Computational Complexity of Flattening Fixed-Angle Orthogonal Chains

Erik D. Demaine* Hiro Ito† Jayson Lynch‡ Ryuhei Uehara§

Abstract

Planar/flat configurations of fixed-angle chains and trees are well studied in the context of polymer science, molecular biology, and puzzles. In this paper, we focus on a simple type of fixed-angle linkage: every edge has unit length (equilateral), and each joint has a fixed angle of 90° (orthogonal) or 180° (straight). When the linkage forms a path (open chain), it always has a planar configuration, namely the zig-zag which alternating the 90° angles between left and right turns. But when the linkage forms a cycle (closed chain), or is forced to lie in a box of fixed size, we prove that the flattening problem — deciding whether there is a planar noncrossing configuration — is strongly NP-complete.

Back to open chains, we turn to the Hydrophobic–Hydrophilic (HP) model of protein folding, where each vertex is labeled H or P, and the goal is to find a folding that maximizes the number of H–H adjacencies. In the well-studied HP model, the joint angles are not fixed. We introduce and analyze the fixed-angle HP model, which is motivated by real-world proteins. We prove strong NP-completeness of finding a planar noncrossing configuration of a fixed-angle orthogonal equilateral open chain with the most H–H adjacencies, even if the chain has only two H vertices. (Effectively, this lets us force the chain to be closed.)

1 Introduction

In this paper, we introduce and investigate a new model of protein folding. We are given an ***equilateral fixed-angle chain*** ("protein"), where each vertex is marked H or P and has a specified fixed angle, and edges all have unit length. The goal is to embed the chain into a given grid (e.g., 2D square, 3D cube, 2D triangular, or 2D hexagonal) while

1. respecting the fixed angles (but each angle is still free to be a left or right turn in 2D or spin in 3D);
2. without self-crossing in the embedding; and
3. maximizing the number of H–H grid adjacencies.

This is a fixed-angle version of the well-studied HP model of protein folding (where the angles are normally free to take on any value), which is known to be NP-hard in the 2D square grid [4] and 3D cube grid [3]. Fixed angles are motivated by real-world proteins; see [7, Chapters 8–9]. In the 2D square grid or 3D cube grid studied here, we can restrict to ***orthogonal*** fixed-angle chains where all fixed angles are 90° or 180°. For example, the popular "Tangle" toy restricts further to all fixed angles being 90°; see [5].

In the 3D cube grid, NP-hardness of fixed-angle HP protein folding follows from [1] which proves NP-hardness of embedding a fixed-angle orthogonal equilateral chain of n^3 vertices into an $n \times n \times n$ 3D cube grid. If we make all vertices Hs, then a cube embedding is the best way to maximize H–H adjacencies, as the cube uniquely minimizes surface area where potential adjacencies are lost.

In this paper, we prove that the fixed-angle HP protein folding problem is NP-hard in the 2D square grid, even if the chain has only two H vertices and those vertices are its endpoints. In other words, given a fixed-angle orthogonal equilateral HP chain, we prove it is strongly NP-hard to find any planar noncrossing embedding where the endpoints (the two H vertices) are adjacent. This result is tight in the sense that any fixed-angle orthogonal equilateral chain with fewer than two H vertices (and hence can have no H–H adjacencies) has a noncrossing embedding, given by zig-zagging the 90° angles to alternate between left and right turns.

Fixed-angle HP protein folding where only the two endpoints are H vertices is nearly equivalent to finding any planar noncrossing embedding of a ***closed*** fixed-angle chain (where the first and last vertex are identified, and vertices are no longer marked H or P). This is called the ***flattening problem*** for fixed-angle closed chains. The only difference is that, in the flattening problem, the first/last vertex has a fixed-angle constraint, whereas in the HP model, the two necessarily adjacent H vertices could form any angle.

Nonetheless, we show that the flattening problem for fixed-angle orthogonal equilateral closed chains is strongly NP-complete. Past work proved strong NP-hardness when this problem was generalized to fixed-angle orthogonal equilateral caterpillar tree (instead of a chain) or when we allow nonorthogonal fixed angles

*Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, edemaine@mit.edu

†School of Informatics and Engineering, The University of Electro-Communications, itohiro@uec.ac.jp

‡Cheriton School of Computer Science, University of Waterloo, jayson.lynch@uwaterloo.ca

§School of Information Science, JAIST, uehara@jaist.ac.jp

(and working off-grid) [6], but left this case open.

Finally our work also addresses two open problems from [1]. We solve one open problem by proving strong NP-completeness of deciding whether a given fixed-angle orthogonal equilateral chain can be packed into a 2D square (whereas [1] proved an analogous result for a 3D cube). We also prove that this problem remains NP-complete when the chain is only a constant factor longer than the side length of the square (and thus the square is sparsely filled), answering the 2D analog of a 3D question from [1].

2 Preliminaries

A ***linkage*** consists of a ***structure graph*** $G = (V, E)$ and edge-length function $\ell : E \to \mathbb{R}^+$. A ***configuration*** of a linkage in 2D is a mapping $C : V \to \mathbb{R}^2$ satisfying the constraint $\ell(u, v) = \|C(u) - C(v)\|$ for each edge $\{u, v\} \in E$. Let $x(C(u))$ and $y(C(u))$ be the x- and y-coordinate of $C(u)$, respectively. A configuration is ***noncrossing*** if any two edges $e_1, e_2 \in E$ intersect only at a shared vertex $v \in e_1 \cap e_2$.

A linkage is ***equilateral*** if $\ell(e) = 1$ for every $e \in E$. A linkage with n vertices is an ***open chain*** if its structure graph G is a path $(v_0, v_1, \ldots, v_{n-1})$, and it is a ***closed chain*** if G is a cycle $(v_0, v_1, \ldots, v_{n-1}, v_n = v_0)$. A ***fixed-angle chain*** is a chain together with an angle function $\theta : V \to [0°, 180°]$, constraining configurations to have an angle of $\theta(v)$ at every vertex v, except for the two endpoints of an open chain. For notational convenience, we define $\theta(v_0) = \theta(v_{n-1}) = 180°$ for an open chain. A fixed-angle chain is ***orthogonal*** if we have $\theta(v_i) \in \{90°, 180°\}$ for every vertex v_i.

The ***embedding problem*** asks to determine whether a given linkage has a noncrossing configuration in 2D. For general linkages, this problem is $\exists\mathbb{R}$-complete [2]. For fixed-angle orthogonal chains, the problem is in NP: given a binary choice of turning left or right at each vertex, we can construct an embedding (say, placing the first vertex at the origin and the second vertex on the positive x axis), and check for collisions and (for closed chains) closure. In fact, for fixed-angle orthogonal *open* chains, every instance is a "yes" instance:

Observation 1 *Every fixed-angle orthogonal open chain has a noncrossing configuration.*

Proof. Intuitively, we embed the chain in a zig-zag. Precisely, let $P = (v_0, v_1, \ldots, v_{n-1})$ be the path structure graph. First we put v_0 at $(0, 0)$, and v_1 at $(1, 0)$. For each $i = 2, 3, \ldots, n-1$, we define $x(C(v_i))$ and $y(C(v_i))$ as follows. When $\theta(v_i) = 180°$, we have no choice: $x(C(v_i)) = x(C(v_{i-1})) + (x(C(v_{i-1})) - x(C(v_{i-2})))$ and $y(C(v_i)) = y(C(v_{i-1})) + (y(C(v_{i-1})) - y(C(v_{i-2})))$. When $\theta(v_i) = 90°$ and $\overrightarrow{C(v_{i-2})C(v_{i-1})}$ is horizontal, we define $x(C(v_i)) = x(C(v_{i-1}))$ and $y(C(v_i)) = y(C(v_{i-1})) + 1$. If it is vertical, we define $x(C(v_i)) = x(C(v_{i-1})) + 1$ and $y(C(v_i)) = y(C(v_{i-1}))$. The obtained configuration is noncrossing because it proceeds monotonically in x and y, with strict increase in one of the coordinates. □

We note that Observation 1 holds for any fixed-angle orthogonal open chain which is not necessarily equilateral.

In the ***HP model***, the structure graph $G = (V, E)$ has its vertices ***bicolored*** by a color function $\omega : V \to \{H, P\}$. For a configuration C of an equilateral orthogonal linkage, a pair (u, v) of vertices forms an ***H–H contact*** if $\omega(u) = \omega(v) = H$, $\|C(u) - C(v)\| = 1$, and $\{u, v\} \notin E$. The ***HP optimal folding problem*** of a bicolored fixed-angle orthogonal equilateral chain asks to find a noncrossing configuration of the linkage in 2D that maximizes the number of H–H contacts.

A variant of the standard 3SAT problem is ***planar 3SAT***, where the graph $G_\phi = (C \cup V, E)$ of the variable set V and clause set C in a 3SAT formula ϕ, with edges between variables and the clauses that contain them, has a planar embedding. We use a variant of planar 3SAT with additional planarity restrictions: if we add edges to form a Hamiltonian cycle κ of $C \cup V$ that first visits all elements of C and then all elements of V, the resulting graph $G'_\phi = G_\phi \cup \kappa$ must also be planar. The ***linked planar 3SAT problem*** asks, given ϕ, G_ϕ, and κ, whether ϕ is satisfiable. Pilz [8] proved this problem NP-complete.

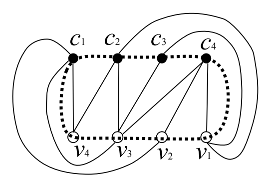


Figure 1: An example instance of linked planar 3SAT, where $c_1 = (\neg v_2 \vee \neg v_3 \vee \neg v_4)$, $c_2 = (v_4 \vee v_3 \vee \neg v_1)$, $c_3 = (\neg v_3 \vee v_1)$, and $c_4 = (v_1 \vee v_2 \vee v_3)$. Hamiltonian cycle κ (drawn dotted) visits $c_1, c_2, c_3, c_4, v_1, v_2, v_3, v_4$ in cyclic order.

3 Embedding Fixed-Angle Orthogonal Equilateral Closed Chains is Strongly NP-complete

In contrast to Observation 1, not all fixed-angle orthogonal equilateral *closed* chains are "yes" instances of the embedding problem. In particular, an orthogonal equilateral closed chain must have an even number of edges to have a configuration in 2D. Even with this property, the length-8 chain $(v_0, v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8 = v_0)$ with angles $\theta(v_2) = \theta(v_6) = 180°$ and $\theta(v_i) = 90°$ for $i = 0, 1, 3, 4, 5, 7$ has configurations in 2D but they have crossings at vertices v_2 and v_6. It is not difficult to show that the embedding problem for fixed-angle orthogonal closed chains is weakly NP-hard by a reduction from the ruler folding problem (see [7, Chap. 2]); this construction requires exponential edge lengths (or equilateral chains with exponentially long straight sections). In this section, we prove that the embedding problem is strongly NP-complete:

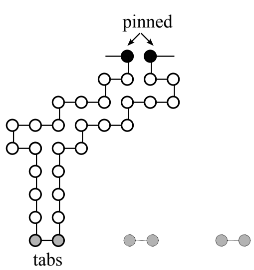
Theorem 1 *Embedding a fixed-angle orthogonal equilateral closed chain in 2D is strongly NP-complete.*

Proof. (Outline.) Section 2 argued membership in NP. To show NP-hardness, we reduce from the linked planar 3SAT problem. We are given a formula ϕ, the associated graph $G_\phi = (C \cup V, E)$, and a Hamiltonian path κ visiting $c_1, c_2, \ldots, c_m, v_1, v_2, \ldots, v_n$ in cyclic order. Because $G_\phi \cup \kappa$ is planar, there is a planar embedding with the clauses $c_1, c_2, \ldots, c_m$ along a single horizontal line from left to right, and the variables $v_1, v_2, \ldots, v_n$ along a lower horizontal line from right to left, as in Figure 1, but with edges routed via orthogonal paths. We can find such an embedding in polynomial time. Note that each edge is either interior or exterior to κ. We can assume that every variable v_i has an incident interior edge and an incident exterior edge, by adding appropriate always-satisfiable clauses $(v_i \vee v_i \vee \neg v_i)$ to κ so that an edge to v_i preserves planarity.

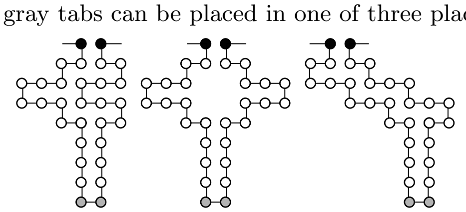
We construct four gadgets that we compose according to the embedding of G_ϕ and κ: the clause gadget, hook gadget, variable gadget, and frame gadget. Some gadgets assume ***pinned*** vertices that cannot move in the plane; we will discuss why they are effectively pinned when we combine the gadgets together.

Figure 2 illustrates the ***clause gadget***. We call the two gray vertices the "tabs" of this gadget. When black vertices are pinned and must turn downward, the tabs have three locations they can be placed. When we flip all vertices in the gadget along the horizontal line through two black vertices, we have three other symmetric options above the horizontal line.

We surround each clause gadget with a ***hook gadget***, as shown in Figure 3, which consists of an upper

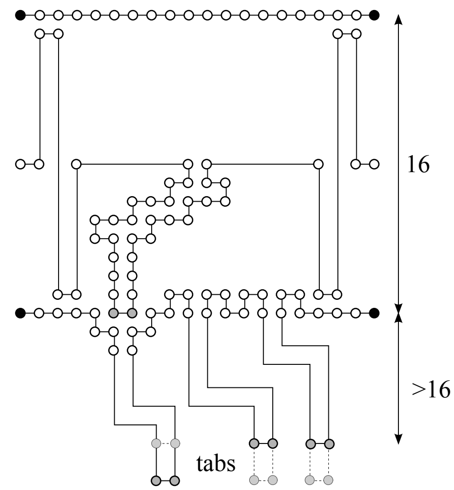


(a) When black vertices are pinned and forced to turn down, the two gray tabs can be placed in one of three places.

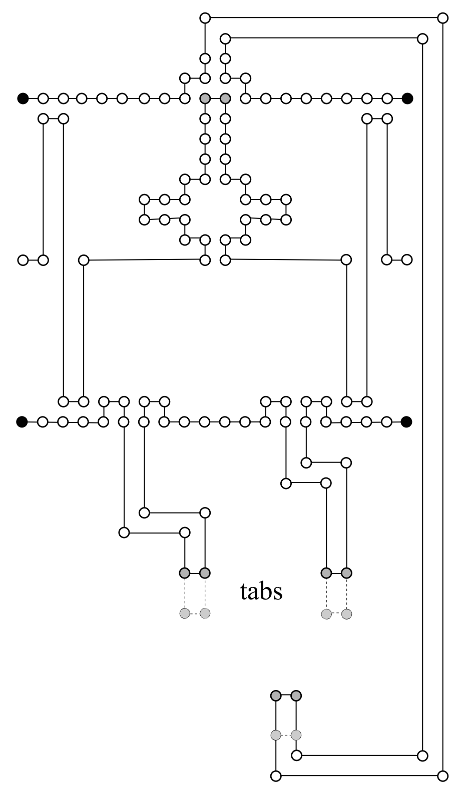


(b) Representative configurations (modulo reflection).

Figure 2: Clause gadget.



(a) Hook gadget with three options on the lower half.



(b) Hook gadget with two options on the lower half and one option on the upper half.

Figure 3: Two versions of the hook gadget. (Some vertices are not drawn to simplify the figure.)

half and a lower half to receive tabs of the clause gadget at distance 8 from the pinned vertices of the clause gadget. Again we assume that both endpoints of these upper and lower halves are pinned, which are depicted by black vertices. We add long flaps beside the clause gadget to prevent it from shifting vertically (relative to the hook gadget). The hook gadget limits the clause to three of its six options, which we arrange to be on the upper or lower halves according to which incident edges of the graph are exterior or interior to κ respectively (see Figure 1). We illustrate the two possibilities of this split modulo reflectional symmetry.

In Figure 3a, the three upper options of the clause are prevented by the upper half which is just a horizontal line, which would cross the clause tabs if the tabs were on the upper half. The lower half consists of three subgadgets, each with their own pair of tabs. When the clause gadget chooses one of the downward options for its tabs, it forces the tabs of the corresponding subgadget to be ***extended*** down by 2 (to avoid crossing), while the other tabs can remain ***retracted*** (which will always be better for avoiding crossings). (The figure shows the unused alternate state with dashed lines.) Each pair of tabs in the hook gadget has distance more than 16 from the clause gadget, and the linkage to the tabs is a doubled zig-zag; together, these guarantee that the tabs of a hook gadget cannot be flipped up because this would cross with the upper half. The doubled zig-zag also prevents the tabs from flipping horizontally. Thus each pair of tabs has exactly two placements (retracted and extended) if the black vertices are pinned.

In Figure 3b, one lower option of the clause (the middle) is prevented by the lower half being horizontal there, while the corresponding upper option is allowed by adding a subgadget to the upper half. Using the same arguments, the pair of tabs of the subgadget on the upper half has two exactly placements: retracted and extended. When the clause gadget chooses the available upper option, the pair of tabs of the subgadget is forced to be extended *up* by 2, which is the opposite of each subgadget on the lower half. Moreover, we arrange that no pair of doubled zig-zag corridors to tabs have the same height.[1]

Figure 4 illustrates the ***variable gadget***. The variable gadget for a variable v consists of two zig-zag paths of length $4k+3$, where k is the number of appearances of v or $\neg v$ as a literal in clauses. The two zig-zag paths are joined by a horizontal baseline, which separates the upper and lower zig-zag paths, forcing only two possible embeddings: the one in the figure and its reflection through the baseline. Both zig-zag paths contain a horizontal segment of length 4 for each appearance of the variable. The heights of the segments on the upper and lower zig-zag paths, measured from the baseline, are either 3 and -1 respectively, or 1 and -3 respectively. Which option depends on whether the corresponding literal uses the variable positive or negated, and on whether the corresponding tab of the hook gadget comes from above or below the variable gadget (which corresponds to whether the tab is from the upper or lower half of the clause/hook gadget, i.e., whether the graph edge is exterior or interior to κ). The heights are $(1,-3)$ if and only if either the literal is v and the tab comes from above, or the literal is $\neg v$ and the tab comes from below; in Figure 4, these are the first, third, and fourth pairs of horizontal segments. In the other cases, the heights are $(3,-1)$.

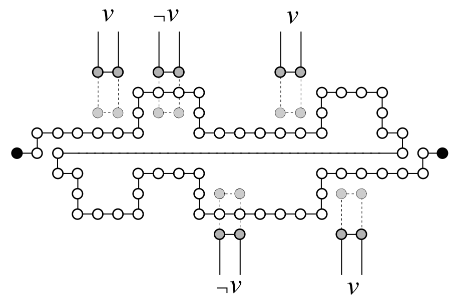


Figure 4: A variable gadget for a variable v that appears five times as v, $\neg v$, $\neg v$, v, and v. The corresponding tabs come from above, above, below, above, and below.

We arrange the variable gadgets with different heights (see Figure 6) so that the minimum vertical distance between two baselines of two variable gadgets is at least $4n$. This minimum distance guarantees that no pair of horizontal segments in variable gadgets for v_i and v_j with $i \neq j$ has the same height, which may cause an unexpected flip between them. In addition, our assumption that every variable has connections to clauses both above and below it means that there is a tab both above and below the variable, forcing an approximately correct height of the baseline.

The last gadget is the ***frame gadget***, shown in Figure 5, which surrounds all other gadgets. For a given closed chain, we consider the minimum rectangle that contains but does not intersect the chain (one step outside the bounding box on all sides). Then we remove an

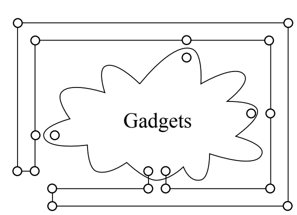


Figure 5: Frame gadget for closed chains.

[1]Otherwise, unexpected pairs of adjacent corridors may be flipped. For example, consider the pair indicated by an arrow at the top of Figure 6. If these two corridors have the same height, the linkage joining the pair can be flipped up locally.

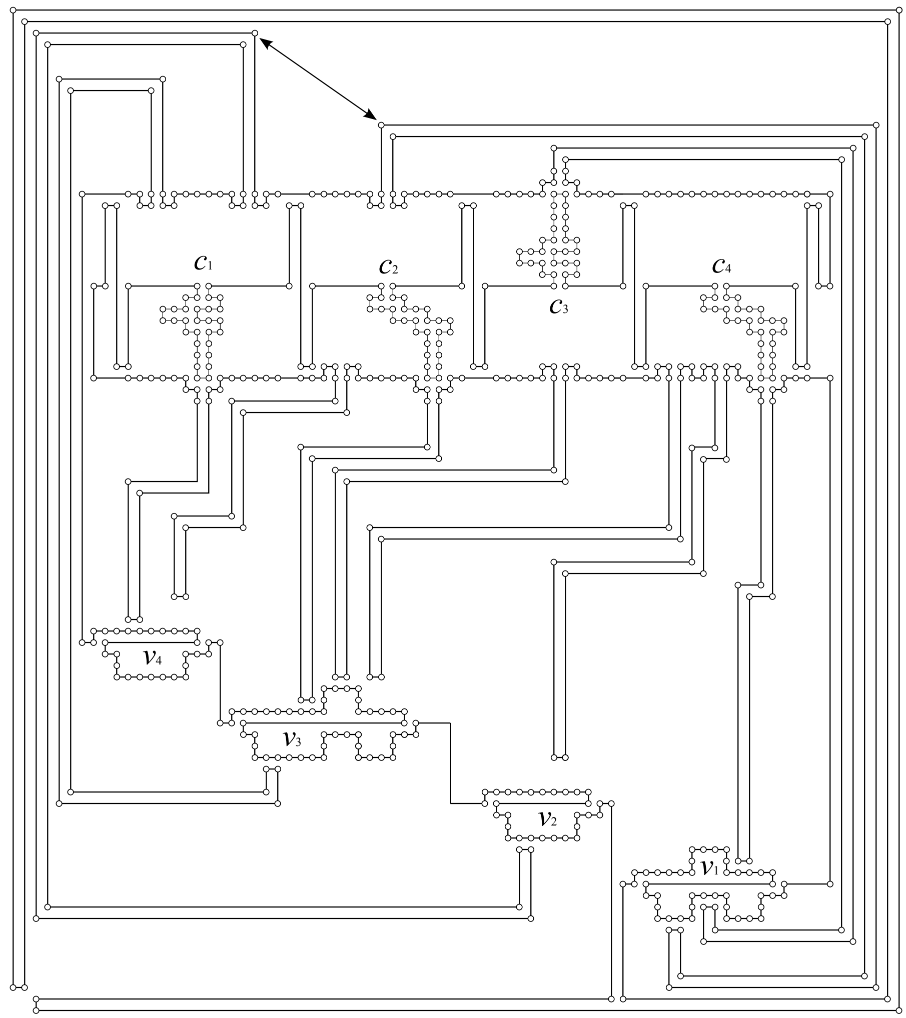


Figure 6: An example of the reduction from the instance in Figure 1, and the solution embedding corresponding to assignment v_1 = true, v_2 = true, v_3 = true, and v_4 = false, where clauses c_1, c_2, c_3, and c_4 choose the variables v_4, v_3, v_1, and v_1 as their true literals, respectively. (Note: vertical distances between two gadgets are not to scale.)

extreme edge $\{u, v\}$ in the gadgets, and attach the frame gadget that essentially doubles the minimum rectangle, as shown in Figure 5. The inside of the frame gadget includes the minimum rectangle, except for three edges, as part of the chain. The doubling prevents any part of the frame from being flipped with respect to the surrounded gadgets. This frame also inhibits the surrounded gadgets from illegal flips to outside the minimum rectangle.[2]

Figure 6 shows how all the gadgets fit together for the example instance from Figure 1. We join together all upper halves of hook gadgets for $c_1, c_2, \ldots, c_m$; all clause gadgets (and their flaps) for $c_m, c_{m-1}, \ldots, c_1$; all lower halves of the hook gadgets for $c_1, c_2, \ldots, c_m$; and all variable gadgets for $v_1, v_2, \ldots, v_n$, in these orders. Finally, we attach the frame gadget by replacing an edge on a path joining the upper halves of the hook gadgets, or an edge on a path joining the variable gadgets.[3]

This reduction can be done in time polynomial in the size of ϕ. It remains to show that an instance (ϕ, G_ϕ, κ) of linked planar 3SAT is satisfiable if and only if the resulting fixed-angle orthogonal equilateral closed chain has a planar embedding. Due to the space limitation, we only outline the proof.

When the linked planar 3SAT instance is satisfiable, at least one literal of each clause is satisfied by the assignment. The clause gadget then chooses the corresponding tabs of the corresponding hook gadget and extends it, while retracting the other tabs. The extended tabs force the corresponding variable gadget to take the true position, to avoid crossing. Because the assignment is satisfiable, all variable gadgets can avoid

[2] In the most common case, including the example in Figure 6, the frame is not necessary, as the hook gadgets will wrap around both sides of the construction.

[3] We omit the case that no edge can be seen from the outside of these gadgets.

crossing with tabs. On the other hand, when the loop has an embedding, all gadgets must be inside the frame gadget. Each clause gadget then has to indicate some tabs to be extended. Because the corresponding variable does not have any crossings, the corresponding variable satisfies the clause. Therefore, the instance of the linked planar 3SAT is satisfiable. □

4 HP Optimal Folding a Fixed-Angle Orthogonal Equilateral Open Chain is Strongly NP-complete

We now turn to orthogonal equilateral open chains in the HP model, where the vertices are bicolored H or P, and we wish to find a noncrossing configuration in 2D that maximizes the number of H–H contacts. In this section, we prove that this problem is NP-complete, despite the chain being open:

Theorem 2 *HP optimal folding of a bicolored fixed-angle orthogonal equilateral open chain is strongly NP-complete, even if the chain has just two H vertices.*

Proof. We use the same reduction in the proof of Theorem 1, except for the frame gadget, which we replace with Figure 7. The inside of the frame gadget covers the minimum rectangle except two edges, but now the bottom doubled edge extends very far to the left, more than 10 times the total length L of all other gadgets. The leftmost two vertices of the bottom doubled edge are H (and the chain is not closed there), and all other vertices in the chain are P.

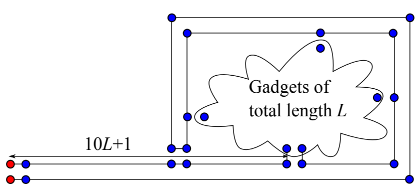


Figure 7: A frame gadget for an HP chain. The two H vertices are drawn red at the far left.

This reduction can be done in polynomial time. Thus it suffices to show that this arrangement of the frame is the only way to obtain the H–H contact at the two H vertices. Because the total length of the gadgets inside of the frame is at most L, we must arrange the two long segments attached to the H vertices in parallel as shown in the figure to make the H–H contact. Thus we must put all other gadgets inside the frame, and hence correctness follows from the proof of Theorem 1. □

5 Packing Fixed-Angle Orthogonal Equilateral Open Chains into Squares is Strongly NP-complete

We now address some of the open questions from [1]. First, the authors ask whether a fixed-angle orthogonal equilateral open chain (or in their terminology, an S–T sequence of squares, where each S square must continue straight and each T square must turn left or right) can be packed into a 2D square. Second, they ask whether the problem remains hard when the chain occupies a small fraction of the volume of the target shape. (They ask this question for the 3D version of the problem, but it naturally extends to the 2D version we consider.) We answer both questions by showing that packing a fixed-angle orthogonal equilateral open chain of length $O(s)$ into an $s \times s$ square is strongly NP-complete. This result is tight up to constant factors: if the chain has length $< s$, then it can be packed into an $s \times s$ square via Observation 1.

Theorem 3 *Embedding a given fixed-angle orthogonal equilateral open chain into an $s \times s$ square is strongly NP-complete, even if the chain has length $O(s)$.*

Proof. We use the same reduction in the proof of Theorem 2, except for the frame gadget, which we replace with Figure 8.

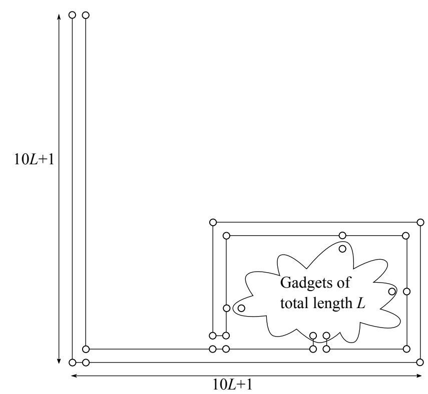


Figure 8: A frame gadget for an open chain which must fit in a $10L + 1$ by $10L + 1$ square.

This frame gadget starts the chain with two connected segments of length s. Any embedding into the $s \times s$ square must place these segments along two boundary edges of the square, say left and bottom as in the figure. The next two segments on the outside of the frame gadget must turn left to remain within the square. At the other end of the chain, we have a vertical (by parity) segment of length $s - 1$ and a horizontal segment of length $> 9L$, which forces these segments against the first two segments. With these segments in place, the prior argument ensures that the rest of the frame and thus the rest of the gadgets are correctly placed.

The chain has length at most $47L + 6$ vertices (from the SAT gadgets, the smaller frame, and the three long bars). Thus the length $l = O(s)$. □

It remains open whether the problem of *densely* packing a fixed-angle orthogonal equilateral open chain of length s^2 into an $s \times s$ square is NP-complete.[4] The analogous problem in 3D is strongly NP-complete [1].

Acknowledgments

This work was initiated at the 3rd Virtual Workshop on Computational Geometry held in March 2022. We thank the other participants of that workshop — in particular Martin Demaine, David Eppstein, Timothy Gomez, and Aaron Williams — for helpful discussions and for providing a fruitful collaborative environment.

References

[1] Z. Abel, E. D. Demaine, M. L. Demaine, S. Eisenstat, J. Lynch, and T. B. Schardl. Finding a Hamiltonian path in a cube with specified turns is hard. *Journal of Information Processing*, 21(3):368–377, 2013.

[2] Z. Abel, E. D. Demaine, M. L. Demaine, S. Eisenstat, J. Lynch, and T. B. Schardl. Who needs crossings? Hardness of plane graph rigidity. In *Proceedings of the 32nd International Symposium on Computational Geometry (SoCG 2016)*, pages 3:1–3:15, Boston, Massachusetts, June 2016.

[3] B. Berger and T. Leighton. Protein folding in the hydrophobic-hydrophilic (*HP*) model is NP-complete. *Journal of Computational Biology*, 5(1):27–40, 1998.

[4] P. Crescenzi, D. Goldman, C. Papadimitriou, A. Piccolboni, and M. Yannakakis. On the complexity of protein folding. *Journal of Computational Biology*, 5(3):423–465, 1998.

[5] E. D. Demaine, M. L. Demaine, A. Hesterberg, Q. Liu, R. Taylor, and R. Uehara. Tangled tangles. In *The Mathematics of Various Entertaining Subjects (MOVES 2015)*, volume 2, pages 141–152. Princeton University Press, 2017.

[6] E. D. Demaine and S. Eisenstat. Flattening fixed-angle chains is strongly NP-hard. In *Proceedings of the 12th Algorithms and Data Structures Symposium (WADS 2011)*, pages 314–325, Brooklyn, New York, August 2011.

[7] E. D. Demaine and J. O'Rourke. *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*. Cambridge University Press, 2007.

[8] A. Pilz. Planar 3-SAT with a Clause/Variable Cycle. In *16th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT 2018)*, pages 31:1–31:13. LIPIcs, Dagstuhl, 2018.

[4] In this context, a packing of a square is *densely* if the chain covers all grid points in the square.