=\tau(C_0(a,b,c,d,e))$, then $C(a,b,c,d,e)$ is a called a *negative* clause gadget.

Next, we give names to some segments of interest of $C(a,b,c,d,e)$.

- The edge $\tau((0,-c-1)(1,-c-1))$ is the *left connection edge* of $C(a,b,c,d,e)$ (the bottom left segment of the outer boundary of $C_0(a,b,c,d,e)$ which is not drawn in Figure 10).
- The edge $\tau((a+1,-d-1)(a+2,-d-1))$ is the *center connection edge* of $C(a,b,c,d,e)$ (the bottom center segment of the outer boundary of $C_0(a,b,c,d,e)$ which is not drawn in Figure 10).
- The edge $\tau((a+b+2,-e-1)(a+b+3,-e-1))$ is the *right connection edge* of $C(a,b,c,d,e)$ (the bottom right segment of the outer boundary of $C_0(a,b,c,d,e)$ which is not drawn in Figure 10).
- The stabbing segment $\tau((0.5,0)(0.5,-c-1))$ is the *left in-stab* of $C(a,b,c,d,e)$ (the leftmost thick segment drawn with an arrow pointing upwards inside $C_0(a,b,c,d,e)$ in Figure 10(b), (c), (d), and (e)).
- The stabbing segment $\tau((a+1.5,0)(a+1.5,-d-1))$ is the *center in-stab* of $C(a,b,c,d,e)$ (the center thick segment drawn with an arrow pointing upwards inside $C_0(a,b,c,d,e)$ in Figure 10(b), (c), (d), and (e)).

- The stabbing segment $\tau((a+b+2.5,0)(a+b+2.5,-e-1))$ is the *right in-stab* of $C(a,b,c,d,e)$ (the rightmost thick segment drawn with an arrow pointing upwards inside $C_0(a,b,c,d,e)$ in Figure 10(b), (c), (d), and (e)).

Lemma A.6. *Any clause gadget $C(a,b,c,d,e)$ admits exactly 8 minimal conforming partitions. Specifically, these 8 minimal conforming partitions are the $\mathcal{R}_{xyz}$ such that exactly $x \in \{0,1\}$ (respectively y, z) reflex segments of $\mathcal{R}_{xyz}$ intersect the left (respectively center, right) in-stab of $C(a,b,c,d,e)$ (Figure 10(b), (c), (d), and (e) show respectively $\mathcal{R}_{000}, \mathcal{R}_{001}, \mathcal{R}_{011}, \mathcal{R}_{111}$). Moreover, only $\mathcal{R}_{000}$ has stabbing number greater than 4.*

Proof. It is enough to prove Lemma A.6 for $C_0(a,b,c,d,e)$. We start by naming more parts of $C_0(a,b,c,d,e)$. The segment $(0,-0.5)(a+b+3,-0.5)$ is the *inner stab* of $C_0(a,b,c,d,e)$ (the horizontal thick (purple or green) segment in Figure 10(b), (c), (d) and (e)). The pixel consisting of a unit square is numbered k if its lower left corner is (Figure 10(a)):

- $(0,-1)$ and $k=1$,
- $(a+1,-1)$ and $k=2$, or
- $(a+b+2,-1)$ and $k=3$.

Lemma A.2 applied to pixels $1,2,3$ indeed shows that there exists exactly 8 minimal conforming partition of $C_0(a,b,c,d,e)$ which are $\{\mathcal{R}_{xyz} : x,y,z \in \{0,1\}\}$ as stated in Lemma A.6.

Given that the three pixels $1,2,3$ all intersect the inner stab of $C_0(a,b,c,d,e)$, we check that the inner stab of $C_0(a,b,c,d,e)$ intersects at most 4 reflex segments of the $\mathcal{R}_{xyz}$ except $\mathcal{R}_{000}$. □

A.6 Proof of Theorem 3.1 When $k=4$ Using Polygons in General Position

It is straightforward to observe that 4-STAB and 4-CSTAB are in $\mathcal{NP}$. We now prove that the problem 4-CSTAB is $\mathcal{NP}$-hard by reducing Problem 1 to 4-CSTAB in polynomial time. Our reduction uses thin polygons, and hence by Lemma A.1, the hardness result holds for 4-STAB.

Let D,ϕ be an instance of Problem 1, that is, a rectilinear planar monotone drawing D of a 3-CNF formula ϕ (D is shaded in Figure 2 and displayed alone in Figure 6). Let u be the number of clauses of ϕ and v be the number of variables of ϕ. The number of variable gadgets and clause gadgets is at most $u+v$. The number of split gadgets is at most a constant factor of the number of clause gadgets. Therefore, it is straightforward to construct the corresponding polygon $P(D,\phi)$, i.e., an instance of the stabbing number problem, (Figure 2(a)) in polynomial time in $u+v$.

Next, we prove that the formula ϕ is satisfiable if and only if $P(D,\phi)$ admits a minimal conforming partition with stabbing number at most 4.

Assume that the polygon $P(D,\phi)$ admits a conforming partition with stabbing number at most 4. By Lemma A.4, each variable gadget of $P(D,\phi)$ admits exactly three minimal conforming partitions with stabbing number at most 4, one standing for true, one standing for false, and the last one being undetermined. Because each out-stab pairs up with an in-stab and their union is a stabbing segment of $P(D,\phi)$, Lemma A.4, Lemma A.5, and Lemma A.6 implies a consistent truth value assignment for all the variables, with the convention that the variable of an undetermined variable gadget is set to true (this convention is arbitrary). Finally, by Lemma A.6 and because the values propagated in the gadgets does not increase, the clause must have at least one of its literals set to true.

If the formula ϕ is satisfiable, then there exists a variable assignment such that all the clauses of ϕ evaluate to true. Here we use the constructions described in the proof of Lemma A.4, Lemma A.5, and Lemma A.6 to obtain a minimal conforming partition of $P(D,\phi)$ with stabbing number at most 4.

A.7 Proof of Theorem 3.1 When $k=4$ Using Polygons in General Position

To prove the $\mathcal{NP}$-hardness for the polygons in general position, we use the same technique as for proving the hardness results for thin polygons, but modify the gadgets. The functionalities of each gadget remains the same, therefore, we only give a high-level overview of the changes.

Forcer gadget. We use a staircase with 6 reflex vertices as the force gadget, which is shown in Figure 11(a) and (c). We need the property that in any conforming partition of the gadget with stabbing number 4, the maximal stabbing segment s perpendicular to the connection edge stabs 3 reflex segments. Suppose for a contradiction that s intersects smaller than three reflex segments. Then at least four vertical reflex segments in the partition would reach the topmost edge of the gadget implying a stabbing number higher than 4, a contradiction. We also need the gadget to have a conforming partition with stabbing number at most 4, which is shown in Figure 11(b).

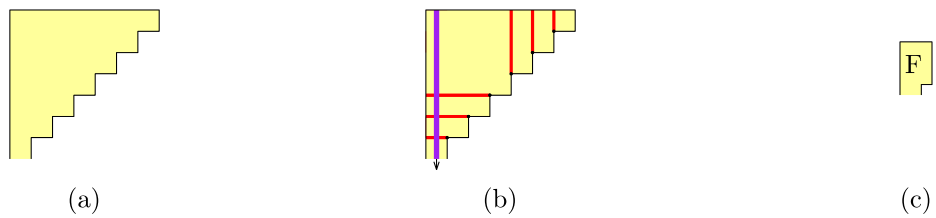


Figure 11: (a) The generic gadget. (b) The unique conforming partition with stabbing number 4. (c) A schematic drawing.

Variable gadget. The variable gadget (Figure 12(a)) is a careful perturbation of the variable gadget that we used previously for thin polygons. Following the previously defined variable gadget, we define the positive and negative connection edges and their corresponding positive and negative out-stabs. This variable gadget in general position also has exactly three minimal conforming partitions with stabbing number at most 4. This can be verified by first observing the stabbing segments imposed by the forcer gadgets, and then using a case analysis on the four reflex vertices between the forcer gadgets. Two of these configurations are used to determine the truth values of the variable (Figure 12(b)-(c)), while the third one (called *undetermined*) is by convention interpreted as true (Figure 12(d)). In a false (true) configuration, exactly 3 (exactly 2) reflex segments intersect the positive out-stab, and exactly 2 (exactly 3) reflex segments intersect the negative out-stab. Similar to the previously defined variable gadget, if a positive (negative) out-stab intersects only two reflex segments, then it forces two vertical reflex segments inside the gadget, which enforces the negative (positive) out-stab to intersect three reflex segments.

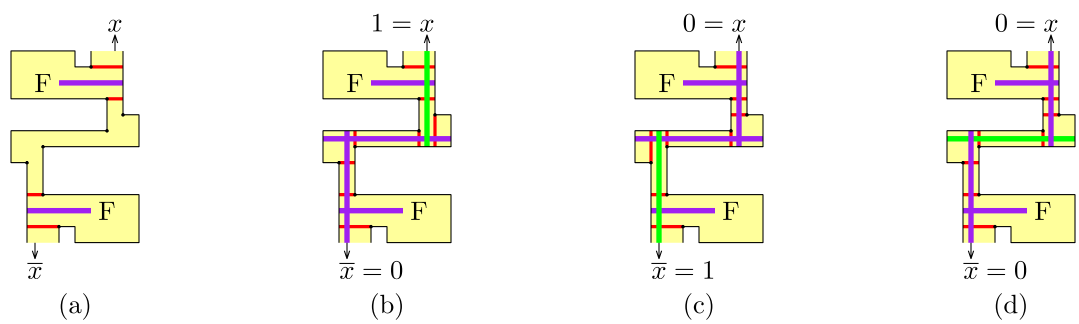


Figure 12: (a) The variable gadget, with reflex segments that are forced by the forcer gadgets. (b) Reflex segments to encode $x = 1$. (c) Reflex segments to encode $x = 0$. (d) The variable gadget is undetermined.

Split gadget. The split gadget in general position (Figure 13(a)) has the same property as the split gadget that we built for thin polygon. This gadget is slightly different as it uses one less forcer gadet. However, from the perspective of the split gadget, we can still define *in connection edge*, *left connection edge*, and *right connection edge* that connect the split gadget to the rest of the polygon, and the corresponding perpendicular maximal stabbing segments as *in-stab*, *left out-stab* and *right out-stab*. The property that we need for this gadget is that the value of the maximal stabbing segment entering from a variable gadget into the split gadget is propagated (either as it is, or with a decreased value) to the two stabbing segments leaving the split gadget.

Consider first the case when the in-stab does not intersect any reflex segment, i.e., corresponds to the value 0 (Figure 13(c)). We now show the left out-stab (similarly, the right out-stab) must propagate 0, i.e., it will intersect 3 reflex segments. Note that the forcer gadget near the left-out stab enforces two horizontal reflex segments. If the left-out stab does not intersect any more reflex segment (i.e., if it propagates 1), then we must have two vertical reflex segments inside the gadget that are imposed by the out-stab. These two vertical reflex segments together with the vertical reflex segments imposed by the in-stab implies a stabbing number larger than 4, a contradiction.

Consider now the case when the in-stab intersects only one reflex segment (corresponding to the value 1). We are now free to propagate either 0 or 1 through the out-stabs. These are illustrated with the partition in Figure 13(b) and (d).

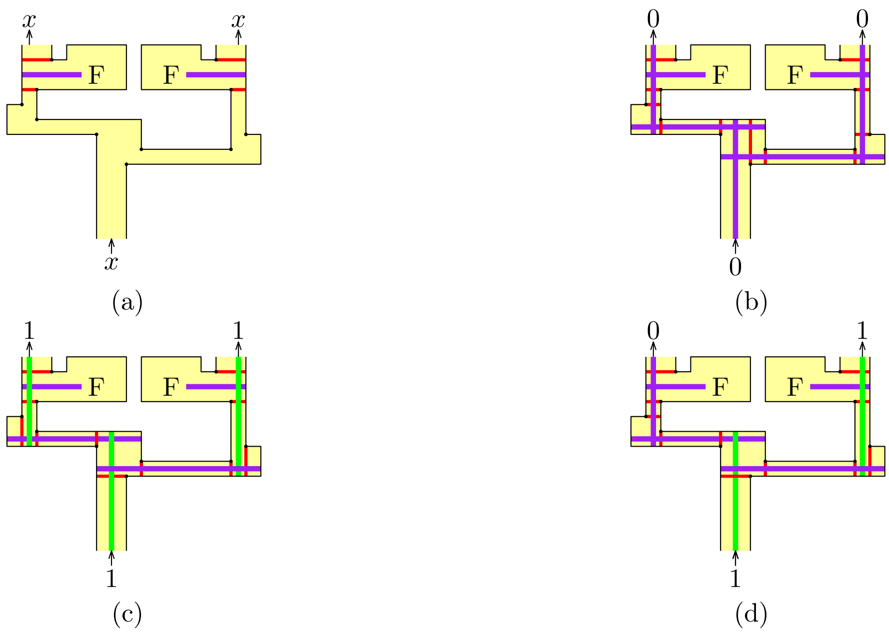
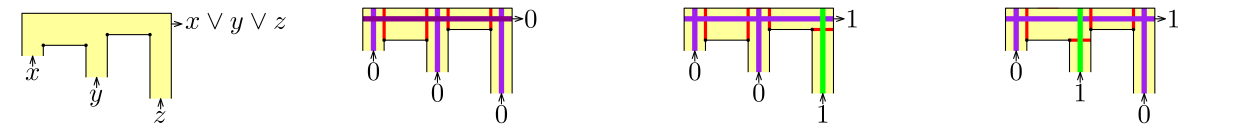


Figure 13: (a) The split gadget, with reflex segments that are forced by the forcer gadgets. (b) A set of reflex segments that propagates 0. (c) A set of reflex segments that propagates 1. (d) It is possible to reduce the propagated value (but it cannot increase).

Clause gadget. A clause gadget in general position has the same properties as the one for thin polygons (Figure 14(a)). The reflex vertices are perturbed such that a stabbing number greater than 4 would require all in-stabs to propagate 0 values (Figure 14(b)). For any other combination of values, there exists a partition with stabbing number at most 4 where each in-stab that propagates 1 intersects exactly one reflex segment, and each in-stab that propagates 0 does not intersect any reflex segment. Figure 14(c) and (d) illustrate such choices when at least one incoming value is 1.

A.8 Proof of Theorem 3.1 When $k > 4$

For the case when $k = 5$, we use the force gadget as shown in Figure 16. Similarly to the proof of Lemma A.3, we show that the out-stab intersects at least four reflex segments in any conforming partition $\mathcal{R}$ with stabbing number at most 5 of F (Figure 15). Indeed, considering only the 16 pixels which are the wedge-pixels of



(a) (b) (c) (d)

Figure 14: (a) The generic clause gadget. (b) If all incoming values are 0, then we require stabbing number 5. (c-d) If at least one incoming value is 1, then there is a choice of reflex segments with stabbing number at most 4.

some reflex vertex (all of which satisfy the premise of Lemma A.2), $\mathcal{R}$ includes the verticals of at most two wedge-pixels per row, that is, $\mathcal{R}$ includes the horizontals of at least two wedge-pixels per row. Thus, in total, $\mathcal{R}$ includes the horizontals of at least eight wedge-pixels. Since $\mathcal{R}$ includes the horizontals of at most two wedge-pixels per column, each column has exactly four horizontal reflex segments included in $\mathcal{R}$. Furthermore, the gadget indeed admits a partition with stabbing number at most 5 (Figure 15(b)).

(a) (b)

Figure 15: (a) The forcer gadget in the context of 5-STAB. (b) A partition with stabbing number at most 5. The out-stab is thick (in purple).

We now prove the hardness for k-CSTAB by generalizing the force gadgets. For the case when k is even, i.e. $k = 4 + 2m$, where m is a positive integer, we generalize the forcer gadget for stabbing number 4 to a forcer gadget for stabbing number k, by adding m rows and m columns, where m is a positive integer such that $k = 4 + 2m$. Then we generalize the polygon $P(\phi)$ for stabbing number 4 to a polygon for stabbing number k. This is done by attaching m forcer gadgets for stabbing number k to the middle of each row or column of adjacent pixels of $P(\phi)$. The hardness reduction for k-CSTAB now follows the same technique that we used to prove the hardness of 4-CSTAB. The case when k is odd is handled similarly by starting with the hardness of 5-CSTAB. Figure 16 illustrates an instance for 5-CSTAB, which corresponds to the same RPM-3-SAT instance that we used in the hardness proof for 4-CSTAB.

B Running Time for 2-CSTAB

In this section, we explain how to check the rules for 2-CSTAB in $O(n \log n)$ time. The main difficulty is that we cannot afford to compute the entire pixelation graph (or equivalently, all intersections between reflex segments), since there may be $\Omega(n^2)$ intersections. Instead, it is possible to derive all required information via *orthogonal ray-shooting queries*, which we define first. For such queries, we have a data structure $\mathcal{S}$ that stores disjoint parallel line segments. A ray-shooting query receives as input a ray that is perpendicular to the segments, and it reports the first segment in $\mathcal{S}$ that is hit by the ray, or that there is no such segment. (Here rays are considered open at the start point, i.e., we do not report a segment that lies exactly on the start point, but the next one afterwards.) Giyora and Kaplan [14] gave an implementation that performs such a query in $O(\log |\mathcal{S}|)$ time and that also permits (within the same running time) to delete segments of $\mathcal{S}$. We maintain the following ray-shooting data structures:

- $\mathcal{R}_{\text{hor}}$ stores all horizontal reflex segments, as well as segments corresponding to all horizontal edges

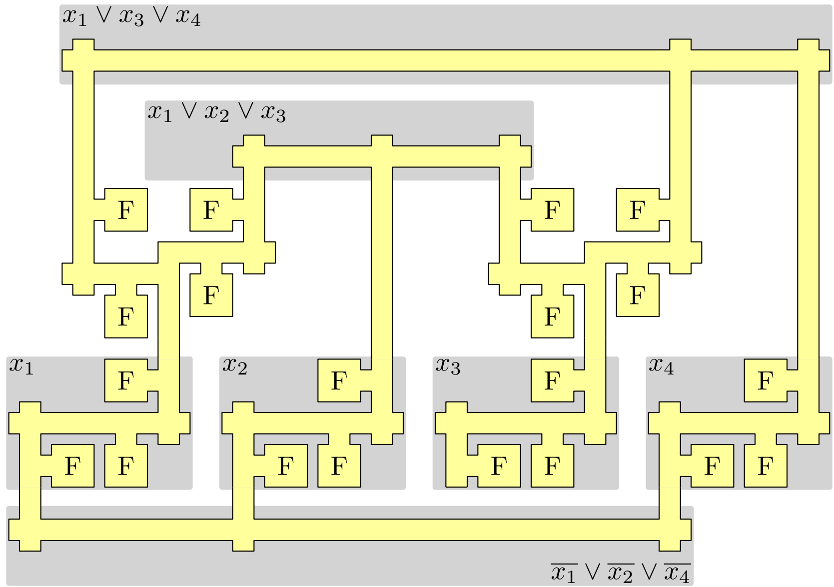


Figure 16: The polygon used to reduce RPM-3-SAT to 5-STAB. (The figure is not to scale and the split gadgets are drawn outside the variable rectangles only to fit the figure in the page.)

of P. During later iterations, we will remove from it reflex segments that have been discovered to be impossible.

- $\mathcal{S}_{\text{hor}}$ stores one representative from each equivalence class of horizontal stabbing segments, as well as segments corresponding to all horizontal edges of P. Slightly abusing notation, we will from now on use the term "stabbing segment" to mean "the representative of one equivalence class of stabbing segments".

  During later iterations, we will remove from $\mathcal{S}_{\text{hor}}$ those stabbing segments where rule (R3) has been applied (which means that they no longer intersect flexible segments).

- $\mathcal{R}_{\text{ver}}$ and $\mathcal{S}_{\text{ver}}$ are defined symmetrically for vertical segments.

Recall that reflex segments and stabbing segments are open segments. We use closed segments for the edges of P; with this, all segments in each data structure are disjoint and parallel as required. We can initially populate these data structures by computing all reflex segments and all equivalence classes of stabbing segments with line-sweeps in $O(n \log n)$ time. Along the way, we can also immediately check for applications of rule (R5), i.e., whether a reflex segment $\mathbf{s}$ is a gate. If so, then $\mathbf{s}$ is fixed, and we add it to a list L_{fixed}.

Note that for any stabbing segment $\mathbf{s}$, we can answer 'is there a reflex segment that intersects $\mathbf{s}$?' with a single ray-shooting query as follows. Assume that $\mathbf{s}$ is horizontal (the other case is symmetric) and perform a ray-shooting query in $\mathcal{R}_{\text{ver}}$ starting at one endpoint of $\mathbf{s}$ and going in the direction along $\mathbf{s}$. This ray always hits some segment because it goes inwards into P and the vertical edges of P are represented in $\mathcal{R}_{\text{ver}}$. Since we report the first segment that the ray hits, $\mathbf{s}$ intersects some reflex segment if and only if we did not hit a segment of P, and we can find this out in $O(\log n)$ time. In particular, we can therefore now detect in $O(n \log n)$ all segments that are impossible due to rule (R4): For each reflex segment (which is a special case of a stabbing segment), test whether it intersects some other reflex segment in $O(\log n)$ time. If so, add it to list $L_{\text{impossible}}$.

With this we have encountered all situations where rules (R4) and (R5) apply, since they only depend only on the structure of the polygon and not on whether reflex segments are fixed or impossible. We obtained initial lists L_{fixed} and $L_{\text{impossible}}$, with every segment at most once in each. Some segments may belong to both lists (which would tell us that no solution exists), but we will not spend time to determine this yet because it will be detected naturally later.

Now we turn to the propagation, i.e., rules (R2) and (R3), for which we need some more data structures and invariants.

- We store a flag with each reflex segment that is initially 'undecided', but may get changed to 'impossible' or 'fixed' later. A flag of 'impossible' or 'fixed' means not only that the segment was in the appropriate list, but also that we have processed the segment in the sense that we have applied all rules that can be applied due to its status. For example, each segment that was added to $L_{\text{impossible}}$ triggers (potentially) rule (R1) and (R2); we set its flag to 'impossible' once we have checked that (R1) does not apply with the current flags, and that we have applied (R2).
- To avoid double-counting, it will be important that from now on, $\mathcal{R}_{\text{ver}}$ and $\mathcal{R}_{\text{hor}}$ do not contain reflex segments that are in $L_{\text{impossible}}$. So we parse $L_{\text{impossible}}$ and remove all these segments from the data structures; this takes $O(n \log n)$ time.
- We have a list L_{R3}, which is initially empty but will get populated with stabbing segments during the propagation whenever we find one where rule (R3) can be applied. To avoid double-counting, it will be important that $\mathcal{S}_{\text{ver}}$ and $\mathcal{S}_{\text{hor}}$ do not contain stabbing segments that are in L_{R3}, which clearly holds initially.

We will do many more ray-shooting queries and other operations that take $O(\log n)$ time. To bound the running time, we will assign each such operation to a flag-change of a reflex segment or an addition to L_{R3}. Since reflex segments change flags at most once, and stabbing segments are added to L_{R3} at most once, therefore the total running time is $O(n \log n)$.

- We first explain how to process a segment that has been added to $L_{\text{impossible}}$; up to symmetry we assume that it is a horizontal segment, say $\mathbf{h}_p$. First check the flag of $\mathbf{h}_p$; obviously there is nothing to do if this is already 'impossible'. If the flag is 'fixed' then $\mathbf{h}_p$ is in both L_{fixed} and $L_{\text{impossible}}$ and we can abort the algorithm since there cannot be a solution.

  So assume for the rest that the flag was 'undecided'. Set it to be 'impossible' and look up the vertical segment $\mathbf{v}_p$. If the flag of $\mathbf{v}_p$ is 'impossible' then by rule (R1) there is no solution and we abort. Otherwise, by rule (R2) segment $\mathbf{v}_p$ should become fixed, so we add it to L_{fixed} if its flag was 'undecided'. The entire running time was $O(1)$, which we count as overhead to the flag-change of $\mathbf{h}_q$.
- Now we explain how to process a segment that has been added to L_{fixed}. Assume up to symmetry that the segment is horizontal, say it is $\mathbf{h}_p$. There is nothing to do if its flag is 'fixed', and we can abort the algorithm if its flag is 'impossible', so assume that the flag was 'undecided' and set it to be 'fixed'.

  Rule (R3) applies to any stabbing segments $\mathbf{s}$ that intersect $\mathbf{h}_p$. To find such stabbing segments, perform multiple ray-shooting queries within $\mathcal{S}_{\text{ver}}$, starting at p and in the direction of $\mathbf{h}_p$, then continue from the point where a segment was hit (still in the direction of $\mathbf{h}_p$), and so on, until we hit a segment of P. Say we found $\ell \geq 0$ stabbing segments, hence did $\ell{+}1$ ray-shooting queries. Each found stabbing segment $\mathbf{s}$ was not in L_{R3} since we removed segments of L_{R3} from $\mathcal{S}_{\text{ver}}$. We now add $\mathbf{s}$ to L_{R3} and remove it from $\mathcal{S}_{\text{ver}}$; the time for this (as well as the corresponding ray-shooting query) is accounted for by the addition of $\mathbf{s}$ to L_{R3}. The last ray-shooting query is accounted for by the flag-change of $\mathbf{h}_p$.
- Now we explain how to handle a stabbing segment $\mathbf{s}$ after it was added to L_{R3}; up to symmetry we may assume that it is horizontal. To apply rule (R3), we first determine all reflex segments that are intersected by $\mathbf{s}$; similar to above we can find these segments (say there are ℓ of them) with $\ell{+}1$ ray-shooting queries in $\mathcal{R}_{\text{ver}}$. Note that none of these ℓ segments is in $L_{\text{impossible}}$ since we remove segments in $L_{\text{impossible}}$ from $\mathcal{R}_{\text{ver}}$. There will be one such segment that is fixed (the one that caused $\mathbf{s}$ to be in L_{R3}); we count the ray-shooting query that led to it (as well as the last ray-shooting query) towards the addition of $\mathbf{s}$ to L_{R3}. Any other segment $\mathbf{v}_q$ that we encounter is not yet in $L_{\text{impossible}}$; we remove it from $\mathcal{R}_{\text{ver}}$, add it to $L_{\text{impossible}}$, and the flag-change that it will undergo there later accounts for the running time for these operations as well as the corresponding ray-shooting query.

We keep applying the above steps for as long as there are entries in one of $L_{\text{fixed}}, L_{\text{impossible}}, L_{\text{R3}}$ that have not been processed yet. At the end, we return the set L_{fixed}, and go to the second part of the algorithm.

C Details of Theorem 5.1

We must give an MSOL formula for k-CSTAB, for which we first review what kinds of formulas are permitted in this logic for graph problems. We are allowed to use variables that are vertex sets or individual vertices, and to quantify over these variables. We are also allowed to use the usual boolean operations, as well as any predicates that can be read in constant time from the graph. In our context, we will build a formula based on the radial graph R_P, and need the following predicates:

- $\xi \in \Xi_P$? This should be true if and only if ξ is a vertex of R_P that corresponds to a pixel. Via negation this also gives us the predicate '$v \in V_P$?'.
- *isQuadrant*(v, ξ, i)? This is defined for $i \in \{1, 2, 3, 4\}$ and should be true if and only if $v \in V_P$, $\xi \in \Xi_P$, and ξ is the pixel incident to v at the ith quadrant.
- *isReflex*(v)? This should be true if and only if $v \in V_P$ is a vertex at a reflex corner of P.

Note that these predicates can clearly be read from the radial graph R_G, presuming vertices and edges have been marked suitably.

We now build a formula $\phi(R_P, k)$ that is satisfied if and only if polygon P has a conforming partition with stabbing number at most k.

- As sketched earlier, we have two variables $\mathcal{V}$ and $\mathcal{H}$ which are meant to be the vertices of V_P that lie on the vertical respectively horizontal segments of a conforming partition.
- Our first requirement is therefore that $\mathcal{V} \subseteq V_P$ (formally expressed via the formula $\forall v : v \in \mathcal{V} \Rightarrow (v \in \Xi_P)$). Similarly we require $\mathcal{H} \subseteq V_P$.
- We next require that every reflex vertex is in at least one of $\mathcal{V}$ and $\mathcal{H}$, i.e., it must use at least one of its reflex segments:
$$\forall v : v \in V_P \wedge \mathit{isReflex}(v) \Rightarrow (v \in \mathcal{H} \vee v \in \mathcal{V}).$$
- We next need to ensure that no two reflex segments cross each other, i.e., no vertex of G_P in the strict interior of P belongs to both $\mathcal{V}$ and $\mathcal{H}$. We can determine whether a vertex lies in the strict interior by checking that there are four pixels that are its four quadrants, and so have:
$$\forall v : v \in V_P \wedge \left(\exists \xi_1, \xi_2, \xi_3, \xi_4 \in \Xi_P : \bigwedge_{i=1}^{4} \mathit{isQuadrant}(v, \xi_i, i) \right) \Rightarrow (v \notin \mathcal{H} \vee v \notin \mathcal{V}).$$
- We also must ensure that $\mathcal{H}$ indeed encodes horizontal reflex segments. To do so, we ensure that if a vertex belongs to $\mathcal{H}$, then so do its horizontal neighbours; the requirement then propagates along the entire horizontal segment (which is a reflex segment by definition of G_P). We can find the horizontal neighbours by looking for an incident pixel and reading the appropriate vertex from the quadrant-information:
$$\forall v, \xi, v' : v \in \mathcal{H} \wedge \xi \in \Xi_P \wedge v' \in V_P \wedge \mathit{isQuadrant}(v, \xi, 1) \wedge \mathit{isQuadrant}(v', \xi, 2) \Rightarrow v' \in \mathcal{H}$$
(and similarly for the other three quadrants).
- Symmetrically we can enforce that $\mathcal{V}$ corresponds to vertical reflex segments.

If we take the conjunction of all the above formulas, then any satisfying assignment hence encodes a conforming partition. Note that this part of the formula has constant size, independent of k. Now we add to the formula to enforce that each stabbing segment intersects at most k rectangles of the conforming partition:

- As sketched earlier, we have k variables $\Xi_1^{\text{hor}}, \ldots, \Xi_k^{\text{hor}}$ with the intended meaning that they cover Ξ_P, and$\xi \in \Xi_i^{\text{hor}}$ means that the *horizontal rectangle-count of* ξ is at most i. By this, we mean that the horizontal stabbing segment that goes through pixel ξ hits at most i rectangles of the conforming partition at or to the left of ξ.
- Our first requirement is therefore that these sets cover all of Ξ_P :

$$\forall \xi : \xi \in \Xi_P \Rightarrow \left(\xi \in \Xi_1^{\text{hor}} \vee \cdots \vee f \in \Xi_k^{\text{hor}}\right)$$

- Consider two pixels ξ, ξ' that share a vertical edge, with ξ left of ξ'. If this vertical edge belongs to the conforming partition, then we must increase the horizontal rectangle-count of ξ. The former can be expressed by testing the vertex that is common to ξ and ξ', and the latter can be expressed via the indices of the sets $\Xi_1^{\text{hor}}, \ldots, \Xi_k^{\text{hor}}$: Whichever one contains ξ, the one for ξ' must have index one larger.

$$\begin{aligned} \forall \xi, \xi', v : & \xi \in \Xi_P \wedge \xi' \in \Xi_P \wedge \mathit{isQuadrant}(v, \xi, 2) \wedge \mathit{isQuadrant}(v, \xi', 1) \\ & \wedge v \in \mathcal{V} \Rightarrow \left(\bigwedge_{i=1}^{k-1} (\xi \in \Xi_i^{\text{hor}} \Rightarrow \xi' \in \Xi_{i+1}^{\text{hor}}) \right) \end{aligned}$$

  On the other hand, if the vertical edge does not belong to the conforming partition, then the horizontal rectangle-count should stay the same.

$$\begin{aligned} \forall \xi, \xi', v : & \xi \in \Xi_P \wedge \xi' \in \Xi_P \wedge \mathit{isQuadrant}(v, \xi, 2) \wedge \mathit{isQuadrant}(v, \xi', 1) \\ & \wedge \neg(v \in \mathcal{V}) \Rightarrow \left(\bigwedge_{i=1}^{k-1} (\xi \in \Xi_i^{\text{hor}} \Rightarrow \xi' \in \Xi_i^{\text{hor}}) \right) \end{aligned}$$

  Note that the length of this part of the formula depends linearly on k.

- Even though the above restrictions do not exactly encode what we wanted (we do not enforce a partition, and we do not require that the horizontal rectangle-count starts at 1), it is enough to force what we want: every horizontal stabbing segment **s** intersects at most k rectangles. To see this, letξ, ξ' be the leftmost and rightmost inner pixel intersected by **s**, and let a be such that$\xi \in \Xi_a^{\text{hor}}$ (noting that a need not be unique). If **s** intersects j segments of the partition, then the restrictions force$\xi' \in \Xi_{a+j}^{\text{hor}}$; since this set only exists for indices up to k and $a \geq 1$ therefore $j \leq k-1$ as required.

Symmetrically, we can add restrictions that force that vertical stabbing segments intersect at most k rectangles, and hence the entire formula is satisfiable if and only if the polygon has a conforming partition with stabbing number at most k.

D Details of Theorem 5.2

Recall that G_P denotes the pixelation graph of polygon P, and that we assume here that P is simple and in general position. We show how to solve k-CSTAB in P by applying a dichotomy argument, somewhat similar to bidimensionality [10]. Define the *length* of a reflex segment **s** of P to be the number of inner vertices of the pixelation graph G_P that lie on **s**. We can compute G_P and the length of all reflex segments in $O(n^2)$ time, and now have two cases. In the first case all reflex segments have length less than $\ell := k(2^{k+2} - 4) + (k-1)$. (The reason for this cutoff point will be clear later; it has been chosen for ease of explanation and almost surely could be improved.) Then every vertex v of G_P is within distance $\lceil \ell/2 \rceil$ of a vertex that lies on P, simply by walking along a reflex segment that defined v. Since P has no holes, therefore all vertices of G_P are within distance $\lceil \ell/2 \rceil$ of some vertex on the outer face of G_P. It follows that the so-called *outerplanarity* of G_P is at most $\lceil \ell/2 \rceil + 1$, which in turn implies that G_P has treewidth $O(\ell)$, see [5] for details. Since ℓ only depends on k and not on n, this means that we can then test k-CSTAB in time $O(f(k, O(\ell))n^2)$ time by

Theorem 5.1, which gives the result in this case. As for the other case, we argue below that then k-CSTAB has no solution, so we simply return a negative answer. So Theorem 5.2 as proved as long as we can show the following lemma:

Lemma D.1. *Let P be a rectilinear polygon without gates that contains a reflex segment* $\mathbf{s}$ *of length at least* $\ell := k(2^{k+2} - 4) + (k-1)$*. Then any conforming partition of P has stabbing number at least $k+1$.*

Proof. We may assume, up to rotation, that $\mathbf{s}$ is horizontal. Assume for contradiction that we had a conforming partition with stabbing number at most k, say it uses horizontal and vertical reflex segments $\mathcal{H}$ and $\mathcal{V}$. This conforming partition splits $\mathbf{s}$ into at most k pieces (one for each rectangle intersected by stabbing segment $\mathbf{s}$). With our choice of ℓ, and since at most $k-1$ vertices of $\mathbf{s}$ lie on segments of $\mathcal{V}$, one of these pieces of $\mathbf{s}$ has at least $2^{k+2} - 4$ vertices that are not on segments of $\mathcal{V}$. Let R_0 be a rectangle that (just barely) includes the vertices of this piece of $\mathbf{s}$, i.e., we thicken the piece in all directions by ε, where $\varepsilon > 0$ is so small that pixels have width and height at least 3ε. See also Figure 17.

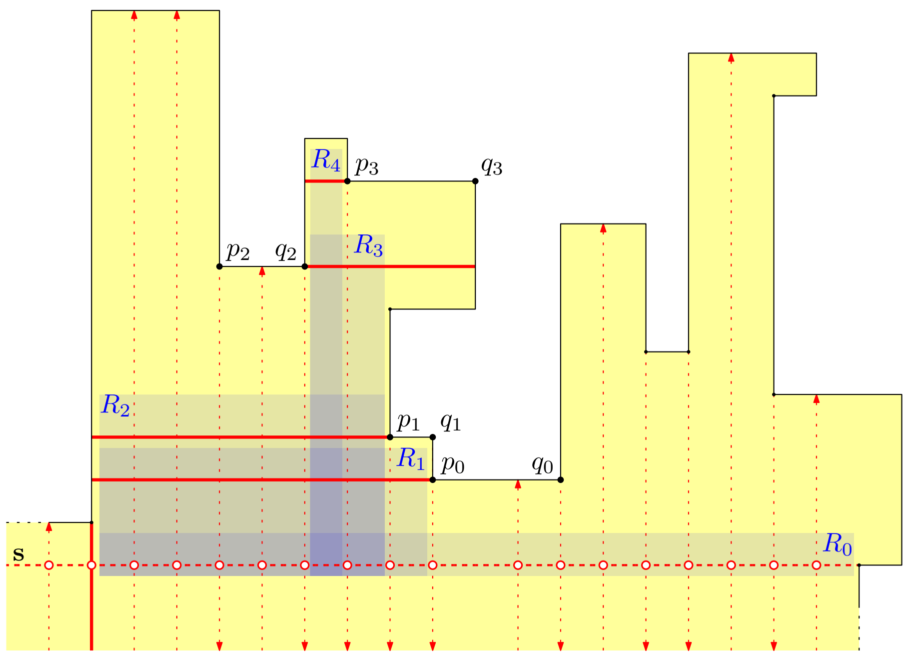


Figure 17: We increasingly narrow the rectangle and raise its top side until we are left with a rectangle that intersects at least k segments of $\mathcal{H}$. . Red (thick) segments are in the partition. For space reasons we do not show the lower half of P (which contains reflex vertices to create the remaining vertices on $\mathbf{s}$.) We write p_d, q_d for the vertices p, q used to determine R_{d+1} from R_d.

So there are at least $2^{k+2} - 4$ reflex segments that intersect R_0. Of those, up to mirroring vertically, at least half have their defining reflex vertex above R_0; we call these the *downward segments* $\mathcal{D}$ and indicate them with downward arrows in Figure 17. Our goal is now to define inductively rectangles R_d (for $d = 0, 1, \dots$) that satisfy the following:

- R_d is non-empty and lies within P,
- R_d intersects at least $2^{k+1-d} - 2$ downward segments,
- R_d intersects at least d segments in $\mathcal{H}$,
- for $d > 0$, the x-range of R_d is within the x-range of R_{d-1}, and the y-range of R_{d-1} is within the y-range of R_d.

We already defined R_0 and one easily verifies that it satisfies all conditions since $|\mathcal{D}| \geq 2^{k+1} - 2$. Now assume that we have found $R_d = [\ell_d, r_d] \times [b_d, t_d]$ with the above properties. If R_d intersects no downward segment, then $2^{k+1-d} - 2 \leq 0$, hence $d \geq k$. The left side of R_d then intersects at least $d \geq k$ segments of $\mathcal{H}$, hence the stabbing segment through it intersects at least $k+1$ rectangles of the partition, impossible.

So R_d intersects at least one downward segment. Raise the top side of R_d until we first hit a horizontal edge $\overline{pq}$ of P (where p is left of q), and set $t_{d+1} = y(p) + \varepsilon$. The goal is now to narrow the width such that the new rectangle is left or right of $\overline{pq}$, with the choice between these made based on which one leads to more intersecting downward segments. Formally, let $\mathcal{D}_\ell$ be the (possibly empty) set of downward segments that intersect R_d and whose x-coordinate is at most $x(p)$. Let $\mathcal{D}_r$ be the set of downward segments that intersect R_d and whose x-coordinate is at least $x(q)$.

Assume that $|\mathcal{D}_\ell| \geq |\mathcal{D}_r|$; the other case is similar (the new rectangle would then be right of q). No downward segment has x-coordinate in the range $(x(p), x(q))$, since by choice of t_{d+1} they would otherwise intersect $\overline{pq}$. So we have $|\mathcal{D}_\ell| + |\mathcal{D}_r| \geq 2^{k+1-d} - 2$ and therefore $|\mathcal{D}_\ell| \geq 2^{k-d} - 1$. Also $|\mathcal{D}_\ell| \geq 1$ since we have downward segments that intersect R_d. This implies $x(p) > \ell_d$, which together with the choice of $\overline{pq}$ and t_{d+1} means that p is a reflex vertex that lies in the interior of the x-range of R_d.

Define $r_{d+1} := x(p) - \varepsilon < r_d$, and keep the other sides of R_d unchanged, i.e., define $R_{d+1} := [\ell_d, r_{d+1}] \times [b_d, t_{d+1}]$. Rectangle R_{d+1} by construction does not intersect $\overline{pq}$. It also intersects no other horizontal edges of P by choice of $\overline{pq}$ and since P has no gates (so no two horizontal edges of P are hit at the same time when raising the top side). So R_{d+1} is within P. It is non-empty because it contains by choice of ε parts of the pixels left of p. Each segment in $\mathcal{D}_\ell$ intersects R_{d+1}, with the exception of $\mathbf{v}_p$, so R_{d+1} intersects at least $2^{k-d} - 2$ downward segments as required. Rectangle R_{d+1} intersects the same segments of $\mathcal{H}$ as R_d, and it additionally intersects $\mathbf{h}_p$, which must be in $\mathcal{H}$ since $\mathbf{v}_p$ is not in $\mathcal{V}$ (recall our choice of piece of $\mathbf{s}$ that defined R_0). So rectangle R_{d+1} satisfies all conditions. We can therefore continue the process indefinitely, but this is impossible since strictly fewer downward segments intersect the rectangles. So eventually we must run into the above contradiction. □

Unfortunately, this proof does not carry over to simple polygons where reflex vertices may align. Consider a polygon that is essentially a rectangle, except that we attach $\Theta(n)$ teeth on the left side and $\Theta(n)$ teeth on the top side. This has stabbing number 2, by inserting the reflex segments that cut off the teeth. But it contains reflex segments of length $\Theta(n)$, and in fact, G_P cotains a $\Theta(n) \times \Theta(n)$-grid and has treewidth $\Theta(n)$. So a dichotomy based on treewidth or length of reflex segments does not work. We conjecture that some other approach would work, and so leave one important open problem.

Conjecture 1. *Testing whether a simple polygon P has conforming stabbing number k is fixed-parameter tractable in k.*