An Algorithm for Folding 3D Orthogonal Graphs with Non-Zero Edge Thickness

Abstract

We present an algorithm that generates crease patterns for folding any 2D orthogonal graph out of a rectangular sheet of paper. The resulting model, which is both efficient and watertight, represents the 2D graph as a 3D extrusion with any desired edge height and edge thickness. In addition, we prove that the crease patterns generated by our algorithm represent models with optimal efficiency, meaning that they achieve the lowest possible scale factor from the original sheet of paper to the folded graph. The efficiency and flexibility of the models generated by our algorithm suggest that it has important implications for a range of engineering applications that aim to create materially efficient, easily manufacturable designs.

1 Introduction

Origami is the traditional Japanese art of folding paper. Originally practiced as a creative art form, nowadays, origami often serves as an inspiration for various engineering challenges, such as the expandable solar arrays created by Zirbel et al. (2015), due to its core principle of creating complex structures from a single flat, uncut sheet of material. By applying geometry to origami designs, one can find ways to fold a model with given dimensions using the smallest possible sheet of material, that is, in the most efficient way. Origami's potential in engineering also stems from the ability of specific models to expand or contract laterally or radially. In addition, some models can adopt multiple stable configurations, making origami well-suited for designing compliant mechanisms that open and close, among other functions.

What also makes origami particularly suitable for engineering is its simplicity. All origami models can be folded with only two basic folds: mountain folds and valley folds, both of which are shown in Figure 1. More complicated folds, such as reverse folds, pleats, or sinks, are combinations of mountain and valley folds. All crease patterns shown in this paper utilize only mountain and valley folds, the former in red and the latter in blue.

Many properties of origami models, including the efficiency of models and even the foldability of crease patterns, are constrained by the choice of materials, folding methods, and underlying geometric principles. For example, we prove later in our paper that the crease patterns generated by our algorithm are the most efficient out of any hypothetical alternative that creates the same models. Although there are limitations on what is or is not foldable out of a flat sheet of material, like paper or sheet metal, a wide variety of designs remain foldable within the constraints of origami. Moreover, it is often more materially efficient to create a design from one flat sheet of material rather than multiple pieces joined by hinges or joints. The presence of these constraints suggests that origami designs can be rigorously analyzed and quantified.

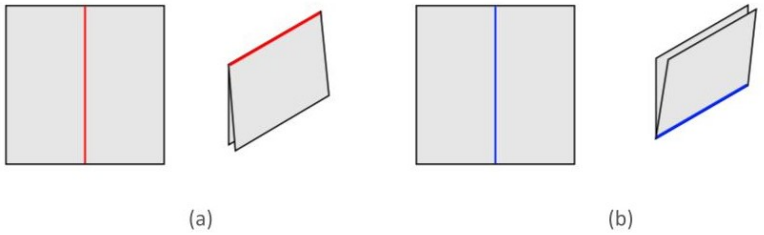


Fig. 1 Example of mountain fold and valley fold. The left panels of (a) and (b) show the crease pattern of a single mountain or valley fold, respectively, and the right panels show the folded models following these crease patterns. Mountain folds are commonly shown as red in crease patterns, and valley folds are commonly shown as blue in crease patterns. These coloring schemes are followed in the rest of paper.

The quantitative nature of origami highlights the valuable role that computing systems can play in its design. Computational origami design now presents an opportunity to design complex, efficient, and foldable structures using fundamental building blocks. Researchers have explored this potential through algorithms that generate foldable crease patterns for orthogonal graphs. Demaine, Demaine, and Ku (2010) and Demaine, Ku, and Yoder (2018) have developed origami algorithms to create orthogonal, triangular, and hexagonal graphs with zero edge thickness. These algorithms can take any 2D graph and generate a crease pattern for a corresponding 3D origami structure capable of forming mazes, paths, or even letters of the alphabet, for example. The algorithms, runnable in linear time, work by layering different ”gadgets” onto the 2D graph, as shown in Figure 2. These gadgets are folded 3D representations of all possible 2D edge intersections.

In general, this class of algorithms is evaluated by three key properties: *seamlessness*, *watertightness*, and *efficiency*.[1] *Seamlessness* refers to the condition where every visible face of the model consists of a single, uncreased layer of paper. *Watertightness* means that the model in its stable state (without glue, tape, or other materials) can hold water without leaking. *Efficiency* is quantified by the scale factor, defined as the ratio between the dimensions of the original sheet of paper and those of the folded model. A smaller scale factor indicates a more efficient crease pattern.

[1] We will formally define these properties in Section 2.

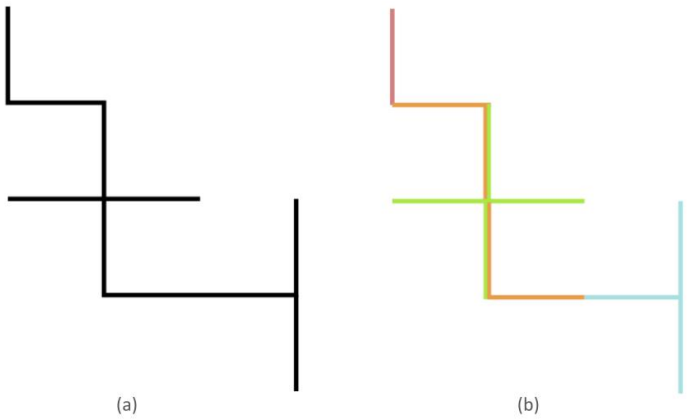


Fig. 2 Example of breaking down a 2D graph shown into different gadgets. (a) shows the 2D graph, and (b) shows the layered gadgets. Each colored segment represents a different gadget with a different number of edge intersections. Line segments with multiple colors represent areas where two different gadgets overlap.

In this paper, we present an algorithm that generates foldable crease patterns of 3D orthogonal graphs with non-zero edge thickness, using a single, uncut rectangular sheet of paper. While algorithms developed in prior studies are limited to models with zero edge thickness, our algorithm generates models with non-zero edge thickness. Figure 3 shows an example of a 3D origami graph generated with zero versus nonzero edge thickness. Our algorithm's ability to incorporate non-zero edge thickness expands the real world applications of origami graphs. For example, if existing algorithms are used to design any engineering models that involve electronics, the resulting models with zero-thickness edges will not be able to house wires or circuit boards. In contrast, our algorithm, which produces models with non-zero edge thickness, can be helpful for designing precise and snug wire housings, for example. This opens up practical applications in civil or mechanical engineering, where such models can serve as efficient housings for electronics, fluid channels, or energy-absorbing components.

The models produced by our algorithm are optimally efficient in that, for a given 3D graph with a specific edge thickness and edge height, they achieve the lowest scale factor possible (Demaine, Ku, and Yoder, 2018). The introduction of edge thickness into our gadgets adds an additional layer of complexity to our efficiency

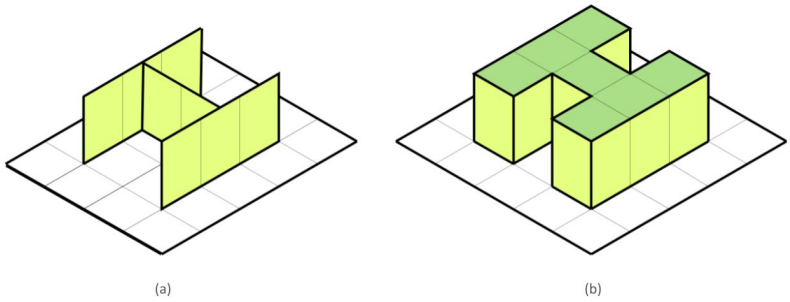


Fig. 3 Comparison between 3D origami models of "H" with zero versus non-zero edge thickness. (a) shows the "H" created by Demaine, Demaine, and Ku (2010), which has zero edge thickness. (b) shows the "H" created by our algorithm, which has a nonzero edge thickness. The walls of the edges are in yellow while the faces of the edges are in green.

analysis because the scale factor now depends on edge thickness as well as edge height.

Our models are also watertight (Demaine and Tachi, 2017). Since there are no loose edges in the folded model, it is able to hold water without leaking. Watertight models, in addition to being more suitable for real-world applications, are also generally deemed more practical to fold because they contain fewer regions with overlapping layers of paper. However, unlike models based on graphs with zero edge thickness, our models are not seamless (Demaine, Ku, and Yoder, 2018). This complication arises because our edges do not have zero thickness. As we will prove later, it is impossible for any 3D graph with non-zero edge thickness to be completely seamless.

Additionally, our algorithm can generate models not foldable using zero-thickness algorithms. As shown in Figure 4 (a), segments of models generated by zero-thickness algorithms must be continuous and connected orthogonally. On the other hand, our algorithm can create continuous diagonal segments, as shown in Figure 4 (b). The individual squares shown in Figure 4 (b) can also be spaced further apart to create discrete, unconnected segments, which are also not foldable with zero-thickness algorithms. This property of our algorithm further enhances its

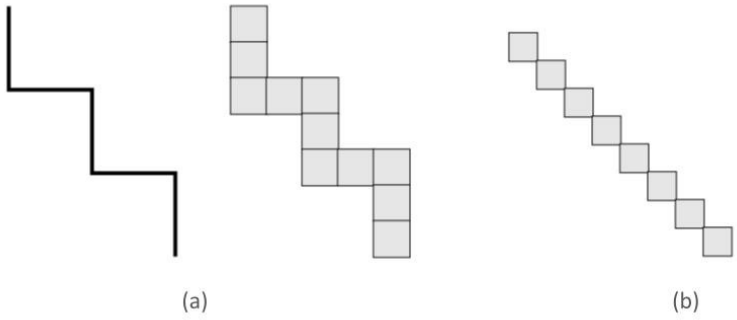


Fig. 4 Different ways of creating a diagonal line using zero edge-thickness and non-zero edge-thickness graphs. (a) shows a method of creating a diagonal line in a zero edge-thickness graph that is replicable in a non-zero edge-thickness graph. (b) shows another method of creating a diagonal line that is unique to non-zero edge-thickness graphs and cannot be created using zero-edge thickness crease patterns.

applicability in real-world settings. The combination of geometric efficiency, structural stability, and adaptability makes such origami-inspired designs well-suited for a wide range of functional and scalable engineering applications.

In Section 2, we formally define the basic terminologies in origami. Section 3 then presents our algorithm for generating foldable crease patterns corresponding to any 2D orthogonal graph. Section 4 provides proofs for the scale factor and seamlessness of the models generated by our algorithm. Finally, Section 5 provides conclusions.

2 Definitions

Before developing the algorithm, we formally define several standard origami-related terminologies that will be referenced throughout the rest of the paper.

Definition 1 : A **mountain fold** is a fold created such that, relative to the original, flat sheet of paper, the resulting crease juts upward out of the paper and forms a ”ridge.” An example is shown in Figure 1 (a).

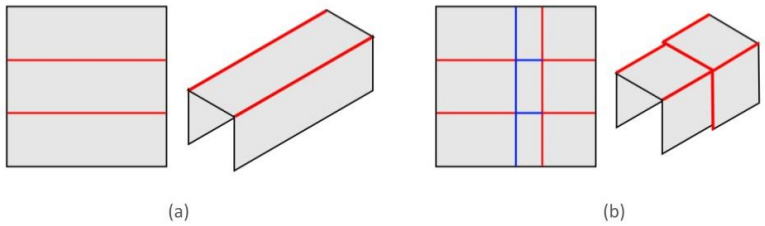


Fig. 5 Examples of seamless and nonseamless models. In (a), the faces are seamless because each face is uncreased (the only creases form the edges) and is covered by only one layer of paper. In (b), the faces are not seamless because they are covered by more than one layer of paper. Since in (a), every visible face is seamless, (a) is also considered a seamless model.

Definition 2 : The opposite of a mountain fold, a **valley fold** is a fold created such that, relative to the original, flat sheet of paper, the resulting crease protrudes downward and forms a ”valley.” An example is shown in Figure 1 (b).

Definition 3 : A **seamless face** is a flat face in the folded model that is uncreased and consists of only one layer of paper. An example of a model with only seamless faces is shown in Figure 5 (a), and a similar model with only non-seamless faces is shown in Figure 5 (b).

Definition 4 : A **seamless model** is a folded model in which all visible faces of the model are **seamless faces** (Demaine, Ku, and Yoder, 2018). Usually, since many models contain at least one seamless face, a **seamless model** can also refer to a folded model where designated ”major” faces of the model, e.g., the walls of folded 3D orthogonal graphs, are seamless.

Definition 5 : An **efficient model** is a model in which the scale factor from the dimensions of the unfolded sheet of paper to the dimensions of the folded model is reasonably small (Demaine, Ku, and Yoder, 2018).

Definition 6 : An **optimally efficient model** is obtained when no other crease pattern associated with the same folded model can achieve a smaller scale factor from the unfolded paper to the folded model.

Definition 7 : A **watertight model** is a model in which the boundaries of the folded model map to the boundaries of the original sheet of paper. In other words, the outer edge of the folded model is simply a collapsed version of the outer edge of the original sheet of paper. Figures 6 (a) and (b) illustrate a non-watertight model, and Figures 6 (c) and (d) illustrate a similar but watertight model.

Definition 8 : In a folded, 3D graph, an **open edge** is an edge in the folded model where the section of paper that forms the edge is part of the boundary of the unfolded paper but is *not* mapped to the boundary of the folded model. An example of an open edge is shown in Figure 7 (b), where the thick green line indicates the section of paper in question. No model with open edges is watertight.

Definition 9 : In a folded, 3D graph, a **closed edge** is an edge in the folded model where the section of paper that forms the edge is part of the boundary of the unfolded paper and *is* mapped to the boundary of the folded model. To be watertight, a folded, 3D orthogonal graph must have closed edges only. An example of a closed edge is shown in Figure 7 (c).

Definition 10 : A folded, 3D graph is called **capped** if all edges are closed. All capped graphs are watertight. Performing the specific set of folds that turns an open edge into a closed one is called **capping** the edge.

3 Algorithm

Our algorithm generates crease patterns using gadgets like those shown in Figure 2. These gadgets are small, folded 3D models that represent the set of all possible edge intersections in a 2D orthogonal graph. By layering the crease patterns of specific gadgets over every edge intersection in the 2D graph, a crease pattern for a 3D folded model resembling the 2D graph can be generated. This method ensures that our algorithm runs in linear time.

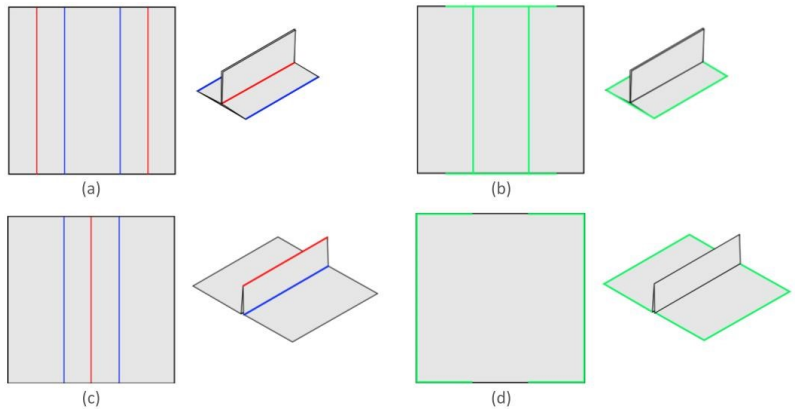


Fig. 6 Examples of watertight and non-watertight models. Panel (a) shows a non-watertight model's crease pattern. Panel (b) highlights in green the boundaries of the non-watertight, folded model, which do not fall along the edges of the original, unfolded paper. In contrast, panels (c) and (d) illustrate a watertight model whose boundaries fall along the edges of the original, unfolded paper.

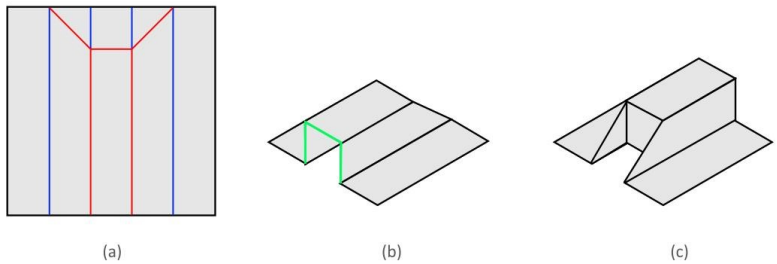


Fig. 7 A model with both an open edge and a closed edge. (a) shows the crease pattern of a model with both an open edge and a closed edge. (b) illustrates the open edge, where the part highlighted in green is part of the boundary of the unfolded paper but not the boundary of the folded model. (c) illustrates the closed edge, where every part of the edge is both part of the boundary of the unfolded paper *and* the boundary of the folded model.

3.1 Gadgets

Most properties of the final, folded 3D graph, such as seamlessness and efficiency, depend on the properties of the gadgets themselves[2]. If a gadget is not seamless, then a folded graph can never be seamless.

[2]Note that watertightness is not on the list of properties dependent on the individual gadgets. This is because the watertightness of a model is *not* affected by the watertightness of the gadgets, as watertightness only concerns the *boundary* of the folded model. In addition, any non-watertight gadget can be capped to form a watertight model.

The inverse statement is not true, however. Even if all the gadgets are seamless, a folded graph may not always be seamless. This is because some properties of the final, folded 3D graph also depend on the intersections between the gadgets themselves; if a folded graph is created with seamless gadgets but non-seamless gadget intersections, then it is non-seamless. However, our algorithm is not affected by this slight nuance, as our gadget intersections are all seamless. This means that, for the folded graphs generated by our algorithm, the seamlessness of the folded graph only depends on the seamlessness of the gadgets.

The efficiency of a folded 3D graph also depends on the gadgets themselves. The smallest possible scale factor from an unfolded sheet of paper to any folded 3D graph is the scale factor of a single capped gadget (assuming all the gadgets have the same scale factor; otherwise, the smallest possible scale factor would be that of the most efficient gadget).[3]

In orthogonal 3D graphs with non-zero edge thickness, there are seven possible gadgets. Since the graphs are orthogonal, four possible edges can intersect in a gadget. Thus, there should be $2^4 = 16$ possible gadgets. However, after accounting for rotation, there are only 6 possible gadgets, plus one "special gadget," all of which are shown in Figure 8.

The crease patterns for all the gadgets are shown in Figure 9. These crease patterns, with the exception of the special, $n = 0$ gadget, all have open edges of the form shown in Figure 7. These open edges, as discussed later, can combine seamlessly. In our gadgets, a 5 x 5 grid is folded to produce a 3 x 3 gadget with varying amounts of extruded edges. A common feature of all the extruded gadgets is that the center square in the 5 x 5 grid is always extruded (marked green in Figure 9), and that the four most extreme corners are always non-extruded (marked white in Figure 9); thus creating a 3 x 3 gadget. The remaining squares are either

[3]It is important, however, to make the distinction between the scale factor in an *optimally efficient* folded graph and the *smallest possible scale factor* of *any* folded graph. In other words, the vast majority of folded graphs have an optimal scale factor greater than the smallest possible scale factor of *any* folded graph.

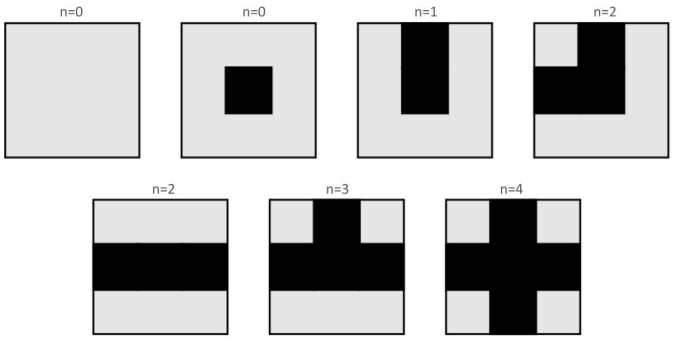


Fig. 8 The seven gadgets used in the algorithm. Filled-in, black squares indicate the region is extruded. The number n above each gadget drawing indicates how many edges are extruded in the gadget. The second gadget marked with $n = 0$ is the "special gadget," where no edges are extruded but the center square is.

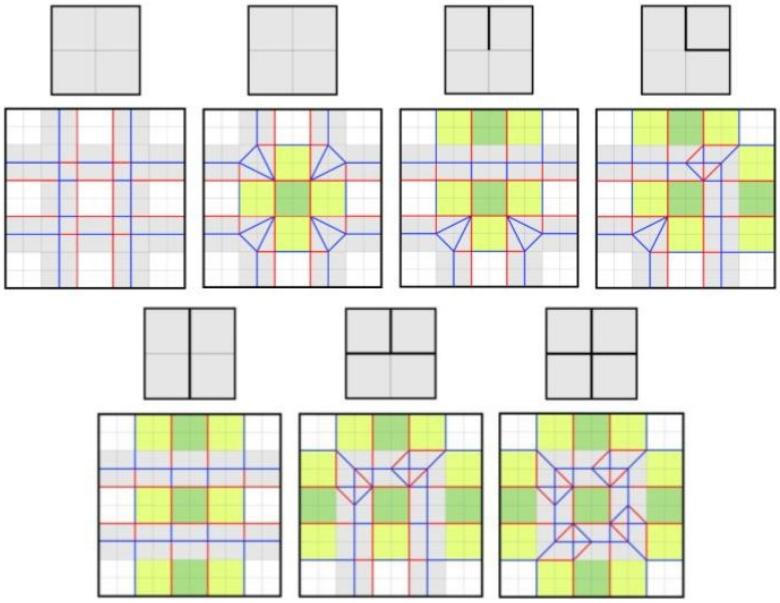


Fig. 9 Crease patterns of the seven gadgets used in the algorithm. Green indicates that the portion of the paper is an extruded face; yellow indicates that the portion makes up the side of an extruded edge; white indicates that the portion is not extruded; and gray indicates collapsed regions of the paper. Red and blue lines indicate mountain and valley folds, respectively.

collapsed layers or make up the walls of the edges; in either case, the scale factor of the gadgets remains the same.

3.2 Varying Edge Height

All the crease patterns shown in Figure 9 have an edge height to thickness ratio of 1:1. These crease patterns can be modified to have different edge height to thickness ratios. There are two different ways to modify these crease patterns: one involving box pleating and one without box pleating. Which of the two methods should be used depends on the desired edge height-to-thickness ratio.

The method without box pleating, which is far simpler than the method with box pleating, can only be used for an edge height to thickness ratio of less than or equal to 2:1. The method involving box pleating can be used for any edge height-to-thickness ratio. However, for edge height to thickness ratios of less than or equal to 2:1, using this method is undesirable because it creates unnecessary complexity and thick layers of paper, which are harder to fold. Thus, the method without box pleating is preferred in this scenario. This method, when applicable, also results in a lower scale factor than the method with box pleating.

3.2.1 Edge height to thickness ratios of less than or equal to 2:1

The crease patterns of the default gadgets shown in Figure 9 can be modified slightly to create new gadgets with different edge height to thickness ratios. The condition for using the method described in this subsection is that the edge height to thickness ratio is less than or equal to 2:1. This also means that creating gadgets with edge thickness to height ratios of less than 1:1, e.g., 1:2 or 1:8, are possible with this method.

These modifications do not require any sinks or box pleats. As a result, using this method, the scale factors of gadgets with different edge height to thickness ratios are different. For example, the scale factor of a gadget with an edge height to thickness ratio of 1:1 versus one with an edge height to thickness ratio of 2:1 are different, with the latter having a larger scale factor. Similarly, gadgets with an edge height to thickness ratio of $1/2$ have a scale factor smaller than $5/3$, which is

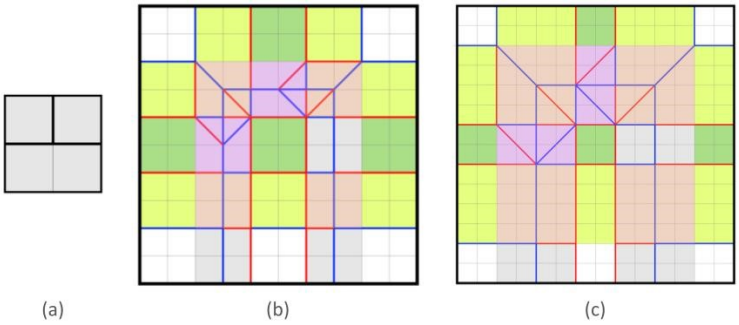


Fig. 10 Comparison of the crease pattern between a gadget shown in Figure 9 and one with identical structure but doubled height. (a) shows a schematic representation of the gadget, (b) shows the crease pattern of the gadget with a 1:1 edge height-to-thickness ratio, and (c) shows the crease pattern of the same gadget but with a 2:1 edge height-to-thickness ratio. Green indicates that the portion of the paper is an extruded face; yellow indicates that the portion makes up the side of an extruded edge; and white indicates that the portion is not extruded. Gray, orange, and purple regions indicate collapsed regions of the paper. Regions with the same color correspond with each other. Red and blue lines indicate mountain and valley folds, respectively.

the scale factor of the gadgets represented in Figure 9 (that have an edge height to thickness ratio of one).

Given that the scale factor of a gadget with an edge height to thickness ratio of 1:1 is $\frac{5}{3}$, a gadget with an edge height to thickness ratio of $h:1$ will have a scale factor s as defined in Equation 1, which shows that each increase of 1 in the edge height to thickness ratio adds a value of $\frac{2}{3}$ to the smallest possible scale factor, 1.

$$s = 1 + \frac{2h}{3}, h \in (0,2] \tag{1}$$

To modify the crease pattern using this method, the areas highlighted in yellow can be stretched or shortened along the axis pointing away from the green squares, as shown in Figure 10, with Figure 10 (b) being modified to Figure 10 (c).

Elongating or shortening the yellow squares affects the rest of the crease pattern minimally because only the other yellow areas or the gray, collapsed areas of the paper are affected by this change. Certain collapsed areas, highlighted in orange in Figures 10 (b) and (c), remain as squares, even after scaling the yellow squares. This means that the sets of folds that collapse the paper in these regions remain the

same, except that they take up more or less area. No matter how long or short the yellow regions are, these areas will always remain squares.

However, other collapsed regions of the paper change in dimensions. This complicates the process by which a crease pattern is modified to reflect variations in its edge height to thickness ratio. Such areas are highlighted in purple in Figures 10 (b) and (c). In the crease pattern of the gadget with a 1:1 edge height to thickness ratio, the region highlighted in purple is a square. However, in the gadget with a 2:1 edge height to thickness ratio, the same region is a two-by-one rectangle.

In order to collapse the paper properly, certain folds must appear in the purple region. Specifically, the purple region must contain both a mountain fold and a valley fold. Both should extend from opposite corners on the right side of the purple region and intersect at a 90-degree angle. In other words, the mountain fold, valley fold, and rightmost or uppermost boundary of the purple region should form a 45°-45°-90° triangle, with the mountain and valley folds as the legs and the boundary of the purple region as the hypotenuse. These folds must form the triangle in the purple region, but strictly not outside (in the orange regions showing in Figures 10 (b) and (c), for example). The single crease seen in the purple region of Figure 10 (b), which extends from the 90° vertex of the 45°-45°-90° triangle, is not necessary for proper collapsing of the hidden layers.

To meet the criteria outlined, the purple region must have at most a height-to-width ratio of 2:1. Since the height and width of the purple region are determined by the height of the edge walls (yellow region) and thickness of the edges faces (green region), this method of varying the edge thickness results in a collapsible model only if the edge height to thickness ratio also does not exceed 2:1.

3.2.2 Edge height to thickness ratios of greater than 2:1

If an orthogonal graph with an edge height to thickness ratio of greater than 2:1, e.g., 4:1 or 6:1, is desired, box pleating must be employed to collapse the paper properly.

There is one core issue underlying creating edge height to thickness ratios of greater than 2:1 using the method presented in the last subsection: the requirement that one valley fold, one mountain fold, and the rightmost or uppermost boundary of the purple-highlighted rectangle must form a 45°-45°-90° triangle within the purple rectangle, as shown in Figures 10 (b) and (c). If a rectangle's height-to-length ratio is greater than 2:1, then the 45°-45°-90° triangle cannot fit within the rectangle, and thus, the crease pattern cannot properly collapse.

This issue in gadget crease patterns with edge height to thickness ratios of greater than 2:1 can be solved by box pleating. As shown in Figure 11 (c), box pleating creates not one but two 45°-45°-90° triangles within the purple rectangle to effectively collapse the model. In other words, in the example shown in Figure 11 (c), box pleating splits the 4:1 purple rectangle into two 2:1 rectangles, which individually can be collapsed properly. However, this folding does result in two times as much thickness in the collapsed layers as in the crease patterns created by the non-box-pleating method.

For any gadget with an $h : 1$ edge height to thickness ratio, where $h > 2$, the purple-highlighted rectangle has dimensions h by 1. Box pleating divides the h by 1 rectangle into p sets of j by 1 rectangles, where $j \in (0,2]$ and $j \cdot p = h$. p must also be an even number, or else the edge walls (marked as yellow in Figure 11) will face inwards instead of outwards. The required number of box pleats can be modeled as a function shown in Equation 2, where any gadget with an $h : 1$ edge height to thickness ratio will require p box pleats to be foldable:

$$p = \lceil \frac{h}{4} \rceil \cdot 2 \tag{2}$$

Consequently, the thickness of the edges will increase by a factor of p as a result of the increased box pleating. The more we increase the edge height to thickness ratio of a gadget, the thicker the resulting edges will be.

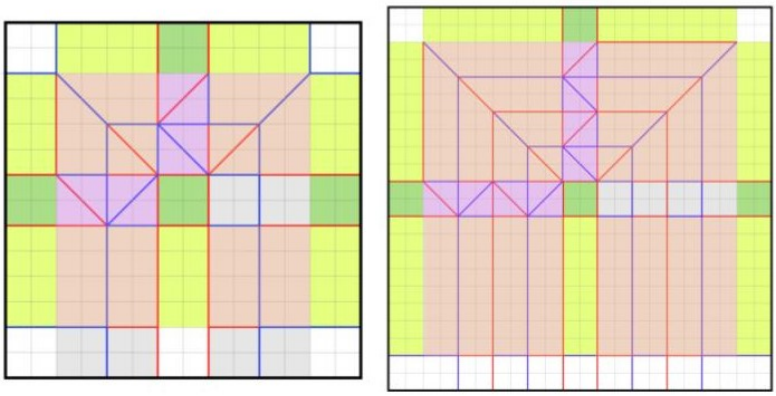


Fig. 11 Comparison of the crease pattern for gadgets with varying edge height-to-thickness ratio. (a) shows the crease pattern of the gadget with a 2:1 edge height-to-thickness ratio, and (b) shows the crease pattern of the same gadget but with a 4:1 edge height-to-thickness ratio. Green indicates that the portion of the paper is an extruded face; yellow indicates that the portion makes up the side of an extruded edge; and white indicates that the portion is not extruded. Gray, orange, and purple regions indicate collapsed regions of the paper. Regions with the same color correspond with each other. Red and blue lines indicate mountain and valley folds, respectively.

The scale factor of any gadget with an $h:1$ edge height to thickness ratio, where $h > 2$, can be calculated using the number of box pleats required, p. The scale factor s is defined as follows:

$$s = 1 + \frac{2p}{3} \tag{3}$$

Substituting Equation 2 into Equation 3 yields

$$s = 1 + \frac{4}{3} \cdot \lceil \frac{h}{4} \rceil \tag{4}$$

3.2.3 Scale factors of gadgets with varying edge height

Given that both methods work for an edge height to thickness ratio of less than or equal to two, we will find in what range from $h \in (0,2]$ each method is most efficient (gives the smaller scale factor). For the box pleating method,

$$s = 1 + \frac{4}{3} \cdot \lceil \frac{h}{4} \rceil.$$

For any $h \in (0,2]$, $s = \frac{7}{3}$ when using the box pleating method. The range under which the non-box-pleating method is more efficient than the box pleating method can be defined using the following inequality:

$$1 + \frac{2h}{3} \leq \frac{7}{3}, h \in (0,2],$$

which holds true for all $h \in (0,2]$. Thus, for all gadgets with an edge height to thickness ratio $h \in (0,2]$, the non-box-pleating method always produces models with the smaller scale factor and is thus always more efficient. Given that gadgets with an edge height to thickness ratio of $h \in (2,\infty)$ can only be folded with the box pleating method, the smallest possible scale factor s for a gadget with an edge height-to-thickness ratio of $h : 1$ is defined as:

$$s = \begin{cases} 1 + \frac{2h}{3} & \text{if} h \in (0,2] \\ 1 + \frac{4}{3} \cdot \lceil \frac{h}{4} \rceil & \text{if} h \in (2,\infty) \end{cases} \tag{5}$$

3.2.4 Varying edge thickness

Any desired increase in edge thickness can be achieved through a corresponding decrease in edge height. For example, a threefold increase in edge thickness corresponds to a threefold decrease in edge height. Any modifications to the crease

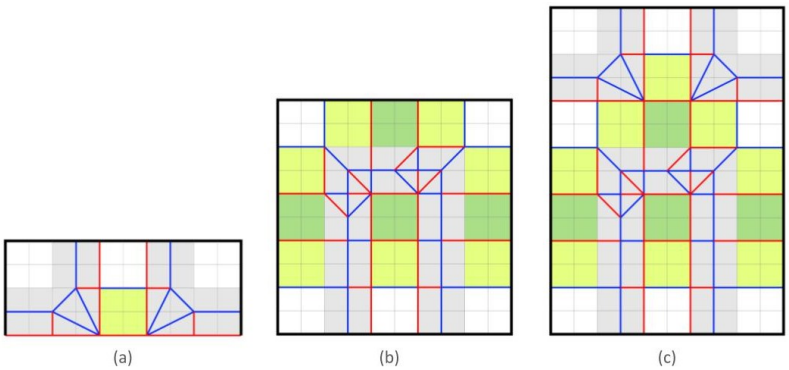


Fig. 12 The process of capping an open edge. (a) shows the crease pattern of the capping unit; (b) shows the crease pattern of the original gadget; and (c) shows the gadget in (b) with one edge capped. Green indicates that the portion of the paper is an extruded face; yellow indicates that the portion makes up the side of an extruded edge; and white indicates that the portion is not extruded. Gray regions indicate collapsed regions of the paper. Red and blue lines indicate mountain and valley folds, respectively.

patterns using any of the two methods discussed will alter the ratio between edge height and thickness, rather than the edge height or the edge thickness individually.

3.3 Capping Open Edges of Gadgets

Any 3D orthogonal graph made up of layered gadgets will result in open edges around the border of the model because the gadgets themselves are uncapped. For each open edge, the crease pattern shown in Figure 12 (a) can be applied to cap the edge. In Figure 12 (c), only the edge pointed upwards is capped; the edge that point laterally are still open (uncapped). If, for a given model, all its edges are capped, then the model as a whole will be watertight.

The scale factor of gadgets is also affected by capping. Let n be an integer such that $n \in [0,2]$ indicates how many lateral sides of the gadget are capped, and m be an integer $m \in [0,2]$ such that m indicates how many vertical sides of the gadget are capped. When adjusting Equation 5 for capping, the scale factor of a gadget will be different along the x and y directions:

$$
\begin{aligned}
s &= (s_x, s_y) \\
s_x &= \begin{cases} \frac{3+2h+(1+h)\cdot n}{3+n} & \text{if } h \in (0,2] \\ \frac{3+\lceil \frac{h}{4} \rceil \cdot 4+(1+h)\cdot n}{3+n} & \text{if } h \in (2,\infty) \end{cases} \\
s_y &= \begin{cases} \frac{3+2h+(1+h)\cdot m}{3+m} & \text{if } h \in (0,2] \\ \frac{3+\lceil \frac{h}{4} \rceil \cdot 4+(1+h)\cdot m}{3+m} & \text{if } h \in (2,\infty) \end{cases}
\end{aligned}
$$

which simplifies to

$$
\begin{aligned}
s &= (s_x, s_y) \\
s_x &= \begin{cases} 1+h\cdot \frac{n+2}{n+3} & \text{if } h \in (0,2] \\ 1+\frac{\lceil \frac{h}{4} \rceil \cdot 4+h\cdot n}{3+n} & \text{if } h \in (2,\infty) \end{cases} \\
s_y &= \begin{cases} 1+h\cdot \frac{m+2}{m+3} & \text{if } h \in (0,2] \\ 1+\frac{\lceil \frac{h}{4} \rceil \cdot 4+h\cdot m}{3+m} & \text{if } h \in (2,\infty) \end{cases}
\end{aligned}
\tag{6}
$$

3.4 Connecting Gadgets

Every gadget has an identical crease pattern motif at its open edges. To connect two gadgets, two edges (one from each gadget) are overlapped by one unit, as shown in Figure 13 (c). Since the open edge motifs are identical in every gadget crease pattern, every set of gadgets can connect using the method shown in Figure 13. Since no other parts of the gadgets, such as the collapsed regions, overlap, any set of connected gadgets can be folded and properly collapsed.

3.5 Scale factors of layered gadgets

Each folded 3D orthogonal graph is made up of an a by b array of gadgets, which can also include empty gadgets. The gadget connections influence the scale factor of the entire model.

The scale factor of a single gadget, taking into account edge height and capping, is shown in Equation 6. Taking into account the overlapping layers in gadget connections, the scale factor of an a by b array of gadgets is defined as follows:

$$
\begin{aligned}
s &= (s_x, s_y) \\
s_x &= \begin{cases} \frac{3+2h+(2h+2)\cdot(a-1)+(1+h)\cdot n}{3+2\cdot(a-1)+n} & \text{if } h \in (0,2] \\ \frac{3+\lceil\frac{h}{4}\rceil\cdot 4+(2h+2)\cdot(a-1)+(1+h)\cdot n}{3+2\cdot(a-1)+n}. & \text{if } h \in (2,\infty) \end{cases} \\
s_y &= \begin{cases} \frac{3+2h+(2h+2)\cdot(b-1)+(1+h)\cdot n}{3+2\cdot(b-1)+n} & \text{if } h \in (0,2] \\ \frac{3+\lceil\frac{h}{4}\rceil\cdot 4+(2h+2)\cdot(b-1)+(1+h)\cdot m}{3+2\cdot(b-1)+m}. & \text{if } h \in (2,\infty) \end{cases}
\end{aligned}
$$

which can be simplified to

$$
\begin{aligned}
s &= (s_x, s_y) \\
s_x &= \begin{cases} 1+h\cdot\frac{2a+n}{2a+n+1} & \text{if } h \in (0,2] \\ 1+(h\cdot\frac{2a+n-2}{2a+n+1}+2\cdot\frac{\lceil\frac{h}{4}\rceil\cdot 2-1}{2a+n+1}) & \text{if } h \in (2,\infty) \end{cases} \\
s_y &= \begin{cases} 1+h\cdot\frac{2b+m}{2b+m+1} & \text{if } h \in (0,2] \\ 1+(h\cdot\frac{2b+m-2}{2b+m+1}+2\cdot\frac{\lceil\frac{h}{4}\rceil\cdot 2-1}{2b+m+1}) & \text{if } h \in (2,\infty) \end{cases}
\end{aligned}
$$

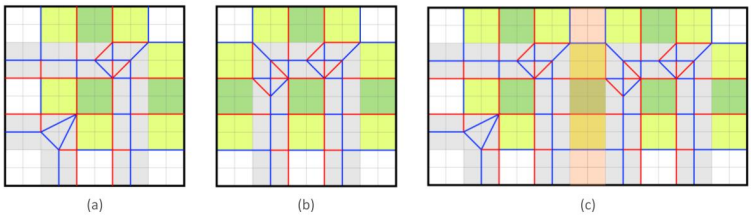


Fig. 13 Example of combining two gadgets. (a) and (b) illustrate the crease patterns of two different gadgets that are combined into (c). The region highlighted in orange is the overlapping region between the two gadgets. Green indicates that the portion of the paper is an extruded face; yellow indicates that the portion makes up the side of an extruded edge; and white indicates that the portion is not extruded. Gray regions indicate collapsed regions of the paper. Red and blue lines indicate mountain and valley folds, respectively.

This form shows the effect of changes in gadget number, height, and capping to the smallest possible scale factor of any model, 1. However, a cleaner simplification is as follows:

$$
\begin{aligned}
s &= (s_x, s_y) \\
s_x &= \begin{cases} 1+h-\frac{h}{2a+n+1} & \text{if } h \in (0,2] \\ 1+h-\frac{\lceil \frac{h}{4} \rceil \cdot 4-3h-2}{2a+n+1} & \text{if } h \in (2,\infty) \end{cases} \\
s_y &= \begin{cases} 1+h-\frac{h}{2b+m+1} & \text{if } h \in (0,2] \\ 1+h-\frac{\lceil \frac{h}{4} \rceil \cdot 4-3h-2}{2b+m+1} & \text{if } h \in (2,\infty) \end{cases}
\end{aligned}
\tag{7}
$$

This form is useful because the terms a and b are always in the denominators of fractions. Assuming h is a finite number, we can now place a limit on the largest scale factor possible for a 3D folded orthogonal graph, which occurs when $a \to \infty$ and $b \to \infty$; in other words, in an infinitely large 3D orthogonal graph. We can

evaluate the largest possible s, which is that of an infinitely large 3D orthogonal graph, by applying limits to Equation 7 as shown:

$$\begin{aligned}
\lim_{a\to\infty} s_x &= \lim_{b\to\infty} s_y \\
\lim_{a\to\infty} s_x &= \begin{cases} \lim_{a\to\infty}(1+h-\frac{h}{2a+n+1}) & \text{if } h \in (0,2] \\ \lim_{a\to\infty}(1+h-\frac{\lceil \frac{h}{4} \rceil \cdot 4-3h-2}{2a+n+1}) & \text{if } h \in (2,\infty) \end{cases} \\
&= 1+h = \lim_{b\to\infty} s_y \\
s_\infty &= (\lim_{a\to\infty} s_x, \lim_{b\to\infty} s_y) \\
&= (1+h, 1+h)
\end{aligned}$$

An infinitely large folded 3D orthogonal graph can be folded from an original sheet of paper with side lengths $1+h$ times larger than the those of the folded graph. Thus, for any folded graph with finite side lengths, the scale factor s from the the original sheet of paper to the folded graph is always less than $1+h$.

4 Proofs

One major difference between our algorithm and those developed by Demaine, Demaine, and Ku (2010) and Demaine, Ku, and Yoder (2018), which generate models with zero thickness edges, is that our crease patterns are not folded seamlessly. As shown in Figure 13, for example, the green squares are continuous in the folded model but contain collapsed layers between them in the crease pattern; these collapsed layers create seams when folded properly. However, as we prove below, for 3D orthogonal graphs with nonzero edge thickness, no seamless crease pattern exists, with the exception of graphs without a single 90° vertex.

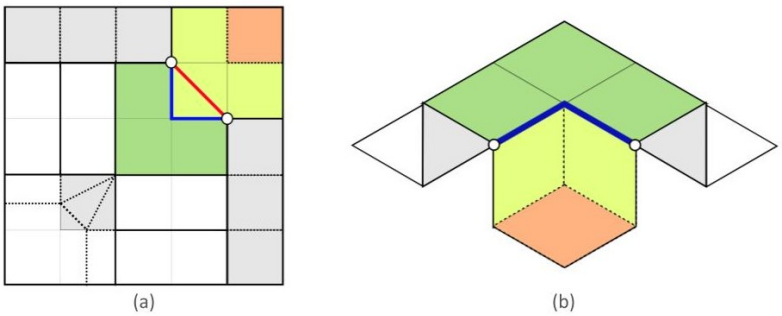


Fig. 14 Visualization of Theorem 1. (a) shows a hypothetical crease pattern for creating a 90° intersection gadget without seams, and (b) shows a hypothetical folded version of (a). Corresponding colors between (a) and (b) indicate corresponding regions of paper.

Theorem 1 *No set of watertight, orthogonal maze gadgets with a wall thickness of* $t > 0$ *and at least one 90° vertex can be seamless.*

Proof Consider the hypothetical crease pattern of a 90° vertex gadget without seams shown in Figure 14 (a) and its hypothetical folding shown in Figure 14 (b). Two white dots are marked as shown in the hypothetical folding, and their corresponding places in the hypothetical crease pattern are also marked. The intrinsic distance (distance along the creases in the folded model) between the two white dots is marked by the thick blue line. The shortest distance between the two points in the crease pattern is marked by the thick red line.

To be foldable, the intrinsic distance must be less than or equal to the shortest distance in the crease pattern. This is because any intrinsic distance in the model cannot increase during the folding process (Demaine, Ku, and Yoder, 2018). The thick blue line and the thick red line form a triangle in Figure 14 (a), and by the triangle inequality, the intrinsic distance in the model is greater than the shortest distance in the crease pattern. Thus, this folding cannot exist.

The only way to increase the length of the red line (shortest distance) without increasing the length of the blue line (intrinsic distance) is to add collapsed regions, which are shown in Figure 9 in gray. Anytime a collapsed region is added, a seam is created. □

Since the edge height to thickness ratio for a folded graph is given, the only variables constraining the scale factor of the graph are the widths of the collapsed regions. It may be tempting to calculate the minimum width w of a collapsed region, where d is the intrinsic distance, by setting the intrinsic distance equal to the shortest distance like shown:

$$\begin{aligned} d &= (\frac{d}{2} + w) \cdot \sqrt{2} \\ d \cdot \frac{\sqrt{2}}{2} &= \frac{d}{2} + w \\ w &= \frac{d}{2} \cdot (\sqrt{2} - 1) \end{aligned} \tag{8}$$

which is less than the width of the collapsed regions shown in Figure 9, which is equal to $\frac{d}{2}$. Thus, it may seem that our crease patterns are not the most efficient possible. However, as we prove, the width of the collapsed regions is constrained by another factor, and that our crease patterns are indeed the most efficient possible, i.e. they produce the smallest possible scale factor.

Theorem 2 *No set of watertight, orthogonal maze gadgets with a wall thickness of $t > 0$ and at least one 90° vertex can have a scale factor smaller than s as defined in Equation 7.*

Proof The scale factor of a folded graph with a specified edge height to thickness ratio is determined by the width of the collapsed region. In our crease patterns, the width w of the collapsed region in a model is equal to $\frac{d}{2} = \frac{h}{r}$, where d is the intrinsic distance, h is the height of the model, and r is the edge height to thickness ratio. Consider the crease pattern shown in Figure 15 (a), where w is less than $\frac{d}{2}$. The shortest possible length of paper between the lines marked in orange is a segment without any collapsed regions, with a length of the height h of the graph edges. If the width w of the collapsed region is constrained so that $w < \frac{d}{2} = \frac{h}{r}$, like the expression

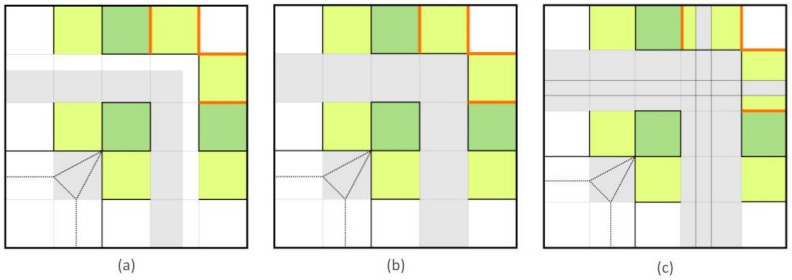


Fig. 15 Visualization of Theorem 2. (a) shows a crease pattern with a collapsed region of $w < \frac{h}{r}$, (b) shows a crease pattern with a collapsed region of $w = \frac{h}{r}$, and (c) shows a crease pattern with a collapsed region of $w > \frac{h}{r}$. Green, yellow, white, and gray regions indicate faces, walls, non-extruded regions, and collapsed regions, respectively. All the crease patterns shown have an edge height-to-thickness ratio r of 1.

derived in Equation 8, there will be insufficient collapsed paper, and since the edge of the collapsed region does not map to the boundary of the non-collapsed region, this crease pattern cannot be folded.

On the other hand, gadgets with a non-efficient scale factor can be created, where $w > \frac{d}{2} = \frac{h}{r}$, like shown in Figure 15 (c). For the height of the gadget to remain as h, a collapsed region with width $w - h$ can be added in the middle of the region between the orange lines.

Thus, the smallest possible width of the collapsed region is $w = \frac{d}{2} = \frac{h}{r}$, which is the width used in our gadgets, as shown in Figure 15 (b). Thus, the scale factor s of our models is the smallest possible scale factor of any 3D folded orthogonal graph. □

5 Conclusion

In this paper, we present an algorithm for transforming any 2D orthogonal graph into a 3D, foldable orthogonal graph with nonzero edge thickness. We also present methods for adjusting the edge height to thickness ratio of our models and derive the corresponding scale factors of the folded graphs. We further demonstrate that the scale factor s of any folded graph generated from our algorithm is always less than $1 + h$, where h is the edge height of the model. Finally, we show that no folded

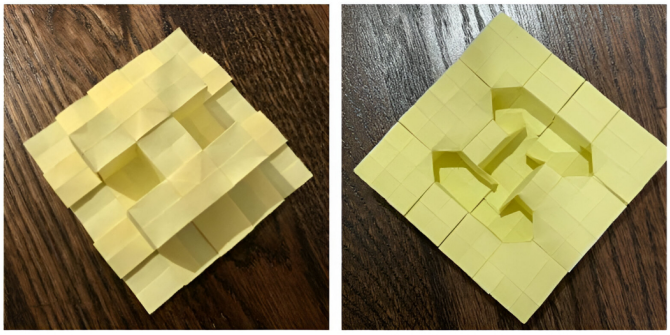
graphs with non-zero edge thickness can be seamless, and that our folded graphs achieve the most efficient (lowest) scale factor possible.

Our algorithm has significant implications for compliant design, a key area in engineering that involves creating structures without hinges or joints. By enabling the efficient creation of complex designs from a flat, continuous sheet of material, our algorithm supports the development of foldable structures that are lightweight, compact, and manufacturable using materials such as sheet metal, plastics, or composites, even if in this paper our algorithm is illustrated in the context of origami.

Moreover, our algorithm complements those of Demaine, Demaine, and Ku (2010) and Demaine, Ku, and Yoder (2018) by allowing for the ability to customize the edge height-to-thickness ratio of the folded models. As a result, our algorithm has important implications for a variety of engineering tasks in fields such as aerospace, biomedical, and mechanical engineering, where the deployability, expandability, and structural stability of devices are highly valued.

Appendix

An orthogonal maze showing the letter ”H,” with an edge height to thickness ratio of 1:1, generated by our algorithm.



References

[1] Erik D. Demaine, Martin L. Demaine, and Jason Ku, "Folding any orthogonal maze," in *Origami^5: Proceedings of the Fifth International Conference on Origami in Science, Mathematics, and Education (OSME 2010)*, A K Peters, Singapore, 2010, pages 449-454.

[2] Erik D. Demaine, Jason S. Ku, and Madonna Yoder, "Folding triangular and hexagonal mazes," in *Origami^7: Proceedings of the Seventh International Conference on Origami in Science, Mathematics, and Education (OSME 2018)*, vol. 2, Tarquin, Oxford, England, 2018, pages 647-652.

[3] Erik D. Demaine and Tomohiro Tachi, "Origamizer: A practical algorithm for folding any polyhedron," in *Proceedings of the 33rd International Symposium on Computational Geometry (SoCG 2017)*, vol. **77**, Brisbane, Australia, 2017, pages 34:1-34:15.

[4] Shannon A. Zirbel, Brian P. Trease, Mark W. Thomson, Robert J. Lang, Spencer P. Magleby, and Larry H. Howell, "HanaFlex: A large solar array for space applications," in *Micro- and Nanotechnology Sensors, Systems, and Applications VII*, Proceedings of the International Society for Optics and Photonics, vol. 9467, International Society for Optics and Photonics, Maryland, 2015, page 94671; doi:10.1117/12.2177730.